

¿Cómo convertirse en un desarrollador Full Stack?

Antes de saber cómo ser un desarrollador web full stack, es necesario definir qué es un application stack. Cuando hablamos de aplicaciones web, nos enfocamos en lo que se ejecuta por encima de del SO y el hardware. Por lo general debes suponer que tienes un hardware de servidor básico en un SO Linux.

Existen 5 partes que componen un stack de aplicaciones que funcionan de la siguiente forma: los requerimientos funcionales se convierten en un prototipo funcional, que se convierte en una aplicación con escalabilidad, luego los datos se trabajan a gran escala. Para ejecutar el código, se empieza por los **requisitos comerciales** (productos para servir a un cliente).

Diseño de interfaz de usuario y la experiencia de usuario: Toma los requisitos comerciales y produce composiciones, diagramas para el equipo de front-end. Crean un flujo no ejecutable que explica cómo debería funcionar la aplicación. Posteriormente, front-end se encarga de tomar ese flujo y convertirlo en un prototipo funcional sin llegar a ser una aplicación completa. **Back-end** toma ese prototipo y lo integra a un sistema que puede soportar miles de solicitudes por minuto. **Ops** se encarga de realmente ejecutar la aplicación en un servidor que mantenga el código y se reinicie si este falla, si existe multitud de usuarios o datos entra en juego **Data** que comienza a trabajar con los datos almacenados desde el back-end.

Según el texto Cómo convertirse en un desarrollador web Full Stack en 2019, para lograr ser un desarrollador web full stack es necesario contar con ciertas habilidades que se emplean en diferentes campos:

1. Desarrollo de front-end: El desarrollo de front-end es esencial para que los usuario puedan interactuar con una aplicación, es por esto que se valora un desarrollador que sea creativo e ingenioso.
2. Desarrollo de back-end: El front-end es la parte que el usuario promedio puede observar y puede actuar sobre él, sin embargo para que el front-end funciones correctamente necesita el back-end, que a menudo es la parte oculta de la aplicación. El back-end se encarga de manejar las transacciones con la base de datos, valida los credenciales de un usuario y se encarga de toda la lógica de la aplicación.
3. Base de Datos: ¿Qué puede gestionar una aplicación sin datos? Realmente, una aplicación sin datos no tiene ningún funcionalidad. Es por esto que una BD (Base de Datos) es vital para la funcionalidad de una aplicación, ya que es necesaria para almacenar, manejar y acceder a los datos.
4. Control de versiones: Cuando una aplicación se encuentra en constante desarrollo y actualización, existen varias versiones de la misma. Bien, ¿qué pasa si nuestra última versión no funciona o directamente echamos a perder la aplicación.

Una vez conocemos los requisitos para ser un desarrollador web full stack, vamos a explorar algunos de los stack más populares en los que un desarrollador puede trabajar:

1. LAMP stack: Es la base de los sitios web alojado en Linux y consta de 4 componentes:
 - a. Linux, sistema operativo open source
 - b. Apache, servidor web
 - c. MySQL, base de datos open source
 - d. PHP, lenguaje de script de código abierto
2. MEAN stack: Sus tecnologías están basada en JavaScript, es muy popular para el desarrollo de aplicaciones web de producción rápida, robusta y mantenibles. Tiene 4 componente principales:
 - a. MongoDB, base de datos no relacional que almacena en formato JSON.
 - b. Express, framework para el desarrollo web ligero y portátil.
 - c. Angular.js, framework para desarrollar programas web HTML5 y JS.
 - d. Node.js, entorno de ejecución de JS del lado del servidor.
3. MERN stack: Al igual que MEAN stack, sus tecnologías están basadas en JS y tiene 4 componentes principales:
 - a. MongoDB, base de datos no relacional que almacena en formato JSON.
 - b. Express, framework para el desarrollo web ligero y portátil.
 - c. React, biblioteca de JS para crear interfaces de usuario.
 - d. Node.js, entorno de ejecución de JS del lado del servidor.

En este punto te puedes estar preguntando, ¿cómo elijo mi especialidad? Bueno, según la experiencia de Liz Eggleston, la ciencia de datos es la especialidad más fácil. Entre back-end y front-end, debes hacer algunas preguntas, si te sientes más conectado a la experiencia de usuario y la forma en que se ejecutan los

procesos comerciales, debes inclinarte por front-end. Si eres del tipo de persona que se cuestiona el funcionamiento de muchas cosas y te interesa conocer ciertos secretos, deberías inclinarte por el back-end.

Opinión personal

Noto mis fortalezas en el back-end, me apasiona crear rutinas que, finalmente y en conjunto, se convertirán en una funcionalidad de una aplicación. Sin embargo, me gustaría desenvolverme de una mejor forma en el front-end y desarrollar mi creatividad para desempeñarme en este área como todo un profesional. Llegar a ser un desarrollador web full stack no me parece objetivo imposible de cumplir, muy por el contrario, me gustaría llegar a ser. Es por esto que quiero pulir mis habilidades en el front-end.

Referencias

- Eggleston, L. (2019). Becoming a Full Stack Developer vs Specializing Within the Web Stack. Retrieved 9 December 2019, from <https://www.coursereport.com/blog/full-stack-developer-vs-specializing-within-the-web-stack#ProgrammingLanguagesinthe%22Stack%22>
- Kaur, H. (2019). How to Become a Full Stack Web Developer in 2019 : A Complete Guide - GeeksforGeeks. Retrieved 9 December 2019, from <https://www.geeksforgeeks.org/how-to-become-a-full-stack-web-developer-in-2019-a-complete-guide/>