Román Kornyeyev y Franco Gianello

Trabajo de investigación

Búsqueda de información sobre tecnologías web utilizadas en el desarrollo de aplicaciones web y descripción breve de las tecnologías más utilizadas.

PRACTICA 2

Despliegue de Aplicaciones Web

**1) Clientes (navegadores). Servidores.**

**2) Desarrollo/Lenguajes**

Me gustaría hacer un pequeño apartado extra y explicar brevemente que es el espectro front-end y back-end. El **front-end** forma parte de todos aquellos elementos visuales que el usuario ve y con los que interactúa directamente, se podría decir que es la cara de nuestra aplicación web/app móvil. Mientras que el **back-end** es el conjunto de todo aquello que pasa por detrás de nuestra aplicación (sea web o móvil), a nivel de servidor y son procesos con los que no interactúa directamente el usuario. En los lenguajes front-end podríamos englobar: HTML, CSS, JavaScript. En los lenguajes back-end podríamos englobar a: PHP, Java, Phyton, etc.

**Cliente:** los clientes en el ámbito del desarrollo web hacen referencia a los navegadores (Chrome, Firefox, Safari) y son los que ejecutan todos los lenguajes front-end (HTML, CSS y JavaScript), de hecho, estos tres son los únicos lenguajes que ejecutan estos clientes. Esto supone que sean incapaces de ejecutar lenguajes entorno servidor (PHP, Java, Phyton). Una forma breve de resumirlo sería que los lenguajes entorno servidor (como PHP) ejecutan una serie de órdenes y escupen un código HTML+CSS+JS, el cuál es interpretado por el cliente (navegador) y representado en la pantalla del usuario. Algunos de los navegadores web más utilizados son:

* **Google Chrome:** navegador web de Google. Utiliza el motor de renderizado Blink y está basado en chromium, la versión de código abierto de Google Chrome. Es muy intuitivo, personalizable y fácil de usar. Una de sus características especiales es que desayuna, come y cena tu RAM. También dispone de una versión para desarrolladores web llamada Chrome Canary. Tiene compatibilidad total (de S.O).
* **Mozilla Firefox:** es un navegador web libre y de código abierto desarrollado para las distintas plataformas, está coordinado por la corporación Mozilla y la Fundación Mozilla. Utiliza el motor de renderización Quantum. Es un navegador ligero y muy personalizable. Dispone de una versión para desarrolladores web llamada Firefox Developer. Tiene compatibilidad total (de S.O).
* **Safari:** es un navegador web de código cerrado desarrollado por Apple. Está basado en el motor de código abierto WebKit. Según las opiniones de los usuarios de los últimos años, por las numerosas incompatibilidades con código CSS moderno prácticamente ha pasado a ser *‘el nuevo internet explorer’*. Tiene una versión para desarrolladores web llamada Webkit Nightly. Únicamente compatible con dispositivos Apple =/
* **Opera y OperaGX:** navegador web creado por la empresa Opera Software. Utiliza el motor de renderizado Blink. Es un navegador web muy completo, ligero, extremadamente personalizable (tanto Opera como OperaGX) y dispone de una VPN integrada totalmente gratuita. A pesar de ello es bastante menos usado que los 3 mencionados anteriormente. Tiene compatibilidad total (de S.O).
* **Edge:** ‘la evolución de Internet Explorer’. Es un navegador web desarrollado por Microsoft basado en Chromium. Es el patito feo de la familia, es de los navegadores menos usados, no obstante, es ligero y bastante personalizable. Cuenta con una compatibilidad total (de S.O).

Todos estos navegadores leen, interpretan y muestran código HTML, CSS y JavaScript (los tres pilares base del desarrollo web):

* **HTML:** ‘HyperText Markup Language’, es un lenguaje de marcado para la elaboración de páginas web. Define el significado y la estructura del contenido web. HTML generalmente va de la mano con CSS y JavaScript, para darle buen aspecto y funcionalidad a las páginas.
* **CSS:** ‘Cascading Style Sheets’, es un lenguaje de diseño gráfico para definir y crear la presentación de un documento estructurado escrito en un lenguaje de marcado (HTML).
* **JavaScript:** JavaScript es un lenguaje de programación interpretado. Es un lenguaje orientado a objetos débilmente tipado.

Podríamos concluir que HTML le da estructura de una página web, CSS le da la apariencia y JavaScript le da funcionalidad.

**Servidor:** el lado del servidor es aquel en el que se ejecutan todos aquellos lenguajes del espectro back-end y que no se pueden ejecutar a nivel cliente (PHP, Phyton, Java). Algunos de los más utilizados en el ámbito web son:

* **PHP:** este acrónimo hace referencia a Hypertext Preprocessor. Fue creado por Rasmus Lerdorf en 1994. Es un lenguaje de programación de alto nivel de uso general que se adapta especialmente al desarrollo web, ya que puede ser incrustado en HTML. En lugar de usar muchos comandos para mostrar HTML (como en C o en Perl), las páginas de PHP contienen HTML con código incrustado que hace “algo”. Tiene una peculiaridad y es que es un lenguaje multiparadigma: imperativo, funcional, orientado a objetos, procedural y reflexivo. Lo que va a hacer este lenguaje es ejecutar código en el servidor, generando HTML y enviándolo al cliente (navegador). También cabe a destacar que es un lenguaje de tipado débil (como Phyton y JavaScript).
* **Java:** es un lenguaje de programación y una plataforma informática que fue comercializada por primera vez en 1995 por Sun Microsystems. Fue comprada por Oracle en 2010. Hoy en día existen muchísimas aplicaciones y sitios web que no funcionarían, probablemente, a menos que tengan Java instalado. Es un lenguaje de programación de alto nivel y orientado a objetos. Al ser un lenguaje mixto (interpretado y compilado a la vez) le concede una enorme ventaja en la compatibilidad con distintos dispositivos, pero como contrapartida, su rendimiento es ligeramente inferior a un lenguaje de programación compilado. Es un lenguaje de programación de tipado fuerte.
* **Python:** es un lenguaje de programación creado por Guido van Rossum, a principios de los 90. Es un lenguaje de programación de alto nivel, de tipado débil y orientado a objetos. La filosofía de este lenguaje hace hincapié en la legibilidad y sencillez de su código. Se usa para desarrollar aplicaciones de todo tipo (Instagram, Netflix, Spotify, entre otros).
* **Ruby:** es un lenguaje de programación creado por Yukihiro Matsumoto en 1995. Es un lenguaje de programación interpretado, es de alto nivel, de tipado fuerte y orientado a objetos.
* **Node.js:** es un entorno en tiempo de ejecución multiplataforma creado en 2009 por Ryan Dahl. Es de código abierto, para la capa del servidor basado en el lenguaje de programación JavaScript, asíncrono, con E/S de datos en una arquitectura orientada a eventos.

**Frameworks:** generalmente picar código por nuestra cuenta nos lleva muchas horas de trabajo, los frameworks ayudan a reducir drásticamente este tiempo proporcionándonos utilidades y herramientas para ello. Por tanto, podríamos definir a los frameworks como cajas de herramientas que simplifican en gran medida la elaboración de una tarea. Existen frameworks y librerías tanto front-end como back-end:

Entre los frameworks **front-end** podríamos distinguir:

Del lado de **CSS** podemos destacar:

* **Boostrap:** ‘el framework de CSS por excelencia’. Es un framework (también considerado como una librería), fue desarrollado por Mark Otto y Jacob Thornton de Twitter y fue lanzado en 2011. Es un framework de CSS y JavaScript de código abierto que favorece el desarrollo web de un modo más sencillo y rápido. Incluye plantillas de diseño basadas en HTML y CSS con la que es posible modificar tipografías, formularios, botones, tablas, navegaciones, menús desplegables y todo tipo de elementos HTML. Esto nos ahorra horas y horas de trabajo.
* **Bulma:** ‘la marca blanca de Boostrap’. Es un framework lanzado en 2016. Está basado en flexbox, en el que solo necesitas un archivo CSS para comenzar a utilizarlo en tus proyectos o puedes optar por la versión en la que descargas los archivos SASS y modificar las variables a tu gusto. A diferencia de Boostrap, bulma no usa JavaScript, sino únicamente CSS y SASS. Al igual que Boostrap, Bulma es muy útil y consigue ahorrarnos horas de trabajo.

Adentrándonos en **JavaScript** podemos destacar:

* **Angular:** ‘El framework top de JS (fron-end)’. Es un framework que fue desarrollado en 2009 por Misko Hevery, inicialmente se llamaba AngularJS, en 2016 fue lanzado Angular2 (que conocemos en la actualidad como Angular). Angular fue desarrollado para aplicaciones web. Está basado en TypeScript, es de código abierto y es mantenido por Google. Se utiliza para crear y mantener aplicaciones web de una sola página en el lado del cliente usando HTML y TypeScript.
* **React:** aunque es una librería (y no un framework), no obstante, veo relevante incluirlo, ya que se podría decir que es el competidor directo de Angular. React fue creado originalmente por Jordan Walke en 2013, pero actualmente React tiene más de mil contribuidores de código abierto. React es una librería de JavaScript de código abierto, diseñada para crear interfaces de usuario con el objetivo de facilitar el desarrollo de aplicaciones en una sola página. Es mantenido por Facebook y la comunidad de software libre. A diferencia de Angular, React no se limita a la web, por ejemplo, React Native puede usarse para desarrollar aplicaciones móviles, React 360 permite crear aplicaciones de realidad virtual, entre otros.
* **Vue.js:** es un framework de JavaScript creado por Evan You y lanzado en 2014, es mantenido por él y por el resto de los miembros activos dele quipo central que provienen de diversas empresas como Netlify y Netguru.

Entre los frameworks **back-end** podríamos distinguir: