



UNIVERSIDAD EL SUR

ALUMNO:

Mario Alberto Cabrera Gil

CARRERA:

Ingeniería en Sistemas Computacionales

DOCENTE:

Iván Ortiz

MATERIA:

Programación Web

CUATRIMESTRE:

5to cuatrimestre
Grupo "A"

TEMA:

Introducción al frontEnd

Índice

INTRODUCCIÓN	1
QUE ES FRONTEND	2
PARA QUÉ SIRVE EL FRONTEND	4
Ejemplos de frontEnd	4
LENGUAJES DE PROGRAMACIÓN EN FRONTEND	5
CONCEPTOS DE FRONTEND	9
¿Qué necesitamos para el desarrollo front-end?	10
SERVIDORES WEB	12
5 características de un servidor web	12
¿Cómo funciona un servidor web?	15
ENTORNO Y HERRAMIENTAS DE DESARROLLO DE UN FRONTEND	17
REFERENCIA	19

INTRODUCCIÓN

En este tema hablare sobre el inicio o vaya la redundancia, hablare sobre la introducción al frontend, ya que es una parte fundamental y muy importante en la programación, daré una breve explicación sobre lo que es el frontend; es parte de un sistema de información al que el usuario accede directamente e interactúa para recibir o utilizar las capacidades de back-end del sistema anfitrión.

Los sistemas front-end y back-end son los puntos inicial y final de cualquier sistema de procesamiento de software o de red. El frontend, que es el dispositivo de hardware que protege el ordenador del tráfico, se coloca en el límite exterior del enlace de red. El back end está compuesto de routers y / o servidores (por ejemplo, base de datos y web).

Esto es lo que veremos y también otras funciones para poder crear páginas web pero como dije anteriormente el frontend es lo más importante, todo esto se basa en sistema de programación.

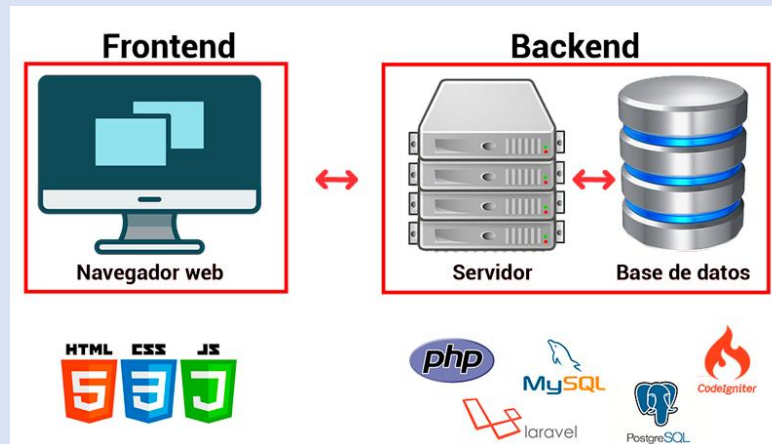
QUE ES FRONTEND

El Front end es la parte de una web que conecta e interactúa con los usuarios que la visitan. Es la parte visible, la que muestra el diseño, los contenidos y la que permite a los visitantes navegar por las diferentes páginas mientras lo deseen. Es una de las dos mitades en las que se divide la estructura de cualquier página web.

Junto a esta se encuentra el Back end, que es el polo completamente opuesto, la capa que accede a datos y software en general para su comunicación. Ambas se reúnen en cualquier site que visite una persona y son las que, trabajando, hacen que funcione en todo momento tal y como lo hace.

Esta parte es la que engloba y muestra todo el trabajo de diseño web y, por lo general, reúne en su interior hasta 3 lenguajes de programación diferentes: HTML, CSS y JavaScript. Cada uno orientado a determinados fines en concreto, se suman para conseguir el resultado final que aparece por la pantalla de cada usuario que entra en una web, sea cual sea.

Se podría decir que el Frontend es como la máscara de una página, pero en realidad va más allá de eso. Es lo que debe mostrar la identidad corporativa de una compañía dentro de Internet, la parte que puede atraer o alejar al consumidor cuando este accede a la página. De ahí que sea tan importante y requiera de tanto esfuerzo.



PARA QUÉ SIRVE EL FRONTEND

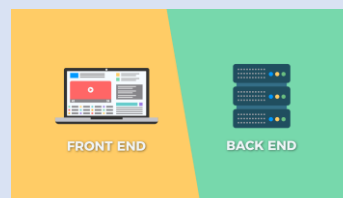
El Front end sirve para que un usuario pueda entrar a una web cualquiera, ya que es la parte que este ve y con la que este interactúa al acceder. Debe estar diseñado para ser fácilmente utilizable y también muy atractivo. De hecho, de su usabilidad y su estética depende la imagen que se transmita de la firma en cuestión en el plano digital.

Si consigue una buena experiencia de usuario, puede hacer que este se quede más tiempo e incluso que traiga a otros, algo que da más visibilidad a cualquier negocio y, por supuesto, ayuda en la generación de leads.

Ejemplos de frontEnd

Ejemplos de Frontend podemos encontrar tantos como páginas web haya abiertas actualmente en todo Internet. Para dar un caso concreto, vamos a tomar como referencia la web de NeoAttack, nuestra agencia de marketing digital.

En el momento que entras en ella ves una imagen, una serie de categorías y varios contenidos que se van abriendo a medida que accedes a ellos. Todo eso que encuentras ahí es el backend, todo el entramado de secciones por el que te mueves.



LENGUAJES DE PROGRAMACIÓN EN FRONTEND

El desarrollo de frontend es uno de los aspectos más importantes de la creación de aplicaciones fáciles de usar para aplicaciones móviles y web.

Actualmente, existen muchas opciones cuando se trata de lenguajes para el desarrollo frontend, y cada una tiene sus ventajas y desventajas.

La elección del lenguaje impacta directamente en el proceso de desarrollo y en la experiencia que obtienen los usuarios, pero elegir uno adecuado puede ser un desafío.

En este artículo, puede encontrar información útil sobre los 10 principales lenguajes de programación frontend de la actualidad, incluidos sus pros y contras.

1. HTML

HTML, una abreviatura de Hypertext Markup Language, es un lenguaje de programación para crear documentos electrónicos denominados páginas que aparecen en la web. Cada página tiene múltiples conexiones a hipervínculos o enlaces a otras páginas.

2. CSS

CSS es un lenguaje de programación que se utiliza para indicar cómo se ofrecen los documentos a los usuarios en términos de diseño y estilo. Un documento generalmente se refiere a una estructura de archivo de texto que utiliza un lenguaje de marcado como HTML, que es ampliamente utilizado y otros como XML o SVG.

3. JavaScript

JavaScript es un lenguaje de programación de uso popular que permite a los usuarios implementar características sofisticadas en páginas web.

4. React

React es un lenguaje de programación desarrollado por Facebook para la creación de interfaces de usuario web rápidas y de vanguardia. Actualmente es una de las bibliotecas de JavaScript más utilizadas para la creación de interfaces web.

5. Vue

Vue es un marco para la creación de interfaces de usuario y es diferente de la mayoría de los marcos monolíticos, ya que ha sido creado para ofrecer una adaptabilidad óptima.

6. TypeScript

TypeScript es otro de los 10 principales lenguajes de programación frontend y un supe conjunto sintáctico de JavaScript. Cualquier código JavaScript se considera un código TypeScript válido, que se ejecuta en los navegadores como JS.

7. Elm

Elm es un lenguaje funcional y un marco orientado al desarrollo frontend. Es la fuente del lenguaje de programación Haskell y mejora la capacidad de mantenimiento de JavaScript.

8. Angular

Angular es un marco de código abierto que se utiliza principalmente para el desarrollo de aplicaciones web de una sola página. Los desarrolladores pueden confiar en él para generar menús animados adecuados para páginas web HTML.

9. JQuery

JQuery es una biblioteca de JavaScript objetiva pero repleta de características. Es una opción viable para características como manejo de eventos, recorrido HTML, Ajax, manipulación de documentos y animaciones.

10. Swift

Swift, desarrollado por Apple en 2014, es una tecnología de desarrollo de frontend adecuada para crear aplicaciones iOS, tvOS, macOS y watchOS.



CONCEPTOS DE FRONTEND

Ingresar al mundo de la programación web puede ser difícil para las personas nuevas. Tantos conceptos suenan extraños y se vuelven problemáticos en la etapa de escritura. ¿Cómo escribir: desarrollador de software o desarrollador front-end? ¿Quizás un desarrollador front-end, o más bien un ingeniero de software front-end? Sin embargo, la más común es la entrada con guión. Desarrollador front-end: la persona responsable de la parte visual de la aplicación (lo que vemos, nosotros = usuarios del sitio).

Por lo general, el desarrollador front-end trabaja con un diseñador gráfico (diseñador web) que proporciona el diseño gráfico. A menudo, los desarrolladores front-end tienen la oportunidad de trabajar con un diseñador de UX que propone y crea un prototipo de interacciones que tendrán lugar en los sitios web. El papel de los desarrolladores front-end es ponerlo todo junto. Se dice popularmente que un desarrollador front-end tiene la tarea de cortar un diseño gráfico, dividir el diseño en partes más pequeñas y codificar (HTML y CSS) en sitios web que los usuarios finalmente usan.

¿Qué necesitamos para el desarrollo front-end?

Aquí está la receta para el éxito. Para crear un sitio web, necesita:

HTML + CSS

Estos son los ingredientes básicos para construir un sitio web. HTML proporciona la estructura de los sitios web: los elementos descritos, y CSS le dice al navegador cómo deben verse estos elementos. Una vez que haya dominado HTML y CSS, puede recrear cualquier diseño (gráfico) y convertirlo en un sitio web.

JavaScript

Un lenguaje de desarrollo front-end de programación web para crear sitios web. Gracias a JavaScript, agregas interactividad a tu sitio web, es decir, desarrollas sitios web que responden a las acciones de los usuarios. Incluso puedes construir juegos. Recomendamos dominar al menos el código JavaScript básico para entender cómo funciona el lenguaje o la lógica del lenguaje de programación web. Luego puedes ir a jQuery.

JQuery

Técnicamente, no es un lenguaje sino una biblioteca de JavaScript. Como un intermediario que te permite escribir JavaScript más fácil y rápido. Ingresas el código jQuery; la biblioteca se preocupa por ti y juega con el traductor para pasar el significado del JavaScript ligeramente complicado al navegador. Aunque hoy en día jQuery es "un poco viejo", recomendamos aprenderlo "a pedido", ya que facilita el aprendizaje de la lógica de programación web y Bootstrap (framework CSS utilizado) y todavía aparece en las ofertas de trabajo.

SERVIDORES WEB

Un servidor web es una plataforma computacional muy potente que resguarda datos para ser consultados por diversos usuarios. Cuenta con un software que entrega la información solicitada por visitantes o miembros de un grupo de trabajo, a través de un dispositivo conectado a la red.

Un servidor web algunas puede ser definido como un hardware y como un software; porque efectivamente posee cualidades de ambos. Es una computadora con una alta potencia que le permite mantenerse conectada a la red las 24 horas del día, los 365 días del año; además de contar con un sistema operativo, de almacenamiento y de procesamiento sumamente eficaces para atender las consultas de los usuarios.

5 características de un servidor web

Aunque existen muchos tipos de servidores web, hay algunas características que todos comparten. Entre ellas, es importante tener en cuenta que:

1. Necesitan un soporte físico

Los servidores web deben tener un hardware, que es el equipo que resguarda la información que será transmitida al ser solicitada por los usuarios. Estos soportes físicos deben tener una amplia capacidad de almacenamiento y suelen ser potentes equipos computacionales compartidos por diversas empresas y usuarios.

2. Requieren un sistema operativo

Para que la información del hardware pueda ser extraída y direccionada a su destino final, es esencial tener un software diseñado para llevar a cabo estas tareas.

3. Cuentan con un Protocolo de Transferencia de Hipertexto

También conocido como HTTP. Son instrucciones que permiten la transmisión de información. Los HTTP establecen la comunicación entre navegadores o plataformas de búsqueda con las bases de datos de los servidores web.

4. Sirven para la búsqueda de información

Los servidores tienen como objetivo responder de manera oportuna a las solicitudes de los usuarios de un sistema. Estas herramientas gestionan la respuesta a los comandos de búsqueda de datos por parte de los clientes, por lo que su objetivo es permitir el acceso a la información que resguarda.

5. Alta capacidad

Un servidor web debe ser potente y contar con una gran capacidad de almacenamiento, ya que aloja diferentes sitios, cuentas e interacciones.

Un servidor web almacena los archivos que forman parte de un sitio electrónico o de una base de datos. El servidor funge como transmisor de esta información de acuerdo con las demandas de sus usuarios. De este modo, clientes y consumidores pueden acceder a la información solicitada vía remota.

Por lo tanto, un servidor web se mantiene siempre activo para recibir las solicitudes de los usuarios, procesarlas y entregar resultados sin excepción; incluso si no encuentra dicha consulta manda el código 404, que significa que no existen los recursos solicitados en su acervo.

Además del almacenamiento de archivos y consultas de navegación web, puede ofrecer servicios adicionales como correo electrónico, manejo de bases de datos, seguridad web, servidor proxy y de sistemas de nombres de dominio (DNS).

¿Cómo funciona un servidor web?

A grandes rasgos, su función inicia cuando un usuario realiza una consulta en los motores de búsqueda, o bien cuando escribe directamente una dirección web (URL) o un dominio. Luego, en menos de un segundo, el servidor web se encarga de localizar el contenido y enviarlo para que el usuario pueda verlo en su dispositivo.

Un servidor funciona como un lugar de almacenamiento de diversos tipos de información, la cual puede constituirse por archivos HTML, CSS, JavaScript (incluidas imágenes, textos, videos o audios), y debe ser ubicada por los programas computacionales que hacen funcionar el sistema operativo.

Si un usuario desea acceder a un sitio web mediante su computadora personal, un servidor debe poner en marcha un programa que extraiga la información de la página electrónica para hacérsela llegar al cliente. Los

comandos HTTP (que suelen anteceder a los dominios electrónicos) funcionan justamente como órdenes que permiten transferir la información de un servidor físico hacia el equipo con el que interactúa el usuario final.

De este modo, la información puede ser consultada desde cualquier lugar con acceso a internet, sin ocupar espacio de almacenamiento dentro de nuestras computadoras.

ENTORNO Y HERRAMIENTAS DE DESARROLLO DE UN FRONTEND

Si estás comenzando en el mundo del Frontend (por ejemplo para currar con JavaScript, HTML o CSS) te estarás preguntando por dónde empezar a picar código. Pues como muchas cosas en la vida, hay muchas opciones, te voy a contar las que yo recomiendo.

Lo primero que tienes que saber es que necesitas tener un PC (computadora) o un portátil (laptop). La mayoría de herramientas para programar están para ordenador, y aunque se puede, no te recomiendo programar en Tablet o Smartphone.

Lo siguiente que tienes que saber es que necesitas dos partes fundamentales: un navegador web (Firefox, chrome, edge, etc.) y un editor de código.

El navegador web probablemente ya lo tengas si estás leyendo esto, pero te recomiendo que mires si lo tienes actualizado, ya que versiones recientes de los navegadores incorporan herramientas que facilitan la vida al developer.

Las herramientas usadas por un Desarrollador Front-End son:

1. Lenguajes: HTML, CSS, JavaScript, SASS.

Los lenguajes de programación son los que utiliza el desarrollador para escribir los códigos que crean las instrucciones que le darán orden, estructura y funcionamiento a las páginas, sitios o plataformas que se estén creando.

2. Frameworks: Materialize, Bootstrap, Angular.

Un framework es un conjunto de archivos que crean un esquema, estructura y metodología que tienen como objetivo el desarrollo ágil de aplicaciones y sitios web.

3. Librerías: JQuery, React.

Las librerías de programación son un conjunto de archivos que facilitan la creación del código. Su finalidad es proporcionar códigos funcionales, soluciones a bugs y diversas funcionalidades diseñadas por otros desarrolladores previamente.

4. Software: Lighthouse.

Los desarrolladores utilizan herramientas con las que realizan pruebas de rendimiento del sitio. Así, obtienen información sobre los puntos de mejora y optimización de la plataforma.

REFERENCIA

Contenidos. (2020, 31 diciembre). ▷ ¿Qué es Front end y para qué sirve? - Neo Wiki | NeoAttack. NeoAttack.
<https://neoattack.com/neowiki/front-end/>

Presta, M. (2021, 20 agosto). Los 10 principales lenguajes de programación de Frontend. Back4App Blog.
<https://blog.back4app.com/es/los-10-principales-lenguajes-de-programacion-de-frontend/>

Bolharin, D. (2022b, agosto 9). ¿Cuáles son los fundamentos del desarrollo front-end? aprenda la guía completa. AppMaster - ultimate all-in no-code platform. <https://appmaster.io/es/blog/los-fundamentos-del-desarrollo-front-end>
<https://blog.hubspot.es/website/que-es-servidor-web>