



UNIVERSIDAD POLITÉCNICA DE SINALOA
PROGRAMA ACADÉMICO DE INGENIERÍA EN
INFORMÁTICA

Tesina

**“Desarrollo de aplicaciones web utilizando
Angular como framework”**

Para obtener la Acreditación de las Estadías Profesionales
y Contar con los Créditos para el Grado de Ingeniero en
Informática.

Autor:

Zuñiga Vazquez

Luis Ramon

Asesor:

Ostos Robles Rodolfo

Asesor OR:

Lic. Daniel Bautista Urenda

Mazatlán, Sinaloa, 11 de Enero de 2020.

Carta de Aceptación de Tema de Tesina.

	UNIVERSIDAD POLITÉCNICA DE SINALOA	
<hr/>		
C. ZUÑIGA VAZQUEZ LUIS RAMON	Folio 2017030783-2020-A021	
Presente.-		
<p>Por medio de la presente me permito comunicarle que es de aceptarse el tema de tesina, el cuál se ha solicitado bajo el título:</p>		
<p>"Desarrollo de aplicaciones web utilizando Angular como framework"</p>		
<p>mismo que usted desarrollará con objeto de dar lugar a los tramites conducentes para la acreditación de la asignatura de Estadías Profesionales de la Unidad Académica de:</p>		
<p>Ingeniería en informática</p>		
<p>Así mismo, Le comunico que para el desarrollo de la citada tesina le ha sido asignado como director de la misma a: Dr. Rodolfo Ostos Robles, y como asesores a Dra. Vanessa Guadalupe Félix Aviña y M.C. Alejandro Pérez Pasten Borja.</p>		
<p>Sin otro particular por el momento, aprovecho la ocasión para enviarle un cordial saludo.</p>		
<p>Atentamente</p> 		
<p>Dr. Rodolfo Ostos Robles Director del Programa Académico de Ingeniería en Informática Universidad Politécnica de Sinaloa</p>		
<hr/>		
	<i>"2020, Año de Leona Vicario, Benemérita Madre de la Patria"</i>	
<p>Carretera Municipal Libre Mazatlán Higuera Km. 3, Col. Genaro Estrada, C.P. 82199, Mazatlán, Sin. Tel (669) 1800695 y 96 www.upsin.edu.mx</p>		

Carta de Aceptación de Tesina.

	UNIVERSIDAD POLITÉCNICA DE SINALOA	
<hr/>		
C. ZUÑIGA VAZQUEZ LUIS RAMON Presente.-	Folio 201703 0783-2020-A021	
<p>Por este conducto le envío un cordial saludo y aprovecho la ocasión para notificarle que el jurado que le fue asignado para evaluar la tesina desarrollada en las etapas profesionales denominada "Desarrollo de aplicaciones web utilizando Angular como framework" y que después de ser revisada en reunión de sinodales, ante la Dirección de la Unidad Académica de Ingeniería en Informática, integrada por:</p>		
PRESIDENTE DEL JURADO: Dr. Rodolfo Ostos Robles		
SINODAL: Dra. Vanessa Guadalupe Félix Aviña		
SINODAL: M.C. Alejandro Pérez Pastén Borja		
<p>Ha decidido autorizar y aceptar la digitalización de la misma por el participante, conforme a la normatividad vigente y cumpliendo con los requisitos para tal caso.</p>		
<p>Agradeciendo la atención a la presente, le reitero a Usted mi atenta consideración y respeto.</p>		
<p>Atentamente</p> 		
<p>Dr. Rodolfo Ostos Robles Director del Programa Académico de Ingeniería en Informática Universidad Politécnica de Sinaloa</p>		
	<p><i>"2020, Año de Leona Vicario, Benemérita Madre de la Patria".</i></p>	
<p>Carretera Municipal Libre Mazatlán Higuera Km. 3, Col. Genaro Estrada, C.P. 82199, Mazatlán, Sin. Tel (669) 1800695 y 96</p>		
<p>www.upsin.edu.mx</p>		

Carta de Aceptación de la Empresa.



Mazatlán Sin, 20 de agosto del 2020.


LIC. EDUARDO CRESPO CAMPA.
DIRECTOR DE VINCULACIÓN, DIFUSIÓN Y EXT. UNIVERSITARIA.
UNIVERSIDAD POLITÉCNICA DE SINALOA.

PRESENTE

Por este medio, hago de su conocimiento que el alumno(a) Luis Ramon Zuñiga Vazquez con número de matrícula 2017030783, de la carrera de Ingeniería en Informática, ha sido aceptado para realizar su estadía práctica, en Xmaz, durante el período que comprende del 7 de septiembre al 4 de diciembre, para cubrir un total de 600 horas.

Dicho alumno realizará actividades dentro del departamento de programación, bajo la supervisión del LIC DANIEL BAUTISTA URENDA, Sin otro particular, le envío un cordial saludo.

Atte.


LSC DANIEL BAUTISTA URENDA
Cel .669 1553264
Ofic. 669 668 7711



www.xmaz.mx

Veracruz #1136 Sánchez Celis.
Mazatlán, Sinaloa. México.

Carta Donde la Empresa Aprueba al Alumno.



Mazatlán Sin, 1 de diciembre del 2020.

LIC. EDUARDO CRESPO CAMPA.
DIRECTOR DE VINCULACIÓN, DIFUSIÓN Y EXT. UNIVERSITARIA.
UNIVERSIDAD POLITÉCNICA DE SINALOA.

PRESENTE

Por este medio, hago de su conocimiento que el alumno(a) el Luis Ramon Zuñiga Vazquez con número de matrícula 2017030783 , de la carrera de Ingeniería en Informática, ha cumplido con 600 horas correspondientes a estadia final , en esta Xmaz, durante el período que comprende del 7 de septiembre al 4 de diciembre.

Dicho alumno realizará actividades dentro del departamento de programación, bajo la supervisión del LIC DANIEL BAUTISTA URENDA, Sin otro particular, le envío un cordial saludo.

Atte.


LSC DANIEL BAUTISTA URENDA
Cel .669 1553264
Ofic. 669 668 7711

Índice Temático.

Resumen.	7
Abstract.	7
Introducción.	7
CAPÍTULO I	8
1.1- Antecedentes.	9
1.1.1.- Localización.	10
1.1.2.- Objetivos y Prioridades de la Empresa.	11
1.1.3.- Organigrama de la Empresa.	12
1.1.4.- Visión.	12
1.1.5.- Misión.	12
1.1.6.- Valores.	13
1.2.- Planteamiento del Problema.	13
1.2.1.- Propuesta de Investigación.	14
1.2.2.- Objetivos	14
1.2.2.1.- Objetivo General.	14
1.2.2.2.- Objetivos Específicos.	14
1.2.3.- Preguntas de Investigación.	15
1.2.4.- Hipótesis.	15
1.2.5.- Limitaciones y Supuestos.	15
1.2.6.- Relevancia.	15
CAPÍTULO II	16
2.1.- Marco Conceptual.	17
2.2.1- Aplicaciones web.	17
2.2.2- Interfaz.	18
2.2.3- Navegadores.	19
2.2.4- Frameworks.	20
2.2.4.1- Angular.	22
2.2.4.2- Node.js.	23
2.2.5- Lenguajes de programación.	25
2.2.5.1- HTML.	26
2.2.5.2- CSS.	27
2.2.5.3- SASS.	28
2.2.5.4- Javascript.	31

2.2.5.5- Typescript.	32
2.2.6- Librerías.	33
2.2.6.1- JQuery.	34
CAPÍTULO III	36
3.1.- Desarrollo.	37
3.1.1.-Herramientas de desarrollo	37
3.1.2.-Preparación de HTML	38
3.1.2.1.-Creación de un nuevo proyecto	39
3.2.-Desarrollo de aplicación web	41
3.2.1.-Página inicial(main)	41
3.2.2.-Preparación de Angular	44
3.2.2.2.-Creación de un nuevo proyecto	45
3.3.-Desarrollo de aplicación web	46
3.3.1.-Página inicial(main)	48
3.4- Resultados.	50
3.5- Discusión.	50
Conclusiones.	50
Bibliografía.	51

Resumen.

El presente documento ha sido realizado con la finalidad de acreditar la carrera de ingeniería en informática en la Universidad Politécnica de Sinaloa. El contenido expuesto en el presente trabajo hace mención del desarrollo de aplicaciones web para una empresa dedicada a la realización de productos procesados, la cual tiene como finalidad obtener una mejor expansión de su producto mediante el uso de las tecnologías web.

Abstract.

This document has been prepared in order to accredit the degree in computer science engineering at the Polytechnic University of Sinaloa. The content exposed in this work makes mention of the development of web applications for a company dedicated to the production of processed products, which aims to obtain a better expansion of its product using web technologies.

Introducción.

Desde finales del siglo XXI la tecnología crece a pasos enormes y cada vez está más presente en el día a día. En algún punto toda empresa debe de buscar o de encontrar la manera de no quedarse atrás y actualizarse.

La implementación de frameworks para el desarrollo de una página ayuda a la agilización de procesos gracias a las librerías con las que cuentan.

Durante la realización de estadías en la empresa **Xmaz** se trabajará en un proyecto de desarrollo web, para uno de sus clientes, ya que dicha empresa dedicada a la fabricación de productos procesados no cuenta con tecnologías webs, esto lo hacen con la finalidad de poder expandir más su mercado mediante el uso de internet.

CAPÍTULO I

Antecedentes y planteamiento del problema

1.1- Antecedentes.

Xmaz es una empresa la cual cuenta con más de 10 años de experiencia sirviendo al sector de Tecnologías de la información, enfocada principalmente al desarrollo de software y venta de equipo de cómputo, ubicada en Mazatlán, Sinaloa México. El objetivo principal de Xmaz es crear sinergia con sus clientes y convertirse en socios de negocios pequeños, medianos y grandes, con el fin de brindarles ayuda para el logro de sus objetivos, tales como, generar una ventaja competitiva, expandir sus mercados, asegurar la continuidad de su negocio facilitando su productividad basado en las estrategias tecnológicas. Sin embargo, Xmaz también ofrece consultoría y desarrollo en sap abarcando clientes de diversas partes de la República Mexicana con los cuales actualmente operan sus puntos de venta.

Logotipo Xmaz.



Figura 1.1 - Fuente: Xmaz

Gracias a la estrategia organizacional de la empresa, continúa con un enfoque de satisfacción a los clientes de diversos sectores, agrícolas, de construcción, de calzado, textil, ferretero, minorista, etc.

Actualmente se busca un crecimiento exponencial basado en la creación de nuevos proyectos, en los cuales se desea alcanzar un mayor número de clientes con la finalidad de brindarles mayores beneficios.

Xmaz tuvo sus inicios en la ciudad de Mazatlán con el establecimiento de ABC consulting en febrero del 2009, gracias a la visión de Daniel Bautista Urenda y otros socios, quienes se aventuraron al inicio de la facturación electrónica, donde comenzó a tener un crecimiento en la localidad, siendo de las primeras empresas de facturación electrónica.

Logotipo ABCconsulting



Figura 1.2 - Fuente: Xmaz.

Por cuestiones personales y estrategia comercial, en el año 2016, se cambió el nombre del negocio a Xmaz donde actualmente se prosigue sin socios. Con el crecimiento de la demanda de usuarios de la facturación electrónica y la entrada de competidores en la zona, fue necesario el establecimiento de mejores estrategias para seguir siendo un negocio rentable y ofrecer un buen servicio a los clientes, empezando a desarrollar nuevos canales de negocios, tales como, desarrollo web y venta de equipo de cómputo.

1.1.1.- Localización.

Xmaz se encuentra ubicado en la calle Veracruz 1136, colonia Sánchez Celis, Mazatlán, Sinaloa, México.

Ubicación de Xmaz



Figura 1.3 - Fuente: Google Maps.

Fachada de Xmaz



Fotografía 1.1 - Fuente: Del autor.

1.1.2.- Objetivos y Prioridades de la Empresa.

El desarrollo de la tecnología es un ámbito en el que Xmaz ha invertido importantes esfuerzos, por lo que sus principales objetivos a corto plazo son lograr posicionar dentro del top 10 de las páginas más populares y visitadas al nuevo portal de facturación electrónica “facturaxmaz.mx”, con el fin de convertirse en una de las principales empresas de desarrollo en la localidad y seguir posicionando el punto de venta SAP dentro de la República Mexicana. Dejando como objetivo a largo plazo, el desarrollo de un nuevo canal de negocio dentro del área de e-commerce, en el cual se podrá brindar un servicio de venta de productos tecnológicos a un mejor precio y con facilidades de pago, con el fin de beneficiar al cliente.

1.1.3.- Organigrama de la Empresa.

Xmaz es una pequeña empresa la cual se encuentra específicamente en el sector de la tecnología, tal como, desarrollo web, desarrollo de aplicaciones móviles y desarrollo de software. Estas tienen una gran importancia dentro de la empresa ya que estas son los pilares fundamentales, por lo cual dentro del organigrama podemos observar la importancia de dichas áreas.

Organigrama de Xmaz



Figura 1.5 - Fuente: Xmaz.

1.1.4.- Visión.

Convertirse en una empresa líder en el sector de las Tecnologías de Información, buscando ser siempre un socio de negocios que aporte al cliente las estrategias que le ayuden al logro de sus metas.

1.1.5.- Misión.

Satisfacer las necesidades de los clientes a través de soluciones tecnológicas innovadoras, con la finalidad de apoyarlos en el logro de sus objetivos.

1.1.6.- Valores.

Valores de Xmaz:

- Respeto
- Confianza
- Calidad
- Innovación
- Eficiencia
- Lealtad

1.2.- Planteamiento del Problema.

La tecnología en la actualidad crece a pasos agigantados y cada vez está más presente en el día a día. En algún punto toda empresa debe de buscar de encontrar la manera de no quedarse atrás y actualizarse.

Debido a esto muchas empresas se ven en la necesidad de incursionar en las nuevas tecnologías, por lo tanto las empresas se ven en la necesidad de buscar empresas de desarrollo web para tener un mejor alcance en el crecimiento y reconocimiento de sus productos para poder lograr obtener un mejor reconocimiento ante el público en general y tener un mejor crecimiento como empresa.

EL desarrollo de la aplicación web que se requiere debe ser rápido y confiable debido a que se debe aprovechar todo el tiempo que se disponga para la creación de dicha aplicación, todo esto con la finalidad de obtener un resultado eficaz, eficiente y competitivo en el mercado.

1.2.1.- Propuesta de Investigación.

Una vez ya planteada la problemática, se realizó el planteamiento del desarrollo de una página web para dar a conocer la empresa, como el lugar donde se encuentra, los productos que tienen en venta y servicios.

Para el desarrollo de este proyecto fue necesario investigar acerca de cómo realizar un sitio web implementando el uso de frameworks con el cual se pueda trabajar. Se realizó una investigación para determinar cuál sería el más adecuado para el desarrollo del proyecto.

1.2.2.- Objetivos.

A partir del planteamiento del problema y la propuesta de investigación se definen los siguientes objetivos que guiarán el rumbo de la investigación.

1.2.2.1.- Objetivo General.

Implementar **Angular** como framework para la facilitación del desarrollo de una página web. Ofreciendo un proyecto dividido en dos partes, Front-end y Back-end.

1.2.2.2.- Objetivos Específicos.

Los siguientes puntos hacen mención a las actividades específicas a realizar para lograr el propósito mencionado con anterioridad en el objetivo general.

Investigar el manejo del framework Angular para la realización del proyecto.

Utilización de *TypeScript*.

Mejorar la capacidad de desarrollo del proyecto para un mejor crecimiento.

Implementar Nodejs para el lado del servidor.

1.2.3.- Preguntas de Investigación.

Durante la creación de nuestro proyecto web surgieron las siguientes cuestiones:

- ¿Qué ventajas ofrece la implementación de un framework para js en la creación de una página?
- ¿Por qué es importante el uso de librerías de Angular y Node js?
- ¿Es necesario el uso de frameworks para la realización de un proyecto?

1.2.4.- Hipótesis.

La implementación de Angular como framework para el desarrollo de un sitio web permite que el proceso de creación sea más rápido debido a que angular cuenta con una extensa cantidad de procesos que nos facilita nuestro trabajo, de esta manera nos enfocamos solamente en la lógica principal y así poder crear una página web funcional.

1.2.5.- Limitaciones y Supuestos.

La creación de una aplicación web es un trabajo extenso, debido a que se debe diseñar y desarrollar toda lógica necesaria para poder llevar a cabo un trabajo eficiente.

Las limitaciones más importantes es el tiempo de estimación que se tiene para el desarrollo de dicha página web.

El supuesto de este proyecto es poder lograr completar todos los objetivos que se marcaron acorde a la creación de la aplicación web, como estar preparados para los cambios de último momento por parte de los líderes del proyecto, además de que el proyecto quede libre para futuras mejoras.

1.2.6.- Relevancia.

La creación de un proyecto con la implementación de Angular genera un marco de posibilidades para el desarrollo de una aplicación web, debido a que Angular provee una gran cantidad de herramientas necesarias para facilitar el desarrollo web gracias a esto se obtiene una página web eficaz, de fácil manejo y creación.

CAPÍTULO II

Estado del arte

2.2.2- Interfaz.

Las interfaces web tienen ciertas limitaciones en las funcionalidades que se ofrecen al usuario. Hay funcionalidades comunes en las aplicaciones de escritorio como dibujar en la pantalla o arrastrar-y-soltar que no están soportadas por las tecnologías web estándar[4].

Los desarrolladores web generalmente utilizan lenguajes interpretados (script) en el lado del cliente para añadir más funcionalidades, especialmente para ofrecer una experiencia interactiva que no requiera recargar la página cada vez (lo que suele resultar molesto a los usuarios). Recientemente se han desarrollado tecnologías para coordinar estos lenguajes con las tecnologías en el lado del servidor. Como ejemplo, AJAX es una técnica de desarrollo web que usa una combinación de varias tecnologías[5].

Interfaz web



Figura 2.2 - Fuente: <https://es.vecteezy.com/arte-vectorial/110250-interfaz-de-usuario-de-web-plana-libre-vector-de-fondo>

2.2.3- Navegadores.

Los Navegadores son herramientas informáticas que utilizamos para, normalmente, navegar por Internet y visitar cualquier página web, además de para hacer otras tareas como ver documentos, observar videos o reproducir contenido multimedia de cualquier tipo. Son un tipo de software realmente habitual y muy utilizado actualmente [6].

Insistimos en que es algo que se usa con mucha frecuencia, ya que moverse por Internet es algo que hace la mayoría del público en los tiempos que corren. Sea para echar un vistazo al correo, para ojear una web de noticias o para entrar en YouTube, el navegador web suele ser el medio indicado para llevar a cabo dicha tarea, de ahí que esté tan presente y su uso está tan normalizado.

Podemos encontrar muchos navegadores diferentes en la actualidad. Numerosas empresas como Google, Microsoft o incluso Mozilla se han lanzado a este terreno para llevar al público propuestas del estilo de Chrome o Internet Explorer, este último todo un nombre histórico en lo que a navegar por la red se refiere [7].

Los Navegadores sirven para poder visitar prácticamente cualquier web que haya en Internet. A pesar de que esta es su principal utilidad, también pueden usarse para reproducir cualquier tipo de contenido que podamos imaginar y que se encuentre alojado en el entorno digital. Desde vídeos hasta música, sonido o imágenes se pueden ver en este software.

Pensando en ellos desde un punto de vista empresarial, además de ayudar a que los miembros de una compañía busquen información o usen algunas herramientas disponibles solo en formato web, es también un programa que se vuelve esencial en el proceso de visibilización de una marca. Cualquier formato publicitario a publicar en Internet sería invisible si no hubiera un navegador

Una de las herramientas que más usamos en nuestro día a día, independientemente de que lo hagamos como usuarios domésticos o como profesionales. Sin los navegadores, consultar el contenido que ofrece y que amplía Internet cada día que

pasa sería una tarea prácticamente imposible. Son el vehículo que permite conectar nuestro dispositivo al entorno digital a través de una conexión a Internet [8].

Navegadores web



Figura 2.3 - Fuente: www.ecured.cu/Navegador_web

2.2.4- Frameworks.

Un Framework, que se podría traducir aproximadamente como marco de trabajo, es el esquema o estructura que se establece y que se aprovecha para desarrollar y organizar un software determinado. Esta definición, algo compleja, podría resumirse como el entorno pensado para hacer más sencilla la programación de cualquier aplicación o herramienta actual [9].

Este sistema plantea varias ventajas para los programadores, ya que automatiza muchos procesos y además facilita el conjunto de la programación. Es útil, por ejemplo, para evitar el tener que repetir código para realizar funciones habituales en un rango de herramientas, como puede ser el acceder a bases de datos o realizar llamadas a Internet. Todas estas tareas son las que se realizan de forma mucho más fácil cuando se trabaja dentro de un framework [10].

Plantea muchas ventajas y, además, es capaz de hacer que se realicen incluso labores mucho más complejas que, por otros medios, serían imposibles de plantear siquiera a la hora de programar algo. No obstante, su utilidad es algo que depende del tipo de programa y del contexto en el que vaya a emplearse.

Existen muchos frameworks diferentes, como pueden ser Meteor para trabajar con JavaScript en web y móviles o el más que popular .NET Framework de Microsoft.

Todos ellos poseen herramientas específicas y funciones características que encajan mejor con el tipo de aplicación a desarrollar, así como el uso que se le pretenda dar una vez terminada. Son, a día de hoy, algo imprescindible para cualquier programador, sea amateur o profesional [11].

Un Framework sirve para poder escribir código o desarrollar una aplicación de manera más sencilla. Es algo que permite una mejor organización y control de todo el código elaborado, así como una posible reutilización en el futuro. Debido a esto, garantiza una mayor productividad que los métodos más convencionales y una minimización del coste al agilizar las horas de trabajo volcadas en el desarrollo.

Por otra parte, su acción es algo que afecta también a los errores, minimizándolos considerablemente. En definitiva, es algo que brinda una ayuda general y más que considerable al programador y desarrollador, haciendo que sus labores sean mucho más sencillas [12].

Frameworks



Figura 2.4 - Fuente: https://medium.com/@Suretek_infosoft/top-web-development-frameworks-2019-a32b4c0db379

2.2.4.1- Angular.

Hoy en día existe un gran mercado en el sector del desarrollo web, y con esto se han creado distintas tecnologías para la creación de plataformas en línea. Para entender qué es Angular primero debemos saber la arquitectura de una plataforma web [13]:

El usuario al ingresar a una dirección en línea realiza una petición que viaja por el internet hacia el servidor en donde está alojada la plataforma que desea visitar [14].

Al haber interacción entre los equipos de los usuarios y los servidores en internet, se debe tomar en cuenta las distintas tecnologías para ambos extremos, el lado del usuario se le conoce como “frontend” y el lado del servidor se le conoce como “backend”.

Pero, ¿en qué extremo se encuentra Angular? La respuesta es el frontend. Angular es un framework mantenido por Google para el desarrollo de aplicaciones web del lado del cliente, su estructura está hecha en base a TypeScript, el cual es una versión avanzada de JavaScript (lenguaje principal para el desarrollo frontend) [15].

Angular comenzó en el 2010 bajo el nombre de AngularJS. AngularJS proporcionaba controladores y directivas para dividir el código en JavaScript para poder manejar la interfaz gráfica de usuario y las peticiones al servidor por aparte. Para la segunda versión de AngularJS, el equipo desarrollador decidió cambiar la nomenclatura del framework a simplemente Angular.

Uno de los objetivos de Angular es fortalecer la estructura MVC en el desarrollo web (Modelo-Vista-Controlador) y las SPA (Single Page Application). El MVC es una arquitectura que se encarga de separar la lógica del código en una aplicación, en donde la vista es la interfaz gráfica de usuario y esta cambia de acuerdo a un controlador que gestiona el contenido en ella. Las Single Page Application son páginas que cargan únicamente un archivo HTML en el cliente y que su contenido cambia dinámicamente a medida el usuario interactúa con ella [16].

Arquitectura de Angular

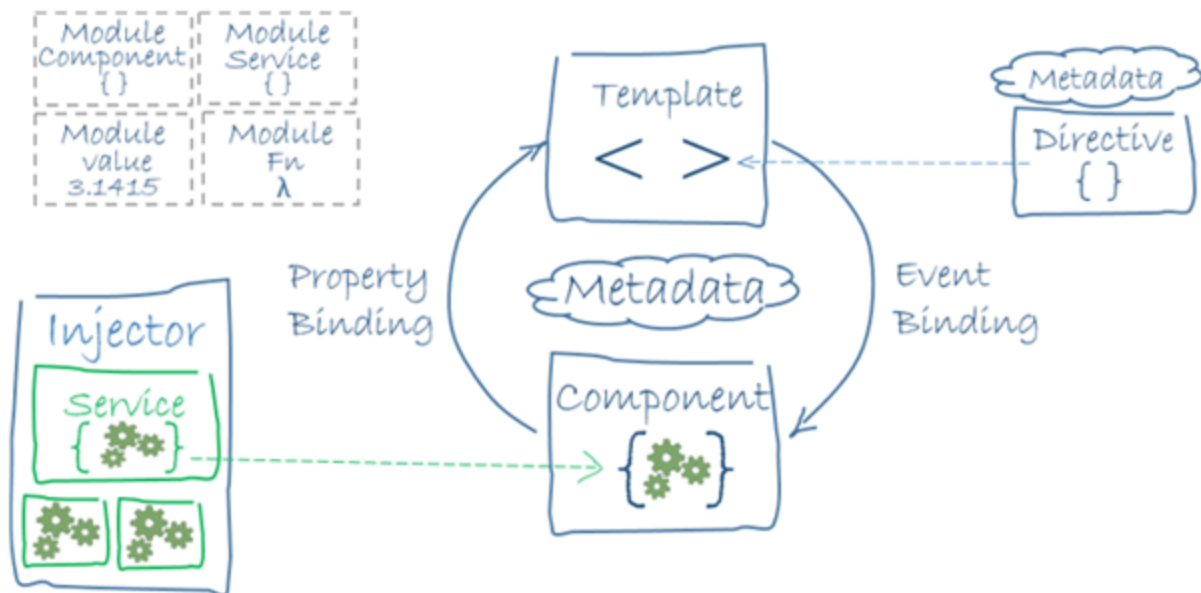


Figura 2.5 - Fuente: <https://openwebinars.net/blog/que-es-angular/>

2.2.4.2- Node.js.

Node.js es un entorno JavaScript que nos permite ejecutar en el servidor, de manera asíncrona, con una arquitectura orientada a eventos y basado en el motor V8 de Google. Es una plataforma que avanza muy rápidamente y cada vez está más presente en el mercado[17].

El motor V8 compila Javascript en código máquina nativo en vez de interpretarlo en el navegador, consiguiendo así una velocidad mucho más alta. Node es de código abierto y puede ejecutarse en Mac OS X, Windows y Linux.

Tradicionalmente Javascript se ha utilizado únicamente para tareas menores en el navegador, pero actualmente ya es un lenguaje de programación tan potente como cualquier otro. Además JavaScript dispone de un modelo de eventos excelente ideal para la programación asíncrona.

JavaScript es un lenguaje conocido por millones de programadores, así que la curva de aprendizaje de Node es reducida, ya que no suele ser necesario aprender un nuevo lenguaje[18].

Además de la alta velocidad de ejecución, Node.js dispone del Bucle de Eventos (Event Loop), que permitirá gestionar enormes cantidades de clientes de forma asíncrona. Tradicionalmente para trabajar de forma asíncrona las aplicaciones se valían de la programación basada en hilos (programming threaded applications), pero esto supone la utilización (normalmente ineficaz) de un espacio de memoria que va escalando a medida que la cantidad de clientes conectados a nuestra aplicación aumenta.

Generalmente cada hilo supone la utilización de 2MB de memoria, lo que en un sistema de 8GB de RAM supone un máximo teórico de conexiones de 4.000 usuarios. Por lo tanto, si necesitamos gestionar grandes cantidades de conexiones tendremos que ampliar el número de servidores.

Node.js resuelve este problema cambiando la manera de realizar las conexiones con el servidor. En vez de generar un nuevo hilo E/S para cada cliente, cada conexión dispara la ejecución de un evento dentro del proceso del motor de Node. De este modo, Node permite que un solo servidor que lo ejecute pueda soportar decenas de miles de conexiones [19].

Nodes.js

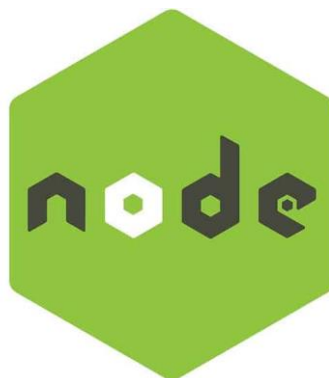


Figura 2.6 - Fuente: <https://www.altiria.com.mx/enviar-sms-node-js/>

2.2.5- Lenguajes de programación.

En informática, se conoce como lenguaje de programación a un programa destinado a la construcción de otros programas informáticos. Su nombre se debe a que comprende un lenguaje formal que está diseñado para organizar algoritmos y procesos lógicos que serán luego llevados a cabo por un ordenador o sistema informático, permitiendo controlar así su comportamiento físico, lógico y su comunicación con el usuario humano [20].

Dicho lenguaje está compuesto por símbolos y reglas sintácticas y semánticas, expresadas en forma de instrucciones y relaciones lógicas, mediante las cuales se construye el código fuente de una aplicación o pieza de software determinado. Así, puede llamarse también lenguaje de programación al resultado final de estos procesos creativos [21].

La implementación de lenguajes de programación permite el trabajo conjunto y coordinado, a través de un conjunto afín y finito de instrucciones posibles, de diversos programadores o arquitectos de software, para lo cual estos lenguajes imitan, al menos formalmente, la lógica de los lenguajes humanos o naturales.

No deben confundirse, sin embargo, con los distintos tipos de lenguaje informático. Estos últimos representan una categoría mucho más amplia, en donde están contenidos los lenguajes de programación y muchos otros protocolos informáticos, como el HTML de las páginas web [22].

Lenguajes de programación



Figura 2.7 - Fuente:

<https://codigoonclick.com/mejores-lenguajes-programacion-para-2018/>

2.2.5.1- HTML.

HTML, que significa Lenguaje de Marcado para Hipertextos (HyperText Markup Language) es el elemento de construcción más básico de una página web y se usa para crear y representar visualmente una página web. Determina el contenido de la página web, pero no su funcionalidad. Otras tecnologías distintas de HTML son usadas generalmente para describir la apariencia/presentación de una página web (CSS) o su funcionalidad (JavaScript)[23].

HTML le da "valor añadido" a un texto estándar en español. Hipertexto se refiere a enlaces que conectan una página Web con otra, ya sea dentro de una página web o entre diferentes sitios web. Los vínculos son un aspecto fundamental de la Web. Al subir contenido a Internet y vincularlo a páginas de otras personas, te haces participante activo de esta Red Mundial [24].

HTML usa "markup" o marcado para anotar textos, imágenes, y otros contenidos que se muestran en el Navegador Web. El lenguaje de marcado HTML incluye "elementos" especiales tales como <head>, <title>, <body>, <header>, <article>, <section>, <p>, <div>, , , y muchos otros más [25].

Html



Figura 2.8 - Fuente: <https://www.hostinger.mx/tutoriales/que-es-html/>

2.2.5.2- CSS.

CSS es un lenguaje de programación que se utiliza para definir el estilo y el aspecto de un documento que se ha escrito a través de un lenguaje de etiquetas, como HTML. Conocido también como hojas de estilo en cascada, es el que se emplea para dar colores, indicar tipos de letra o incluso señalar aspectos como el espacio entre elementos para dotar de estilo a una web [26].

Es uno de los pilares fundamentales del desarrollo y el diseño web, como también una de las mayores pesadillas de todo programador. Dado que es el conjunto de reglas que se indican a los navegadores para que muestren los elementos de una página, realizar las instrucciones incorrectas puede desembocar en la presentación de una web imposible de visualizar correctamente.

Generalmente, cada una de las reglas CSS que se emplean están compuestas por una serie de propiedades, o properties, que poseen valores con los que se indica la presentación de todo el contenido dictado en HTML; y, por otra parte, selectores con los que se indican qué elementos se verán afectados por dichas propiedades y sus valores.

Todo este conjunto es el que conforma los archivos que se han de indicar a los navegadores para determinar la apariencia de las páginas web. No obstante, lo complicado de su manejo y la necesidad de aprender nociones de programación son algo que suele llevar a muchos a confiar en plantillas predeterminadas u otras soluciones mucho más cómodas que hacen estas tareas infinitamente más sencillas, aunque renunciando a la total personalización [27].

CSS sirve para poder dar un estilo a cualquier web y hacerlo con las reglas que se deseen. Es el lenguaje con el que el programador y el navegador se comunican para indicar a este último cómo ha de distribuir los contenidos de una web en base a las instrucciones indicadas en los correspondientes ficheros.

A su vez, tiene como principal finalidad conseguir estilos diferentes y a medida de las necesidades de cada página. Aunque, desafortunadamente, exige un amplio

conocimiento de programación y grandes nociones de lenguaje HTML también para conseguir la conexión perfecta entre estilo y contenidos [28].

Css



Figura 2.9 - Fuente:

<http://lineadecodigo.com/css/centrar-texto-con-css-en-tu-pagina-web/>

2.2.5.3- SASS.

Para entender qué es SASS y por qué puedes necesitar este lenguaje, debes entender el significado de CSS. Las páginas web se basan inicialmente en el lenguaje de marcado HTML. Pero si deseas diseñar en HTML, en cada página HTML tienes que vincular el contenido al diseño. En pocas palabras: el HTML no está pensado para dar formato de diseño y, por lo tanto, está muy limitado en este sentido. La hoja de estilo en cascada (CSS, del inglés Cascading Style Sheet) define el aspecto visual de las páginas HTML. Al igual que una plantilla, el CSS define el diseño de cada elemento de la página en el código HTML: fuente, color del texto y fondo. Los diseñadores web utiliza CSS para definir todos estos elementos de diseño [29].

Por ejemplo, para establecer que todos los encabezados tengan el formato de fuente Calibri de 22 píxeles, debes definirlo en la hoja de estilos, que proporciona directrices sobre el aspecto de tu página web. Estas reglas se encuentran normalmente en un archivo separado al que enlazan las páginas HTML, lo que reduce en gran medida la carga de información por página.

A pesar de su gran utilidad, CSS tiene ciertos límites, que se hacen especialmente evidentes cuando se ha trabajado con este lenguaje durante años. Su principal

ventaja es también uno de sus inconvenientes: el lenguaje es muy simple. SASS lo hace un poco más sofisticado y facilita enormemente el trabajo de creación del diseño [30].

Todos los navegadores modernos pueden mostrar fácilmente CSS, pero esto no se aplica a SASS (a pesar de usarse desde hace más de diez años) u otros lenguajes avanzados de hojas de estilo. Por lo tanto, SASS generalmente se considera como un preprocesador. El código en el documento SASS se convierte primero a CSS antes de que el sistema entregue el código fuente y por lo tanto la página web.

En un primer momento, puede parecer que SASS no es necesario para diseñar una buena página web. Al fin y al cabo, con CSS ya se pueden definir los puntos básicos para poner una web en marcha. Además, aunque quieras trabajar con SASS de ahora en adelante, no puedes olvidarte de CSS. El conocido lenguaje de hojas de estilo seguirá siendo la base de la presentación de las páginas web en el futuro: SACon SASS, sin embargo, tienes algunas opciones que CSS por sí solo no te puede ofrecer [31]:

- Variables: con SASS, la información puede almacenarse en variables para su posterior reutilización. Por ejemplo, es posible almacenar un valor de color de forma centralizada bajo una variable más práctica.
- Funciones matemáticas: también puedes utilizar operadores matemáticos como +, -, *, / o %. Esto te permite influir en las especificaciones de tamaño, por ejemplo.
- Funciones: otras funciones también facilitan el trabajo en el diseño. Se pueden utilizar, por ejemplo, para ajustar valores de color o analizar listas.
- Bucles: otra ventaja de SASS es que puedes utilizar bucles que se repiten hasta que se alcanza un estatus definido.
- Distinción de casos: las sentencias condicionales “si” (if) y “de lo contrario” (else) funcionan de manera similar. Los comandos especificados solo se ejecutan si se producen ciertos estados.

- Mixins: las llamadas “mixins” son plantillas. Puedes crearlas tú mismo o simplemente integrarlas en tu propio código cuando utilices un marco de trabajo.
- Huecos: el SASS original (a diferencia del SCSS) utiliza sangrías y saltos de línea para estructurar el código. No necesita paréntesis para mostrar los anidamientos, ni punto y coma para marcar los extremos de las líneas.
- Anidación de selectores: CSS no permite anidar el código. Con SASS, sin embargo, el usuario puede visualizar las dependencias, disminuyendo la redundancia y el esfuerzo de escritura de código.
- Sucesiones: es posible traspasar propiedades de un selector a otro. Esto también ahorra la escritura innecesaria y reduce las líneas de código.
- Archivos en partes: para integrar elementos de código en un archivo SASS.SS y otros lenguajes de estilo necesitan a CSS.

Sass



Figura 2.10 - Fuente: <https://devcode.la/tutoriales/funciones-sass/>

2.2.5.4- JavaScript.

JavaScript es un lenguaje con muchas posibilidades, utilizado para crear pequeños programas que luego son insertados en una página web y en programas más grandes, orientados a objetos mucho más complejos. Con JavaScript podemos crear diferentes efectos e interactuar con nuestros usuarios [32].

Este lenguaje posee varias características, entre ellas podemos mencionar que es un lenguaje basado en acciones que posee menos restricciones. Además, es un lenguaje que utiliza Windows y sistemas X-Windows, gran parte de la programación en este lenguaje está centrada en describir objetos, escribir funciones que respondan a movimientos del mouse, aperturas, utilización de teclas, cargas de páginas entre otros [33].

Es necesario resaltar que hay dos tipos de JavaScript: por un lado, está el que se ejecuta en el cliente, este es el JavaScript propiamente dicho, aunque técnicamente se denomina Navigator JavaScript. Pero también existe un JavaScript que se ejecuta en el servidor, es más reciente y se denomina LiveWire JavaScript [34].

JavaScript



Figura 2.11 - Fuente:

<https://www.drouiz.com/blog/2016/05/22/javascript-y-jquery/>

2.2.5.5- Typescript.

TypeScript es un lenguaje de programación de código abierto desarrollado por Microsoft, el cual cuenta con herramientas de programación orientada a objetos, muy favorable si se tienen proyectos grandes. Anders Hejlsberg, arquitecto principal del desarrollo del lenguaje de programación C#, es el principal participante en el desarrollo de este lenguaje [35].

TypeScript convierte su código en JavaScript común. Es llamado también Superset de JavaScript, lo que significa que, si el navegador está basado en Javascript, este nunca llegará a saber que el código original fue realizado con TypeScript y ejecutará el JavaScript como lenguaje original [36].

TypeScript



Figura 2.12 - Fuente: <https://www.npmjs.com/~typescript-bot>

2.2.6- Librerías.

En programación, una librería es un archivo o conjunto de archivos que se utilizan para facilitar la programación. Las librerías, también llamadas "frameworks", consisten en archivos de código a los que llamamos al principio de la página, por ejemplo, una librería JavaScript será un archivo en JavaScript que insertamos al principio de la página [37].

Podemos insertar cualquier archivo al principio de la página, sin embargo, no todos los archivos pueden considerarse librerías. Para que un archivo sea una librería, este puede tener que usarse en distintas páginas, no sólo del mismo sitio, sino de varios sitios. Además, debe modificar el código de tal manera que nos facilite la inserción de nuevo código en la página [38].

Un ejemplo de librería sería la inclusión de un archivo reset.css que nos resetea la página para que ésta se vea igual en todos los navegadores. El archivo vale para páginas de distintos sitios, y nos asegura que el estilo de la página se verá igual en todos los navegadores [39].

Librerías



Figura 2.13 - Fuente: <https://www.100cia.site/index.php/programacion-y-computacion/item/17603-conoce-las-librerias-hash-para-programadores-de-c>

2.2.6.1- JQuery.

JQuery es una librería perteneciente al lenguaje de programación JavaScript, uno de los más usados en materia de desarrollo web, de código abierto y con la capacidad de mejorar la interactividad de una página web sin tener nociones de programación o un profundo conocimiento de este lenguaje. Es algo que facilita enormemente la tarea de desarrollo y diseño de páginas web [40].

Muchos de los plugins que encontramos para CMS como WordPress, tanto aquellos que son de pago como los gratuitos, han sido contruidos utilizando esta popular librería. Por lo general, es algo que está ligado a la solución de necesidades puntuales (típica tarea de un plugin), como puede ser la creación de un sistema de galería de fotos o la estructura de un menú que se adapte a pantallas móviles.

Debido a los componentes que tiene, la librería jQuery es un elemento indispensable en cualquier página web. Sin ella, es imposible que los plugins que se descarguen e instalen en el servidor puedan funcionar, al carecer de los recursos para realizar determinadas tareas. Por lo general, la descarga de esta se puede hacer a través de su propia web, y su instalación ha de efectuarse en una carpeta denominada “js” [41].

Un elemento con el que tanto programadores como diseñadores web están más que familiarizados. Gracias a jQuery se puede mejorar exponencialmente la experiencia de un usuario en una web, por lo que familiarizarse con ella y, sobre todo, saber aprovecharla, es esencial.

Como ya hemos mencionado, jQuery sirve para dar una capa extra de usabilidad a una web y mejorar el grado de interacción con el usuario. Es una librería indispensable para el desarrollo y la implementación de plugins en cualquier página web, por lo que goza de una doble utilidad que la vuelve indispensable en el entorno digital.

Funciona para cualquier compañía que quiera mejorar su web implementando cosas como galerías, sliders y demás elementos interactivos. Si se usa bien, es una buena forma de hacer que el usuario se sienta más cómodo en la web y permanezca más tiempo en ella, como también para que otros vengan y se queden por la misma razón.

Ayuda a mejorar una web haciendo que los visitantes lo tengan más fácil para navegar por ella y puedan hacer más cosas [42].

JQuery



Figura 2.14 - Fuente: <https://programa.fandom.com/es/wiki/Jquery>

Conclusión.

Mediante la información obtenida con anterioridad nos percatamos de todos los detalles y/o requerimientos que se necesitan para obtener un óptimo trabajo en el desarrollo de aplicaciones webs, debido a que todos estos puntos mencionados con anterioridad se pueden llegar a relacionar en un solo proyecto con la finalidad de así obtener mejores resultados.

CAPÍTULO III

Diseño y Desarrollo

3.1- Desarrollo.

Para la creación de una página web es necesario utilizar una herramienta de desarrollo la cual nos permita solucionar las necesidades que se generen durante el transcurso de la codificación.

Por lo cual surge la idea de realizar una página web usando angular como framework, debido a que esto nos permite una mejor estructura de código además de poder implementar más módulos para así en un futuro poder expandir más dicho proyecto.

Por motivos de privacidad, los ejemplos mostrados a continuación son similares al proyecto original, sin embargo es la misma estructura de trabajo ,ya que la empresa no permite bajo ningún motivo utilizar el material, código o trabajo generado bajo su dirección.

3.1.1- Herramientas de desarrollo.

Angular tiene un gran número de librerías que ayudan a la facilitación de codificar procesos, el lenguaje es TypeScript el cual es un superset de JavaScript, de esta manera poder crear una página web dinámica e interactiva haciéndola mucho más atractiva para el lado del cliente.

Con la implementación de TypeScript, HTML, CSS se van a realizar toda la parte visual del lado del cliente y servicios Ajax para poder hacer peticiones a los procesos realizados en el servidor estructurado con Node js y las librerías correspondientes para el óptimo funcionamiento.

Una de las ventajas y razones por las cuales fue angular el framework para realizar el frontend es por su amplio ecosistema de librerías, las cuales permiten el desarrollo eficaz y ahorro de tiempo del proyecto.

A continuación, se muestra un trabajo realizado en HTML y en angular para que se pueda notar la diferencia entre un framework y otro.

3.1.2 Preparación de HTML

Es necesario la creación del fichero "index.html" para la página principal (home page), La carpeta "CSS" Para las hojas de estilo, la carpeta "imágenes" Para las imágenes y la carpeta "js" para los ficheros de código JavaScript. Y, por último, las carpetas "pagina1" y "página 2" equivaldrían a las carpetas destinadas a organizar los diferentes ficheros de las diferentes páginas.

Nombre	Fecha de modificación	Tipo	Tamaño
css	22/01/2020 16:46	Carpeta de archivos	
font	20/01/2020 18:42	Carpeta de archivos	
img	22/01/2020 16:51	Carpeta de archivos	
contactos.html	22/01/2020 16:42	Chrome HTML Do...	4 KB
index.html	22/01/2020 16:26	Chrome HTML Do...	4 KB
integrantes.html	22/01/2020 16:56	Chrome HTML Do...	4 KB
tienda.html	22/01/2020 16:26	Chrome HTML Do...	4 KB

Figura 3.1 - Fuente: Autor

3.1.2.1 Creación de un nuevo proyecto

Para desarrollar una página web en HTML es necesario crear un documento HTML. Básicamente un documento HTML es un archivo de texto que tienen la extensión .html o .htm, en este documento se escribe todo el texto y las etiquetas HTML necesarias para la creación de una página.

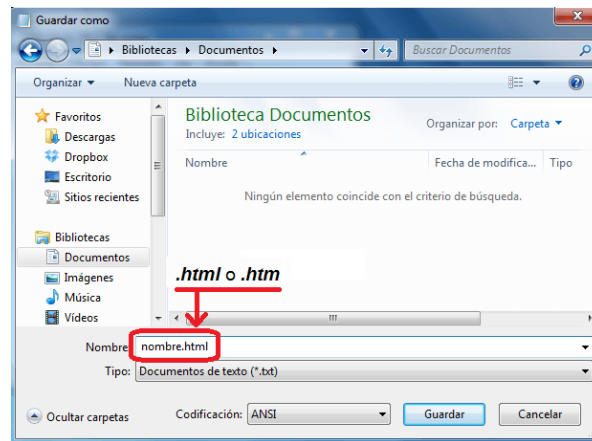


Figura 3.2 - Fuente: Autor

Una vez hecho esto tendremos un archivo HTML, este archivo por su extensión .HTML ya es una página web, notarás que el icono de este archivo es el icono del navegador de Internet que utilizas por default y se abre al hacer doble clic en él con el navegador como una página web cualquiera. Si abrimos el archivo HTML que acabamos de crear con el navegador se abrirá una página completamente en blanco. Hasta el momento tenemos una página web vacía, así que para visualizar algo en el navegador al momento de abrirla tendremos que agregar algo de texto.

Para editarlo tendremos que hacer clic derecho y la opción "abrir con" para elegir el bloc de notas o cualquier otro editor de textos y de esta manera poder editar el código fuente de nuestra página.

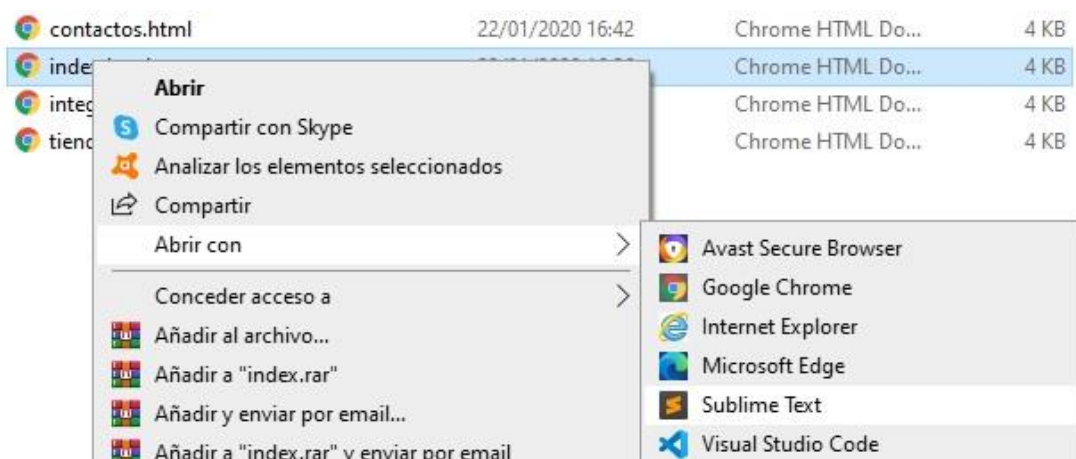


Figura 3.2 - Fuente: Autor

Esta es la estructura básica mínima de un documento HTML, la primera línea indica el tipo de documento, La segunda línea (<html>) indica el inicio del documento HTML, esta etiqueta tiene su respectiva etiqueta de cierre (</html>) al final de nuestro documento.

Después sigue la etiqueta <head> que al igual tiene su etiqueta de cierre </head>, estas etiquetas indican la cabecera del documento HTML.

Dentro del HEAD encontramos la etiqueta <title> con su etiqueta de cierre </title> dentro de estas etiquetas se escribirá el título de nuestra página. Después pasamos al cuerpo del documento que está delimitado por las etiquetas <body> y </body> dentro del cuerpo del documento es donde vamos a escribir todo aquello que se visualizará en el navegador al abrir la página, en este caso lo que se visualiza es el "Texto de prueba".

```
1  <!DOCTYPE html>
2  <html>
3      <head>
4          <title>Un titulo</title>
5      </head>
6      <body>
7          Texto de prueba
8      </body>
9  </html>
```

Figura 3.3 - Fuente:Autor

3.2- Desarrollo de aplicación web

3.2.1- Página inicial(main)

Una vez hecho lo anterior podemos comenzar a trabajar en nuestro proyecto, dentro de nuestro apartado <head> ponemos el título(url) de nuestra página dentro de la etiqueta <title> además de que en esta sección incluimos los enlaces a los demás index como los css con los que contamos.

```
<!DOCTYPE html>
<html lang="es">
<head>
    <title>www.heartagram.com</title>
    <link rel="icon" type="image/jpg" href="C:\Users\luisv\Desktop\Web\Web-Thom\Website2\img\him.jpg">
    <meta charset="utf-8">
    <meta name="viewport" content="width=device-width, user-scalable=no, initial-scale=1, maximum-scale=1, minimum-scale=1">
    <link rel="stylesheet" href="css/fontello.css">
    <link rel="stylesheet" type="text/css" href="css/stylos.css">
    <link rel="stylesheet" type="text/css" href="css/menu.css">
    <link rel="stylesheet" type="text/css" href="css/banner.css">
    <link rel="stylesheet" type="text/css" href="css/tienda.css">
    <link rel="stylesheet" type="text/css" href="css/integrantes.css">
</head>
```

Figura 3.4 - Fuente:Autor

Dentro de nuestro apartado <header> ponemos lo que será nuestro menú de nuestra página, como también los links a los cuales se le da la funcionalidad a los botones del menú, para el diseño y tipografías que se utilizaron se uso de css que se crearon.

```
<header>
  <div class="contenedor">
    <h1 class="icon-him">HIM</h1>
    <input type="checkbox" id="menu-bar" onclick="desplegar()">
    <label class="icon-menu" for="menu-bar"></label>
    <nav class="menu" id="menu">
      <a href="#">Inicio</a>
      <a href="integrantes.html">Integrantes</a>
      <a href="tienda.html">Tienda</a>
      <a href="contactos.html">Fechas</a>
    </nav>
  </div>
</header>
```

Figura 3.5 - Fuente:Auto

En este apartado está la etiqueta <main> aquí se muestra lo que conforma nuestro banner(imágenes,mensajes,botones) como la parte central de nuestra página, para el diseño y tipografías que se utilizaron se uso de css que se crearon.

```
<main>
  <section id="banner">
    
    <div class="contenedor">
      <h2>HIM</h2>
      <p>¿You Miss Me?</p>
      <a href="#">Leer mas</a>
    </div>
  </section>
  <section id="bienvenido">
    <h2>WELCOME</h2>
    <p>Bienvenidos a la pagina oficial de HIM.</p>
  </section>
  <section id="tienda">
    <h3>Lo ultimo de articulos de nuestra tienda</h3>
    <div class="contenedor">
      <article>
        
        <h4>Gran variedad en camisetass</h4>
      </article>
      <article>
        
        <h4>Nuevos diseños en chamarras</h4>
      </article>
      <article>
        
        <h4>Disponible toda la discografia</h4>
      </article>
    </div>
  </section>
```

Figura 3.6 - Fuente: Autor

La parte de <section> es la parte final de nuestra página donde mostramos la información, para el diseño y tipografías que se utilizaron se uso de css que se crearon.

```
<section id="integrante">
  <h3>Conoce mas acerca de cada uno de los integrantes</h3>
  <div class="contenedor">

    <div class="artista">
      
      <h4>Ville Valo</h4>
    </div>

    <div class="artista">
      
      <h4>Janne Puurtinen</h4>
    </div>

    <div class="artista">
      
      <h4>Mikkoo Lindström</h4>
    </div>

    <div class="artista">
      
      <h4>Juhana Rantala</h4>
    </div>

    <div class="artista">
      
      <h4>Juippi</h4>
    </div>

  </div>
</section>
</main>
```

Figura 3.7 - Fuente:Autor

Para poder realizar todo lo anteriormente mencionado se realizó el uso de CSS para darle una mejor apariencia y un mejor manejo a nuestra página web.

```
#banner{
  margin-top: 40px;
  position: relative;
}

#banner img{
  width: 100%;
  height: 100%;
}

#banner .contenedor{
  position: absolute;
  top: 50%;
  left: 50%;
  transform: translateX(-50%) translateY(-50%);
  color: #ffcc00;
}

#banner h2{
  font-size: 28px;
}

#banner a{
  display: block;
  width: 100px;
  color: #ffcc00;
  text-decoration: none;
  padding: 2px;
  margin-top: 10px;
  border: 3px solid #ffcc00;
}

#banner a:hover{
  background: rgba(166,166,166,0.3);
}
```

Figura 3.7 - Fuente:Autor

3.2.2 Preparación de Angular

Es necesario instalar angular cli para poder implementar Angular en el proyecto que se necesita. A continuación, una imagen donde muestra y explica los pasos a seguir para poder utilizar Angular.

Paso 1. Instalar la última versión de NodeJS, para eso entraremos a su web oficial y descargaremos el instalador más actualizado <https://nodejs.org/es/>

Paso 2. Desinstalar los paquetes anteriores de Angular CLI

```
npm uninstall -g angular-cli  
npm uninstall -g @angular/cli
```

Paso 3. Borrar la caché del gestor de paquetes npm

```
npm cache verify  
npm cache clean --force
```

Paso 4. Instalar la última versión de Angular CLI

```
npm install -g @angular/cli@latest
```

Figura 3.8 - Fuente: <https://victorroblesweb.es/2018/05/23/instalar-angular-6-con-angular-cli/>

3.2.2.1 Creación de un nuevo proyecto

Para la creación de un nuevo proyecto en Angular, se ejecuta el comando “ng new”, seguido del nombre del proyecto.

```
C:\Users\luisv\Desktop\capa>ng new HiM
? Would you like to add Angular routing? Yes
? Which stylesheet format would you like to use? CSS
CREATE HiM/angular.json (3542 bytes)
CREATE HiM/package.json (1247 bytes)
CREATE HiM/README.md (1021 bytes)
CREATE HiM/tsconfig.json (458 bytes)
CREATE HiM/tslint.json (3185 bytes)
CREATE HiM/.editorconfig (274 bytes)
CREATE HiM/.gitignore (631 bytes)
CREATE HiM/.browserslistrc (853 bytes)
CREATE HiM/karma.conf.js (1015 bytes)
CREATE HiM/tsconfig.app.json (287 bytes)
CREATE HiM/tsconfig.spec.json (333 bytes)
CREATE HiM/src/favicon.ico (948 bytes)
CREATE HiM/src/index.html (289 bytes)
CREATE HiM/src/main.ts (372 bytes)
CREATE HiM/src/polyfills.ts (2835 bytes)
CREATE HiM/src/styles.css (80 bytes)
CREATE HiM/src/test.ts (753 bytes)
CREATE HiM/src/assets/.gitkeep (0 bytes)
CREATE HiM/src/environments/environment.prod.ts (51 bytes)
CREATE HiM/src/environments/environment.ts (662 bytes)
CREATE HiM/src/app/app-routing.module.ts (245 bytes)
CREATE HiM/src/app/app.module.ts (393 bytes)
CREATE HiM/src/app/app.component.html (25757 bytes)
CREATE HiM/src/app/app.component.spec.ts (1048 bytes)
CREATE HiM/src/app/app.component.ts (207 bytes)
CREATE HiM/src/app/app.component.css (0 bytes)
CREATE HiM/e2e/protractor.conf.js (869 bytes)
CREATE HiM/e2e/tsconfig.json (294 bytes)
CREATE HiM/e2e/src/app.e2e-spec.ts (636 bytes)
CREATE HiM/e2e/src/app.po.ts (301 bytes)
✓ Packages installed successfully.
```

Figura 3.9 - Fuente: Autor

Una vez realizado el paso anterior se crea una carpeta con el nombre y adentro cuenta con archivos necesarios para su funcionamiento.

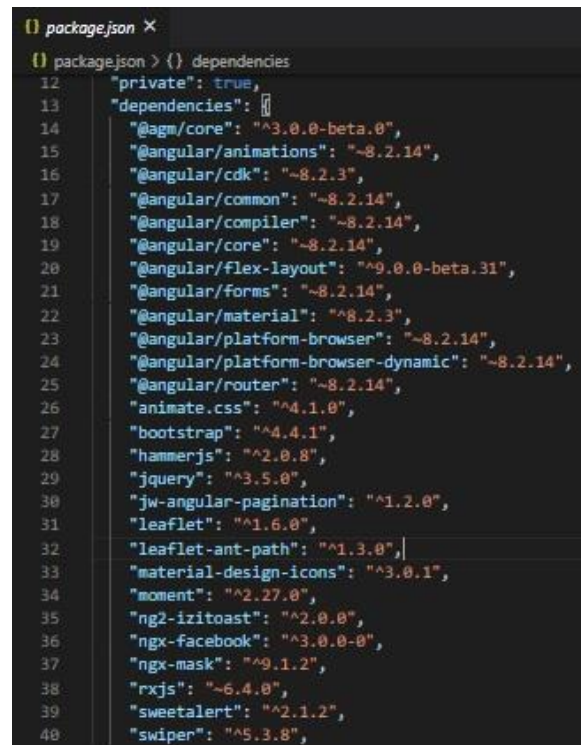
Nombre	Fecha de modificación	Tipo	Tamaño
e2e	09/10/2020 19:00	Carpeta de archivos	
node_modules	12/10/2020 12:55	Carpeta de archivos	
src	09/10/2020 19:00	Carpeta de archivos	
.editorconfig	09/10/2020 19:00	Archivo EDITORC...	1 KB
.gitattributes	09/10/2020 19:00	Documento de te...	1 KB
.gitignore	09/10/2020 19:00	Documento de te...	1 KB
angular.json	09/10/2020 19:00	Archivo JSON	6 KB
browserslist	09/10/2020 19:00	Archivo	1 KB
debug.log	09/10/2020 19:00	Documento de te...	1 KB
karma.conf.js	09/10/2020 19:00	Archivo JavaScript	2 KB
package.json	09/10/2020 19:00	Archivo JSON	2 KB
package-lock.json	09/10/2020 19:00	Archivo JSON	483 KB
README.md	09/10/2020 19:00	Archivo MD	1 KB
tsconfig.app.json	09/10/2020 19:00	Archivo JSON	1 KB
tsconfig.json	09/10/2020 19:00	Archivo JSON	1 KB
tsconfig.spec.json	09/10/2020 19:00	Archivo JSON	1 KB
tslint.json	09/10/2020 19:00	Archivo JSON	2 KB

Figura 3.10 - Fuente: Autor

Para la creación de la página web no fue necesario utilizar cada uno de los archivos que vienen por defecto a la hora de crear un proyecto. Solamente la carpeta “src” y el archivo package.json.

3.2.2.2 Declaración de librerías

Utilizando el entorno de desarrollo nos dirigimos al archivo package.json para poder agregar las librerías que van a ser utilizadas y empezar el proyecto.



```
package.json X
package.json > {} dependencies
12 "private": true,
13 "dependencies": {
14   "@agm/core": "^3.0.0-beta.0",
15   "@angular/animations": "~8.2.14",
16   "@angular/cdk": "~8.2.3",
17   "@angular/common": "~8.2.14",
18   "@angular/compiler": "~8.2.14",
19   "@angular/core": "~8.2.14",
20   "@angular/flex-layout": "^9.0.0-beta.31",
21   "@angular/forms": "~8.2.14",
22   "@angular/material": "^8.2.3",
23   "@angular/platform-browser": "~8.2.14",
24   "@angular/platform-browser-dynamic": "~8.2.14",
25   "@angular/router": "~8.2.14",
26   "animate.css": "^4.1.0",
27   "bootstrap": "^4.4.1",
28   "hammerjs": "^2.0.8",
29   "jquery": "^3.5.0",
30   "jw-angular-pagination": "^1.2.0",
31   "leaflet": "^1.6.0",
32   "leaflet-ant-path": "^1.3.0",
33   "material-design-icons": "^3.0.1",
34   "moment": "^2.27.0",
35   "ng2-izitoast": "^2.0.0",
36   "ngx-facebook": "^3.0.0-0",
37   "ngx-mask": "^9.1.2",
38   "rxjs": "~6.4.0",
39   "sweetalert": "^2.1.2",
40   "swiper": "^5.3.8",
```

Figura 3.11 - Fuente: Autor

3.3 Desarrollo de aplicación web

La estructura de un proyecto en Angular consiste en la creación de componentes, cada uno se puede crear de manera manual o generarlos con la ayuda de angular cli aunque lo más recomendable es generarlos mediante la consola para poder ahorrar tiempo.



Figura 3.12 - Fuente: Autor

Funciones de cada archivo.

- **Main.component.css:** Archivo donde se pueden crear los estilos que pueda necesitar algún elemento de la vista.
- **Main.component.html:** Archivo que se encarga de la vista de la página.
- **Main.component.spec.ts:** archivos de especificaciones son pruebas unitarias para sus archivos fuente. La convención para aplicaciones angulares es tener un archivo. spec.ts para cada archivo .ts. Se ejecutan utilizando el marco de prueba jazmín de Jasmine a través del corredor de prueba Karma.
- **Main.component.ts:** Archivo de TypeScript donde vamos a llamar a las funciones creadas en el archivo de Services.

3.3.1- Página inicial(main)

Para diseñar la página se debe de tomar en cuenta que funciones va a tener y así darnos la idea de cómo poder realizar un diseño adecuado.

Para lo anterior se debe de modificar el archivo menu.component.html creando una barra de navegación con ayuda de bootstrap. Una de las ventajas de bootstrap es la enorme cantidad de diseños que se pueden usar en el proyecto con solo copiarlo además dicho menú es responsivo permitiendo que en cualquier dispositivo que se use el menú se va a adaptar para detalles como tipografía o cosas mínimas se utiliza un css.

```
<!-- MENU -->
<header id="header">
  <!-- LOGO Y LOGIN -->
  <div id="top-menu">
    <div class="container">
      <div>
        <h1 role="banner" class="logo-prin">
          <a routerLink="/" (click)="subir()" title="HIM">
            
          </a>
        </h1>
      </div>
    </div>
  </div>
  <!-- MENU OPCIONES -->
  <div class="inner">
    <!-- MENU DE ESCRITORIO -->
    <div id="desktop">
      <div class="nav list-menu dropdown">
        <ul class="nav" id="menu-header">
          <li id="fixed-logo" class="nav-item">
            <a class="nav-link" (click)="subir()" routerLink="/" routerLinkActive="link-active"></a>
          </li>
          <li class="nav-item">
            <a class="nav-link hvr-underline-from-left" (click)="subir()" routerLink="/" routerLinkActive="link-active">Inicio</a>
          </li>
          <li class="nav-item dropdown">
            <a class="nav-link hvr-underline-from-left" [matMenuTriggerFor]="<!-- menuTriggerFor -->" routerLinkActive="link-active">Integrates <i class="material-icons arrow">keyboard_a
            </a>
          </li>
          <li class="nav-item">
            <a class="nav-link hvr-underline-from-left" (click)="subir()" routerLink="" routerLinkActive="link-active">Tienda</a>
          </li>
          <li class="nav-item">
            <a class="nav-link hvr-underline-from-left" (click)="subir()" routerLink="" routerLinkActive="link-active">Fechas</a>
          </li>
        </ul>
      </div>
    </div>
  </div>
</header>
```

Figura 3.13 - Fuente: Autor

Para desarrollar el área del banner se creó un componente de nombre b.component.html en cual todo su desarrollo se hizo de manera manual y para su edición de topografía, como diseño del mismo banner y hacerlo responsivo se utilizó su propio css.

```
<section id="banner">
  <div class="banner-contacto no-responsive" style="background-image: url(assets/images/img/grupall1.jpg);">
    <div class="informacion2">HIM</div>
    <div class="informacion">¿You Miss Me?</div>
    <div>
      <button class="b">Ver Mas <i class="fas fa-arrow-right"></i>
    </div>
  </div>
</section>
```

Figura 3.13 - Fuente: Autor

Para desarrollar el área central de nuestra página se creó un nuevo componente de nombre swiper.component.html creando un swiper. para lo cual se implementó una librería que contiene angular para usar swiper en el proyecto con solo copiarlo para detalles como tipografía o cosas mínimas se utiliza un css.

```
<div class="swiper-container">
  <div class="swiper-wrapper">
    <div class="swiper-slide">Slide 1</div>
    <div class="swiper-slide">Slide 2</div>
    <div class="swiper-slide">Slide 3</div>
    <div class="swiper-slide">Slide 4</div>
    <div class="swiper-slide">Slide 5</div>
    <div class="swiper-slide">Slide 6</div>
    <div class="swiper-slide">Slide 7</div>
    <div class="swiper-slide">Slide 8</div>
    <div class="swiper-slide">Slide 9</div>
    <div class="swiper-slide">Slide 10</div>
  </div>
  <!-- Add Arrows -->
  <div class="swiper-button-next"></div>
  <div class="swiper-button-prev"></div>
</div>
```

Figura 3.14 - Fuente: Autor

Para desarrollar el área final de nuestra página se creó un nuevo componente de nombre card.component.html ,se utilizó la herramienta bootstrap para poder crear un “card group” responsivo donde se muestran las fotos de los integrantes del grupo como una leve información de cada personaje.

```
<mat-card class="example-card">
  <mat-card-header>
    <div mat-card-avatar class="example-header-image"></div>
    <mat-card-title>Shiba Inu</mat-card-title>
    <mat-card-subtitle>Dog Breed</mat-card-subtitle>
  </mat-card-header>
  
  <mat-card-content>
    <p>
      The Shiba Inu is the smallest of the six original and distinct
      spitz breeds of dog from Japan.
      A small, agile dog that copes very well with mountainous terrain,
      the Shiba Inu was originally
      bred for hunting.
    </p>
  </mat-card-content>
  <mat-card-actions>
    <button mat-button>LIKE</button>
    <button mat-button>SHARE</button>
  </mat-card-actions>
</mat-card>
```

Figura 3.14 - Fuente: Autor

4.1- Resultados.

Una vez desarrollada la aplicación web se obtuvo como resultado una notoria optimización del 80 % en cuanto al tiempo de desarrollo debido a que angular nos ahorra tiempo en la creación de código ya que nos permite una gran cantidad de reutilización tanto de código como de hojas de estilos, de este modo puedes obtener un trabajo más óptimo y limpio ya que todo el trabajo fue realizado en diferentes componentes, a diferencia de HTML que todo se realiza en un solo archivo, por lo cual el uso de angular nos ayuda a generar un proyecto mejor estructurado para un futuro crecimiento.

4.2- Discusión.

Los resultados obtenidos en este proyecto muestran que después de haber utilizado la página web, se mostraron mayores resultados en cuanto la expansión de su producto.

Por lo anterior se acepta la hipótesis planteada hipótesis, podemos observar que la implementación de Angular permite ahorrar tiempo en procesos, tener un proyecto estructurado listo para crecer del lado del cliente y completamente separado del lado del servidor.

Conclusiones.

Al realizar un proyecto para una determinada empresa es necesario investigar y decidir cuál es la mejor manera de poder completarlo, recordando que una de los factores más importantes en la creación de un proyecto es el tiempo, con el cual contamos herramientas como Angular que están en constante crecimiento permitiendo cumplir con lo requerido.

Es importante tener una amplia comunicación con el cliente para abarcar todos los puntos, saber qué es lo que quiere y necesita así mismo el tener reuniones en determinado tiempo para mostrar los avances y asegurarse de escuchar si requiere de alguna modificación, ahorrando el realizar código que al final el cliente no va a necesitar.

Bibliografía.

- [1] Ecured (s.f). “Aplicación web” [Internet]. Disponible en https://www.ecured.cu/Aplicaci%C3%B3n_web
- [2] gcp global. (n/a). ¿Qué son las aplicaciones web?. n/a, de gcfglobal Sitio web: <https://edu.gcfglobal.org/es/informatica-basica/que-son-las-aplicaciones-web/1/>
- [3] Profesor Hugo Delgado. (30/11/2018). Elementos básicos de un sitio Web y sus partes. n/a, de disenowebakus Sitio web: <https://disenowebakus.net/elementos-de-un-sitio-web.php>
- [4] Fernando Paul Lara Galicia. (03/06/2020). ¿Qué es una interfaz web?. n/a, de godaddy Sitio web: <https://mx.godaddy.com/blog/que-es-una-interfaz-web/>
- [5] Joaquín Barberá . (N/A). Que es una interfaz web. N/A, de diseñowebmurcia1 Sitio web: <https://www.xn--diseowebmurcia1-1qb.es/interfaz-web/>
- [6] Ecured (s.f). “Navegador web” [Internet]. Disponible en https://www.ecured.cu/Navegador_web
- [7] tu blog tecnológico. (N/A). Los principales navegadores web de Internet. N/A, de tublogtecnologico Sitio web: <http://tublogtecnologico.com/navegadores-web/>
- [8] neoattack. (N/A). Navegadores. N/A, de neoattack Sitio web: <https://neoattack.com/neowiki/navegadores/>
- [9] NeoAttack (s.f). “Framework” [Internet]. Disponible en <https://neoattack.com/neowiki/framework/>
- [10] Brenda Salazar. (26/01/19). Ventajas de utilizar framework. N/A, de ordaweb Sitio web: <https://www.ordaweb.com/blog/ventajas-de-utilizar-framework-1408.html>
- [11] EDGAR TEBAR. (FEB 13, 2020). Frameworks en el desarrollo web: las mejores prácticas para tu negocio online. N/A, de wearemarketing Sitio web: <https://www.wearemarketing.com/es/blog/frameworks-en-el-desarrollo-web-las-mejores-practicas-para-tu-negocio-online.html>

- [12] Gabriela Muenta. (8 de enero de 2020). ¿Para qué se utiliza un Framework en Internet?. N/A, de rockcontent Sitio web: <https://rockcontent.com/es/blog/framework/#:~:text=En%20pocas%20palabras%2C%20un%20Framework,lo%20que%20se%20quiere%20realizar.>
- [13] Chus Naharro. (N/A). Arquitectura de la información: cómo definir la estructura de una web. N/A, de publisuites Sitio web: <https://www.publisuites.com/blog/arquitectura-de-la-informacion/>
- [14] Valentina Giraldo. (25 de enero de 2017). Descubre qué son los motores de búsqueda, qué tipos existen y para qué sirve cada uno de ellos. N/A, de rockcontent Sitio web: <https://rockcontent.com/es/blog/motores-de-busqueda/>
- [15] Abner García. (Feb 16, 2019). ¿Qué es Angular?. N/A, de medium Sitio web: <https://medium.com/ngesyfirebase/qu%C3%A9-es-angular-877c99a607f4#:~:text=Pero%2C%20C2%BFen%20qu%C3%A9%20extremo%20se,La%20respuesta%20es%20el%20frontend.>
- [16] wikipedia. (25 oct 2020). ANGULAR. N/A, de wikipedia Sitio web: [https://es.wikipedia.org/wiki/Angular_\(framework\)](https://es.wikipedia.org/wiki/Angular_(framework))
- [17] Drauta (s.f). “¿Qué es Node.js y para qué sirve?” [Internet]. Disponible en <https://www.drauta.com/que-es-nodejs-y-para-que-sirve>
- [18] wikipedia. (13 jun 2020). Node.js. N/A, de wikipedia Sitio web: <https://es.wikipedia.org/wiki/Node.js>
- [19] Jean Carlos Mariños Urquiaga. (N/A). ¿Qué es Node.js?. N/A, de devcode Sitio web: <https://devcode.la/blog/que-es-nodejs/>
- [20] Concepto.de (s.f). “Lenguaje de programación” [Internet]. Disponible en <https://concepto.de/lenguaje-de-programacion/>
- [21] Wikipedia. (18 nov 2020). Lenguaje de programación. N/A, de Wikipedia Sitio web: https://es.wikipedia.org/wiki/Lenguaje_de_programaci%C3%B3n

- [22] Redator Rock Content. (20 de abril de 2019). ¿Qué es un lenguaje de programación y qué tipos existen?. N/A, de rockcontent Sitio web: <https://rockcontent.com/es/blog/que-es-un-lenguaje-de-programacion/>
- [23] MDN web blogs (2019). “HTML” [Internet]. Disponible en <https://developer.mozilla.org/es/docs/Web/HTML>
- [24] Wikipedia. (20 nov 2020). HTML. N/A, de Wikipedia Sitio web: <https://es.wikipedia.org/wiki/HTML>
- [25] MND webdocs. (N/A). HTML: Lenguaje de etiquetas de hipertexto. N/A, de developer Sitio web: [https://developer.mozilla.org/es/docs/Web/HTML#:~:text=HTML%20\(Lenguaje%20de%20marcado%20de,la%20estructura%20del%20contenido%20web.&text=%22Hipertexto%22%20se%20refiere%20a%20enlaces,aspecto%20fundamental%20de%20la%20Web.](https://developer.mozilla.org/es/docs/Web/HTML#:~:text=HTML%20(Lenguaje%20de%20marcado%20de,la%20estructura%20del%20contenido%20web.&text=%22Hipertexto%22%20se%20refiere%20a%20enlaces,aspecto%20fundamental%20de%20la%20Web.)
- [26] Neoattack (s.f). “CSS” [Internet]. Disponible en <https://neoattack.com/neowiki/css/>
- [27] MDN webdocs. (N/A). CSS. N/A, de developer Sitio web: <https://developer.mozilla.org/es/docs/Web/CSS>
- [28] Gustavo B.. (mayo 13, 2019). ¿Qué es CSS?. N/A, de ¿Qué es CSS? Sitio web: <https://www.hostinger.mx/tutoriales/que-es-css/>
- [29] Digital Guide (2018). “SASS: ¿El mejor CSS?” [Internet]. Disponible en <https://www.ionos.mx/digitalguide/paginas-web/desarrollo-web/sass/>
- [30] Wikipedia. (3 nov 2020). Sass. N/A, de Wikipedia Sitio web: <https://es.wikipedia.org/wiki/Sass>
- [31] desarrolloweb. (08 de noviembre de 2018). Qué es Sass, como usar Sass. N/A, de desarrolloweb Sitio web: <https://desarrolloweb.com/articulos/que-es-sass-usar-sass.html>

- [32] MDN webdocs. (8 ago. 2020). ¿Qué es JavaScript?. N/A, de developer Sitio web: https://developer.mozilla.org/es/docs/Learn/JavaScript/First_steps/Qu%C3%A9_es_JavaScript
- [33] Wikipedia. (13 nov 2020). JavaScript. N/A, de wikipedia Sitio web: <https://es.wikipedia.org/wiki/JavaScript>
- [34] Maestros del web (2007). “¿Qué es Javascript?” [Internet]. Disponible en <http://www.maestrosdelweb.com/que-es-javascript/>
- [35] DevCode (s.f). “¿Qué es Typescript?” [Internet]. Disponible en <https://devcode.la/blog/que-es-typescript/>
- [36] Wikipedia. (16 sep 2020). TypeScript. N/A, de Wikipedia Sitio web: <https://es.wikipedia.org/wiki/TypeScript>
- [37] AprendeWeb (s.f). “Librerías” [Internet]. Disponible en <https://aprende-web.net/librerias/>
- [38]]wikipedia. (22 oct 2020). Biblioteca (informática). N/A, de Wikipedia Sitio web: [https://es.wikipedia.org/wiki/Biblioteca_\(inform%C3%A1tica\)#:~:text=En%20inform%C3%A1tica%2C%20una%20biblioteca%20o,la%20funcionalidad%20que%20se%20invoca.](https://es.wikipedia.org/wiki/Biblioteca_(inform%C3%A1tica)#:~:text=En%20inform%C3%A1tica%2C%20una%20biblioteca%20o,la%20funcionalidad%20que%20se%20invoca.)
- [39]programacion-en-c.wixsite. (N/A). Librerías y tipo de datos. N/A, de programacion-en-c.wixsite. Sitio web: <https://programacion-en-c.wixsite.com/programacion-en-c/librerias-y-tipo-de-datos>
- [40] NeoAttack (s.f). “Jquery” [Internet]. Disponible en <https://neoattack.com/neowiki/jquery/>
- [41] Wikipedia. (25 sep 2020). jQuery. N/A, de Wikipedia Sitio web: <https://es.wikipedia.org/wiki/JQuery>
- [42] norfipc. (N/A). JQuery, ejemplos prácticos para usar en páginas web. N/A, de norfipc Sitio web: <https://norfipc.com/codigos/jquery-ejemplos-practicos-usar-paginas-web.php>

