¿Qué es Node js?

es un entorno en tiempo de ejecución multiplataforma, de código abierto, para la capa del servidor (pero no limitándose a ello) basado en el lenguaje de programación ECMAScript, asíncrono, con E/S de datos en una arquitectura orientada a eventos y basado en el motor V8 de Google. Fue creado con el enfoque de ser útil en la creación de programas de red altamente escalables, como por ejemplo, servidores web.​ Fue creado por Ryan Dahl en 2009 y su evolución está apadrinada por la empresa Joyent, que además tiene contratado a Dahl en plantilla.

Sus aplicaciones

1. NETFLIX
2. PAYPAL
3. UBER
4. LINKEDIN
5. EBAY

¿Qué es un microservicio?

s una aproximación para el desarrollo de software que consiste en construir una aplicación como un conjunto de pequeños servicios, los cuales se ejecutan en su propio proceso y se comunican con mecanismos ligeros (normalmente una API de recursos HTTP). Cada servicio se encarga de implementar una funcionalidad completa del negocio. Cada servicio es desplegado de forma independiente y puede estar programado en distintos lenguajes y usar diferentes tecnologías de almacenamiento de datos.

¿Qué es API?

Las API permiten que sus productos y servicios se comuniquen con otros, sin necesidad de saber cómo están implementados. Esto simplifica el desarrollo de las aplicaciones y permite ahorrar tiempo y dinero. Las API le otorgan flexibilidad; simplifican el diseño, la administración y el uso de las aplicaciones, y proporcionan oportunidades de innovación, lo cual es ideal al momento de diseñar herramientas y productos nuevos (o de gestionar los actuales).

¿Qué es FTP?

El Protocolo de transferencia de archivos es un protocolo de red para la transferencia de archivos entre sistemas conectados a una red TCP, basado en la arquitectura cliente-servidor.

Arquitecturas:

Cliente- servidor:

La **arquitectura cliente**-**servidor** es un modelo de diseño de software en el que las tareas se reparten entre los proveedores de recursos o servicios, llamados **servidores**, y los demandantes, llamados **clientes**. Un **cliente** realiza peticiones a otro programa, el **servidor**, quien le da respuesta.

Características:

* Es quien inicia solicitudes o peticiones, tienen por tanto un papel activo en la comunicación (dispositivo **maestro** o **amo**).
* Espera y recibe las respuestas del servidor.
* Por lo general, puede conectarse a varios servidores a la vez.

Al **receptor de la solicitud** enviada por el cliente se conoce como servidor. Sus características son:

* Al iniciarse esperan a que lleguen las solicitudes de los clientes, desempeñan entonces un papel pasivo en la comunicación (dispositivo **esclavo**).
* Tras la recepción de una solicitud, la procesan y luego envían la respuesta al cliente.
* Por lo general, acepta las conexiones de un gran número de clientes (en ciertos casos el número máximo de peticiones puede estar limitado).

En la arquitectura C/S sus características generales son:

* El Cliente y el Servidor pueden actuar como una sola entidad y también pueden actuar como entidades separadas, realizando actividades o tareas independientes.
* Las funciones de Cliente y Servidor pueden estar en plataformas separadas, o en la misma plataforma.
* Cada plataforma puede ser escalable independientemente. Los cambios realizados en las plataformas de los Clientes o de los Servidores, ya sean por actualización o por reemplazo tecnológico, se realizan de una manera transparente para el usuario final.
* La interrelación entre el hardware y el software están basados en una infraestructura poderosa, de tal forma que el acceso a los recursos de la red no muestra la complejidad de los diferentes tipos de formatos de datos y de los protocolos.
* Su representación típica es un centro de trabajo (PC), en donde el usuario dispone de sus propias aplicaciones de oficina y sus propias bases de datos, sin dependencia directa del sistema central de información de la organización.

¿Qué es MVC?

s un patrón de arquitectura de software, que separa los datos y la lógica de negocio de una aplicación de su representación y el módulo encargado de gestionar los eventos y las comunicaciones. Para ello MVC propone la construcción de tres componentes distintos que son el **modelo**, la **vista** y el **controlador**, es decir, por un lado define componentes para la representación de la información, y por otro lado para la interacción del usuario.[1](https://es.wikipedia.org/wiki/Modelo%E2%80%93vista%E2%80%93controlador#cite_note-1)​[2](https://es.wikipedia.org/wiki/Modelo%E2%80%93vista%E2%80%93controlador#cite_note-burbeck-2)​ Este patrón de arquitectura de software se basa en las ideas de reutilización de código y la separación de conceptos, características que buscan facilitar la tarea de desarrollo de aplicaciones y su posterior mantenimiento.

Otras arquitecturas:

Patrón de capas:

Este patrón se puede utilizar para estructurar programas que se pueden descomponer en grupos de subtareas, cada una de las cuales se encuentra en un nivel particular de abstracción. Cada capa proporciona servicios a la siguiente capa superior.

Las 4 capas más comúnmente encontradas de un sistema de información general son las siguientes.

* **Capa de presentación** (también conocida como **capa UI** )
* **Capa de aplicación** (también conocida como **capa de servicio** )
* **Capa de lógica de negocios** (también conocida como **capa de dominio** )
* **Capa de acceso a datos** (también conocida como **capa de persistencia** )

Patron maestro-esclavo:

Este patrón consiste en dos partes; **maestro** y **esclavos**. El componente maestro distribuye el trabajo entre componentes esclavos idénticos y calcula el resultado final de los resultados que devuelven los esclavos.

Patron del agente:

Este patrón se usa para estructurar sistemas distribuidos con componentes desacoplados. Estos componentes pueden interactuar entre sí mediante invocaciones de servicios remotos. Un componente de **intermediario** es responsable de la coordinación de la comunicación entre los **componentes**.

Los servidores publican sus capacidades (servicios y características) a un intermediario. Los clientes solicitan un servicio del intermediario y el intermediario redirecciona al cliente a un servicio adecuado desde su registro.

¿Qué microservicios ofrece Facebook?

* GUARDA EL CONTENIDO PARA LEERLO DESPUÉS.
* ELIGE QUÉ QUIERES VER PRIMERO EN EL MURO.
* ASEGÚRATE DE NO PERDERTE NINGÚN EVENTO.
* JUEGA CON MESSENGER.
* CONECTA EN DIRECTO CON TUS AMIGOS.
* AÑADE UN VÍDEO COMO FOTO DE PERFIL.
* COMPARTE UNA FOTO EN 360 GRADOS.

¿Qué microservicios ofrece Twitter?

* Para hacer encuestas.
* Envíe tweets cuando esté ausente.
* Estadísticas de su perfil.
* Búsqueda de información.
* Compartir fotografías.
* Reduzca las direcciones web.
* Video en línea.
* Preguntas y respuestas.

¿Qué microservicios ofrece Google?

* Google Search Engine. Google Search Engine es el motor de búsqueda más popular y ampliamente utilizado en el mundo.
* Google Chrome.
* Gmail.
* Android.
* YouTube.
* Google Adsense.
* Google Drive.
* Google Ads.

¿Qué microservicios ofrece YouTube?

* YouTube TV – Modo de YouTube Televisión. YouTube TV te permite utilizar el servicio como si fuese una "Smart TV" (Televisión Inteligente).
* YouTube Music – Videos de tus artistas favoritos.
* YouTube Charts – Videos más populares de YouTube.
* Watch Movies – Películas completas.

¿Que es REST?

**REST es una interfaz para conectar varios sistemas basados en el protocolo HTTP**(uno de los protocolos más antiguos) y nos sirve para obtener y generar datos y operaciones, devolviendo esos datos en formatos muy específicos, como XML y JSON.

¿Qué es SOAP?

es un protocolo estándar que define cómo dos objetos en diferentes procesos pueden comunicarse por medio de intercambio de datos XML. Este protocolo deriva de un protocolo creado por Dave Winer en 1998, llamado XML-RPC. SOAP fue creado por Microsoft, IBM y otros. Está actualmente bajo el auspicio de la W3C. Es uno de los protocolos utilizados en los servicios Web.

¿Qué es RESFUL?

es un protocolo estándar que define cómo dos objetos en diferentes procesos pueden comunicarse por medio de intercambio de datos XML. Este protocolo deriva de un protocolo creado por Dave Winer en 1998, llamado XML-RPC. SOAP fue creado por Microsoft, IBM y otros. Está actualmente bajo el auspicio de la W3C. Es uno de los protocolos utilizados en los servicios Web.

Aunque muchas veces se usan los términos Rest y Restful como sinónimos, no lo son. Rest (Representational State Transfer), es un modelo de arquitectura web basado en el protocolo HTTP para mejorar las comunicaciones cliente-servidor, mientras que Restful Web Service o Restful API son programas basados en REST.

¿Qué es bioinformática?

La bioinformática es una disciplina que permite aplicar herramientas de la informática al estudio y la gestión de datos de la biología. En su campo interactúan diversas áreas del conocimiento como las ciencias de la computación, la estadística y la química.

¿Qué son las variables de entorno?

es una variable dinámica que puede afectar al comportamiento de los procesos en ejecución en un ordenador.

Son parte del entorno en el que se ejecuta un proceso. Por ejemplo, un proceso en ejecución puede consultar el valor de la variable de entorno TEMP para descubrir una ubicación adecuada para almacenar archivos temporales, o la variable HOME o USERPROFILE para encontrar la estructura de directorios propiedad del usuario que ejecuta el proceso.

Fueron introducidos en su forma moderna en 1979 con la versión 7 de Unix, por lo que están incluidos en todos los sabores y variantes del sistema operativo Unix a partir de ese momento, incluyendo Linux y macOS. Desde PC DOS 2.0 en 1982, todos los sistemas operativos de Microsoft, incluyendo Microsoft Windows y OS/2, también los han incluido como una característica, aunque con sintaxis, uso y nombres de variables estándar algo diferentes.