

Investigación 10

Frontend y Backend

Alan Wiler Zarate Chino
Universidad Católica Boliviana “San Pablo”
La Paz, Bolivia
alan.zarate@ucb.edu.bo
09/10/2022

Resumen— En la siguiente investigación se define que el desarrollo frontend consiste en la conversión de datos en una interfaz gráfica del lado del usuario y el backend es el término utilizado para referirse al área lógica de toda página web, es decir la arquitectura interna del sitio que asegura un funcionamiento correcto de los elementos.

Abstract— In the following investigation, it is defined that frontend development consists of the conversion of data into a graphical interface on the user's side and the backend is the term used to refer to the logical area of any web page, that is, the internal architecture of the site that ensures correct operation of the elements.

I. INTRODUCCIÓN

Cuando se habla de programación o desarrollo web, surgen terminos como frontend y backend que confunden a la mayoría de personas que no se dedican a la programación. En el campo del desarrollo de aplicaciones web, para funcionar correctamente se necesitan técnicas para definir las acciones que realizaran. Esta diferenciación se las puede realizar del lado del cliente y del lado de los servidores, dando lugar al frontend y backend respectivamente.

II. ¿QUE ES VUE?

Vue es un framework open source de JavaScript, el cual permite construir interfaces de usuarios de una forma muy sencilla. La curva de aprendizaje es relativamente baja, un conocimiento profundo en JavaScript es de gran utilidad para el uso del framework y una experiencia con callbacks, promesas, objetos, entre otros temas más. [1]

Si queremos hacer uso de Vue no será necesario instalar absolutamente nada en la computadora, si así se desea se puede utilizar este framework mediante un CDN. [1]

Vue fue creado por Evan You ex trabajador de Google, quien, es importante mencionar, fue desarrollador Angular. Vue fue lanzado en el año 2014. Aunque inicialmente fue pensado para ser una biblioteca personal, la comunidad hizo que el proyecto creciera a un ritmo impresionante, posicionándolo

hoy en día como uno de los Frameworks web más populares, junto con Angular y React. [1]

Una de las características más importantes de Vue es el trabajo con componentes. Un componente Vue, en términos simples, es un elemento el cual se encapsula código reutilizable. Dentro de un componente podremos encontrar etiquetas HTML, estilos de CSS y código JavaScript. Los componentes nos permiten desarrollar proyectos modularizados y fáciles de escalar, si nosotros así lo deseamos podemos reemplazar un componente por otro de una forma muy sencilla, como si de piezas de lego se tratasen. [1]

Ejemplo.

Un componente simple posee 3 secciones, HTML, CSS y JavaScript

```
<template>
<h1 class="text-center"> Hola Cody </h1>
</template>

<script>
</script>

<style>
</style>
```

Usando componentes con variables y estructura más compleja.

```
<div id="app">
  <div class="books" v-for="book in books">
    <p> Titulo : {{ book.title }} </p>

    <div v-if="book.has_description">
      <p> {{ book.description }} </p>
    </div>

    <div v-else>
      <p> El libro no cuenta con una descripción! </p>
    </div>
  </div>
</div>
```

En el ejemplo se observa la iteración de libros, en cada iteración se pinta el título. Si el libro tiene descripción entonces se muestra, en caso contrario se muestra el mensaje “El libro no cuenta con una descripción”.

Otra de las principales características de Vue, es que este, es un framework reactivo, los sistemas reactivos como su nombre indica son sistemas que reaccionan al cambio, dependiendo de algún evento interno dentro del sistema es posible realizar un cambio de estado y con eso modificar el comportamiento o la vista de nuestra aplicación.

Por ejemplo, si el valor de una variable se modifica, podemos reaccionar al cambio alterando nuestro DOM, nuestra vista, esto sin la necesidad de recargar la página. En esta ocasión, a nuestra variable (contador) tomar un valor mayor a cinco, modificamos el DOM, utilizando un div diferente al inicial.

```
<body>
  <div id="app">
    <div v-if="count <= 5">
      <h3> Contador {{ count }}</h3>
    </div>
    <div v-else>
      <h3> El valor es mayor a 5!</h3>
    </div>
  </div>
</body>
```

El modelo se encuentra almacenado en un objeto llamado data. Todos los atributos que agreguemos a dicho objeto serán monitoreados ante posibles cambios. De esta forma se garantiza el sistema reactivo de Vue. También es posible agregar n cantidad de métodos necesarios.

```
var app = new Vue({
  el: '#app',
  data: {
    count: 1
  },
  methods: {
    change: function(){
      this.count += 1
      setTimeout(this.change, 1000)
      console.log(this.count)
    }
  }
})
```

El hecho de trabajar con componentes y bajo una arquitectura basada en eventos es algo muy útil para un desarrollador frontend, no solo por la facilidad de desarrollo si no por la interacción que tendrán tus usuarios. Además, con la implementación de un CDN, el desarrollo es mucho más sencillo.

Encaja perfectamente en la creación de aplicaciones con interfaces de usuario altamente adaptables y del tipo Single-Page (Página Única). [1]

Ventajas

- HTML empoderado. Esto significa que Vue.js tiene muchas similitudes con Angular y esto puede ayudar a optimizar el manejo de bloques HTML usando diferentes componentes. [2]
- Documentación detallada. Vue.js tiene una documentación muy completa, que puede acelerar la curva de aprendizaje para desarrolladores y ahorrar mucho tiempo en el desarrollo de una aplicación usando sólo conocimientos básicos de HTML y JavaScript. [2]
- Adaptabilidad. Proporciona un rápido periodo de cambio desde otros frameworks a Vue.js por sus similitudes con Angular y React en términos de diseño y arquitectura. [2]
- Excelente integración. Vue.js puede usarse para construir tantas aplicaciones de página única o complejas interfaces web de aplicaciones. Lo principal son las pequeñas partes interactivas que pueden ser integradas fácilmente en una infraestructura existente sin efectos negativos en el sistema. [2]
- Gran escalado. Vue.js puede ayudarnos a construir grandes plantillas reutilizables en poco tiempo de acuerdo con su sencilla estructura. [2]
- Pequeño tamaño. Vue.js puede ocupar cerca de 20KB manteniendo su velocidad y flexibilidad que permite alcanzar un mejor rendimiento en comparación con otros frameworks. [2]

Desventajas

- Falta de recursos. Vue.js aún tiene poca cuota de mercado comparado con Angular o React, lo que significa que los recursos disponibles de este framework aún están en su fase inicial. [2]
- Riesgo de excesiva flexibilidad. En ocasiones, Vue.js puede tener problemas para integrarse en grandes proyectos y aún no hay experiencia acerca de posibles soluciones, aunque a buen seguro estarán disponibles pronto. [2]
- Falta de documentación completa en inglés. Esto nos lleva a una complejidad en ciertas fases del desarrollo, sin embargo, de cada vez hay más material traducido al inglés. [2]

Empresas que usan Vue.js: Xiaomi, Alibaba, WizzAir, EuroNews, Grammarly, Gitlab y Laracasts, Adobe, Behance, Codeship, Reuters.

A la fecha de 2019 se tiene la siguiente oferta de empleo para los 3 frameworks más populares de frontend.

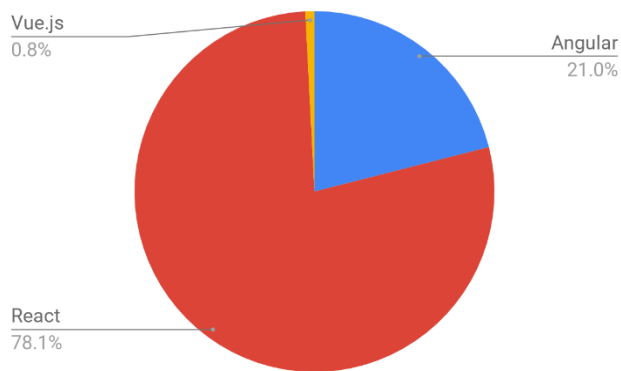


Figura 1. Gráfica de las tecnologías más demandadas en el año de 2019 de desarrollo frontend

III. FRONTEND Y BACKEND

Frontend

El frontend es la parte del desarrollo web que se dedica a la parte frontal de un sitio web, en pocas palabras del diseño de un sitio web, desde la estructura del sitio hasta los estilos como colores, fondos, tamaños hasta llegar a las animaciones y efectos. Es esa parte de la página con la que interaccionan los usuarios de esta, es todo el código que se ejecuta en el navegador de un usuario, al que se le denomina una aplicación cliente, es decir, todo lo que el visitante ve y experimenta de forma directa. [3]

Un front-end, es la persona que se dedica básicamente al diseño web, pero esto no significa que no toque código, tanto el front-end como el back-end están en contacto con código todo el tiempo. Dentro del área de front-end se trabaja con lenguajes mayormente del lado del cliente, como:

HTML (lenguaje de marcado de hipertexto, se utilizan etiquetas que estructuran y organizan el contenido de la web).

CSS (se encarga del formato y diseño visual de las páginas web escritas en html) para darle estructura y estilo al sitio.

Javascript (un lenguaje de programación, rápido y seguro para programar centros de datos, consolas, teléfonos móviles o Internet) para complementar los anteriores y darle dinamismo a los sitios web.

HTML y CSS son lenguajes de marcado y estilo. Javascript es un lenguaje de programación.

De ellos se generan una gran cantidad de frameworks y librerías que van aumentando las capacidades que se tiene para generar cualquier tipo de interfaz de usuario, como React, Redux, Angular, Bootstrap, Foundation, LESS, Sass, Stylus y PostCSS, entre otros. [3]

Principalmente se conoce como el lado del cliente.



Figura 2. Tecnologías de frontend

Backend

Mientras que el frontend es la capa de programación ejecutada en el navegador del usuario, el backend procesa la información que alimentará el frontend de datos. Es la capa de acceso a los datos, ya sea de un software o de un dispositivo en general, es la lógica tecnológica que hace que una página web funcione, lo que queda oculto a ojos del visitante. [3]

El backend de una solución, determina qué tan bien se ejecutará la aplicación y qué experiencia, positiva o negativa, obtendrá el usuario de su uso. [3]

Trabajar en este apartado supone algo totalmente diferente al frontend, ya que exige el dominio de otros términos de programación, lenguajes que requieren una lógica, ya que esta área es también la encargada de optimizar recursos, de la seguridad de un sitio y otros factores. [3]

Aquí se utilizan frameworks como PHP, Javascript, Python y Ruby, entre otros.

Un framework de desarrollo es el que trabaja en el lado del servidor, se encarga de que las personas puedan disfrutar de una grata experiencia, su propósito principal es que los procesos de desarrollos estén más acelerados, que se puedan volver a utilizar los fragmentos de códigos ya existentes y que se realicen buenas prácticas mediante el uso de distintos patrones. [3]

Con ellos se pueden escribir códigos o desarrollar aplicaciones con el mínimo esfuerzo.

Las herramientas que se utilizan son editores de código, compiladores, debuggeadores para revisar errores de seguridad y gestores de bases de datos.

El flujo de trabajo de un backend consiste en darle funciones a un sitio; mientras que el frontend hace un sitio estático, el

backend después le da funciones y adapta el sistema programado a ese sitio web.

Otras funciones que se gestionan son:

- Creación de funciones que simplifiquen el proceso de desarrollo.
- Acciones de lógica.
- Conexión con bases de datos.
- Uso de librerías del servidor web (un ejemplo puede ser para implementar temas de caché).

Además, debe velar por la seguridad de los sitios web y optimizar al máximo los recursos para que las páginas sean ligeras.

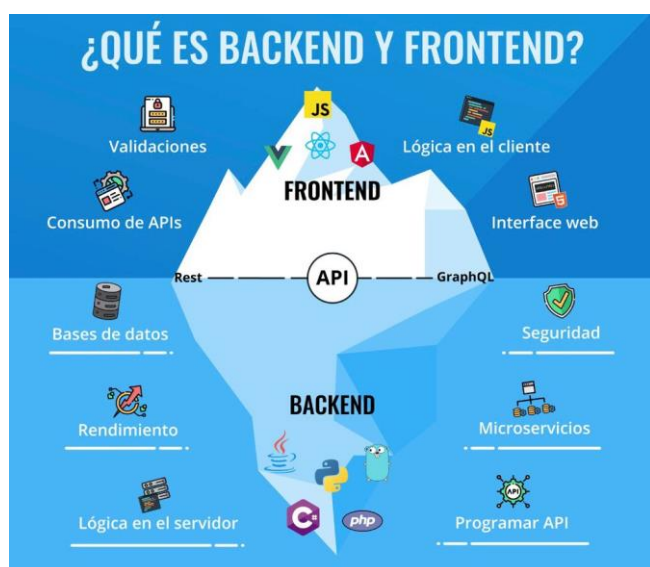


Figura 3. Representación de lo que es visible para el usuario

Funcionamiento del frontend y el backend

Considera que una web se conforma por una gran variedad de documentos que se relacionan entre ellos por medio de enlaces, lo que significa que si quieres entrar a una web y escribes la dirección URL en el navegador, se traduce como que estás solicitando que se muestre dicha página web. [4]

Lo siguiente que hará el dispositivo es verificar qué servidor de software tiene el sitio. El servidor recibe esta información, verifica la petición que hizo y te permite ejecutar la acción. [4]

En ocasiones, se puede presentar el caso de que no se requiere una conexión a la base de datos, por ejemplo, accedes a una página y cuando inicias sesión, se hace de manera automática una petición que conecta a la base de datos para verificar los accesos y la suscripción que se tiene, es allí donde el backend devuelve la respuesta al servidor. [4]

Después aparece el frontend, que es quien va a recibir la información que transmitió el backend y la va a acomodar en la interfaz del sitio web o perfil del usuario. [4]

Ventajas del frontend y el backend

Mayor escalabilidad

Se trata de un entorno mucho más escalable, al estar separado, es posible que una de las dos partes necesite más recursos en un algún momento, por lo que se hace más sencillo y eficaz dividir los recursos. [4]

Distintos equipos de desarrollo

Al tener estas dos partes que se diferencian, puedes tener en tus equipos diversos perfiles dedicados únicamente a su parte del trabajo.

Ampliación de plataformas disponibles

Imagina que tienes una web con backend y frontend separados; si requieres desarrollar una app nativa para dispositivos móviles, el desarrollo de la misma será mucho más sencillo, ya que solamente te enfocarías en desarrollar la app. La API de comunicación ya la tendrías desarrollada y quizás solamente requeriría de pequeños ajustes, además, en el proceso de desarrollo, la parte front de tu web existente, no se vería afectada en ningún momento. [4]

Las migraciones y actualizaciones son más sencillas

Normalmente, cuando ya está desarrollado el backend, lo que más se suele cambiar es la parte frontend, por esa razón, si te piden actualizaciones de diseño en la web o cambio de alojamiento, es más fácil, ya que la parte de servidor siempre va a estar funcionando por muchas actualizaciones que se tengan en la parte visual. [4]

Stack

Un stack es el conjunto de tecnologías y herramientas que se utilizan para desarrollar una aplicación, es decir, son todas las tecnologías que usa el frontend y el backend. [4]

A este término se le denomina desarrollador Fullstack, se refiere a un programador capaz de manejar todas estas herramientas, lenguajes y aspectos relacionados con la creación y el mantenimiento de una aplicación web. [4]

IV. EJEMPLO

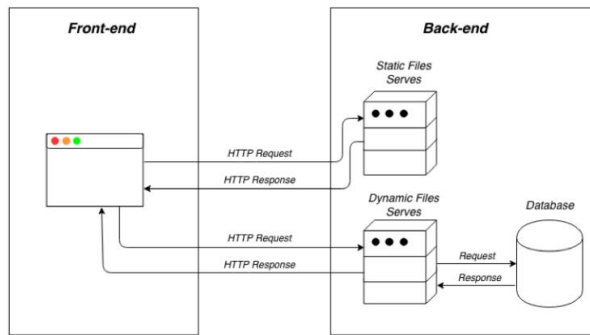


Figura 4. Diagrama del protocolo de comunicación entre el frontend y backend

Tecnologías utilizadas en un MERN stack

- MongoDB
- ExpressJS
- ReactJS
- NodeJS

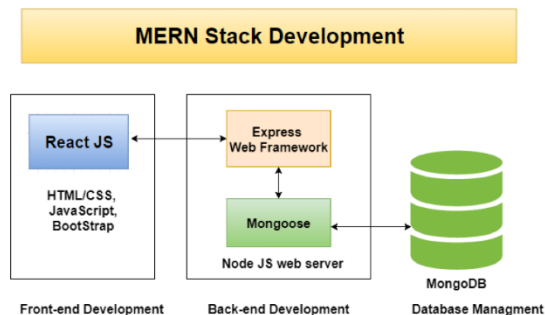


Figura 5. Diagrama de MERN stack

V. CONCLUSIÓN

El desarrollo de front-end es el proceso de hacer que un sitio web funcione y se vea bien en un navegador. Por lo general, lo hacen diseñadores y desarrolladores, a veces con aportes del personal de marketing. Los desarrolladores front-end pueden codificar en HTML, CSS y JavaScript para integrar el diseño del cliente con una aplicación, o pueden trabajar con frameworks como AngularJS, VueJS o ReactJs entre los más populares.

El desarrollo de back-end es el proceso de administrar el almacenamiento de datos y acceder a los datos de una base de datos para mostrarlos en una página web para que los usuarios puedan consumirlos en cualquier dispositivo.

Los desarrolladores de back-end a menudo deben ser fuertes en habilidades como Java, C ++, Ruby on Rails, PHP o Python, mientras que los desarrolladores de front-end pueden necesitar conocer algunos de estos lenguajes. Sin embargo, a menudo utilizan herramientas o frameworks como Bootstrap

o Tailwind, que les permiten crear sus interfaces web de forma rápida y sencilla.

VI. BIBLIOGRAFIA

- [1] García, E. (2019, Abril 1) “¿Qué es Vue.js?” [codigofacilito.com](https://codigofacilito.com/articulos/que-es-vue) [Online] disponible: <https://codigofacilito.com/articulos/que-es-vue> (accedido el 8 de octubre de 2022)
- [2] Vuejs “What is Vue” [vuejs.org](https://vuejs.org/guide/introduction.html) [Online] disponible: <https://vuejs.org/guide/introduction.html> (accedido el 8 de octubre de 2022)
- [3] Font, G. (2019, Marzo 8) “ReactJS vs Angular5 vs Vue.js ¿Cuál elegir?” [medium.com](https://javifont.medium.com/reactjs-vs-angular5-vs-vue-js-cu%C3%A1l-elegir-a9358068cf94) [Online] disponible: <https://javifont.medium.com/reactjs-vs-angular5-vs-vue-js-cu%C3%A1l-elegir-a9358068cf94> (accedido el 8 de octubre de 2022)
- [4] Bautista, I. (2021, Marzo 30) “Backend y Frontend, ¿Qué es y cómo funcionan en la programación?” [Servnet.mx](https://www.servnet.mx/blog/backend-y-frontend-partes-fundamentales-de-la-programacion-de-una-aplicacion-web) [Online] disponible: <https://www.servnet.mx/blog/backend-y-frontend-partes-fundamentales-de-la-programacion-de-una-aplicacion-web> (accedido el 8 de octubre de 2022)