

Curso de Node.js, Javascript y Git

Requerimientos para el curso

- Instalar SourceTree para el SO que corresponda (<https://www.sourcetreeapp.com/>)
- Crear sus usuarios para el repositorio de Bitbucket usando la invitación que recibieron por correo
- Descargar el instalador de Node para al SO que corresponda (<https://nodejs.org/>)
- Descargar el instalador de MongoDB para el SO que corresponda (<https://www.mongodb.org/downloads>)
- Instalar un editor de textos de preferencia:
 - o Sublime Text 2 (<http://www.sublimetext.com/2>)
 - o Notepad++ (<https://notepad-plus-plus.org/>)
 - o Brackets (<http://brackets.io/>)
 - o Bloc de notas (=P)

Temario

Día 1

1. ¿Qué es Node.js?
2. Instalación y crear mi primer programa “Hola mundo”
3. Introducción a Javascript Programación Orientada a Objetos

Día 2

4. Ejercicios con Javascript utilizando Node.js
5. Instalación de Express y crear un proyecto

Día 3

6. Manejar rutas y peticiones http
7. ¿Qué es MongoDB?
8. Instalación de MongoDB y ejercicio con Node.js

Día 4

9. ¿Qué es Socket.io y la librería NET?
10. Instalación de Socket.io y ejercicio con Node.js
11. Ejercicio con la librería NET

Día 5

12. Control de versiones con Git
13. Ejercicio con Javascript y Node.js utilizando Git

Lugar, Fecha y horario:

Del Lunes 31 de Agosto al Viernes 4 de Septiembre.

De las 8:00 a 10:00am

Sala de VideoConferencias (A2)

Contenido

Contenido	2
¿Qué es Node.js?	3
¿Qué se puede hacer con node.js?	3
¿Qué problemas resuelve node.js?	3
¿Por qué Javascript?	3
Características de node.js	3
¿En donde se utiliza node.js actualmente?	4
Notas interesantes para leer:	4
Tecnologías relacionadas con node.js	5

¿Qué es Node.js?

Node.js es un software de código abierto que funciona como framework del lado del servidor, que utiliza para la ejecución del código interpretado de Javascript el motor V8 Javascript de Chrome. Node.js utiliza una arquitectura orientada a eventos.

Node.js permite que el servidor y las aplicaciones de escritorio se comuniquen por medio de Javascript.

En general podemos decir que Node.js es un entorno de ejecución y una biblioteca, que nos permite crear aplicaciones altamente escalables. Esto último es la meta número uno declara por node.js, “proporcionar una manera fácil para construir programas de red escalables”.

¿Qué se puede hacer con node.js?

Podemos generar desde pequeños módulos, hasta aplicaciones complejas, ya que node.js permite manejar cientos de miles de conexiones concurrentes; así mismo, por la manera en que node.js está implementado, es totalmente adecuado para lo que tiene que ver con intercomunicación a gran escala, como aplicaciones de mensajería, juegos multijugador y aplicaciones en tiempo real.

¿Qué problemas resuelve node.js?

Node.js modifica la forma en que se realiza una conexión con el servidor, ¿cómo?, hace que en lugar de generar un nuevo hilo de OS para cada conexión (y de asignarle la memoria acompañante), cada conexión dispara una ejecución de evento dentro del proceso del motor de node. Node afirma que un servidor que lo ejecute puede soportar decenas de miles de conexiones concurrentes.

¿Por qué Javascript?

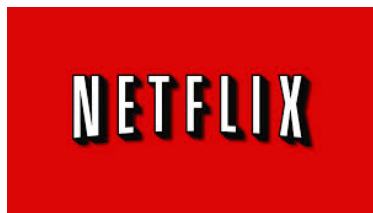
JavaScript es un gran lenguaje para la programación asíncrona, ya que fue diseñado para ser usado en programación orientada a eventos en lugar de otros lenguajes orientados a objetos, como por ejemplo, Java. Es especialmente atractivo para realizar aplicaciones ‘no bloqueantes’ y de alta concurrencia y disponibilidad.

Características de node.js

- Se utiliza Javascript del lado del servidor
- Utiliza un ciclo de eventos en lugar de hilos nuevos por cada conexión
- Cada operación de I/O de node es asíncrona
- Puede manejar miles de conexiones simultáneas con una sobrecarga mínima en un solo proceso
- Amplia gama de herramientas a través de npm
- Ahorro en infraestructura. Especialmente en el número de servidores necesarios para las aplicaciones. Por ejemplo, cuando LinkedIn migró todo su backend desde Ruby on Rails hacia Node.js redujo desde 30 hasta 3 su número de servidores. (<http://www.nodehispano.com/2012/10/linkedin-migra-desde-rails-hacia-node-nodejs/>)

- Conjunción entre BackEnd y FrontEnd. Los equipos de BackEnd y FrontEnd móvil pueden ser combinados en un conjunto único.
- Flexibilidad.
- Fácil de usar. Sobre todo si tienes experiencia con JavaScript.

¿En donde se utiliza node.js actualmente?



En el siguiente link se puede ver un listado más extenso de las empresas que utilizan node.js

<https://github.com/joyent/node/wiki/projects,-applications,-and-companies-using-node>

Notas interesantes para leer:

<https://www.talentbuddy.co/blog/building-with-node-js-at-netflix/>

<http://venturebeat.com/2012/01/24/why-walmart-is-using-node-js/>

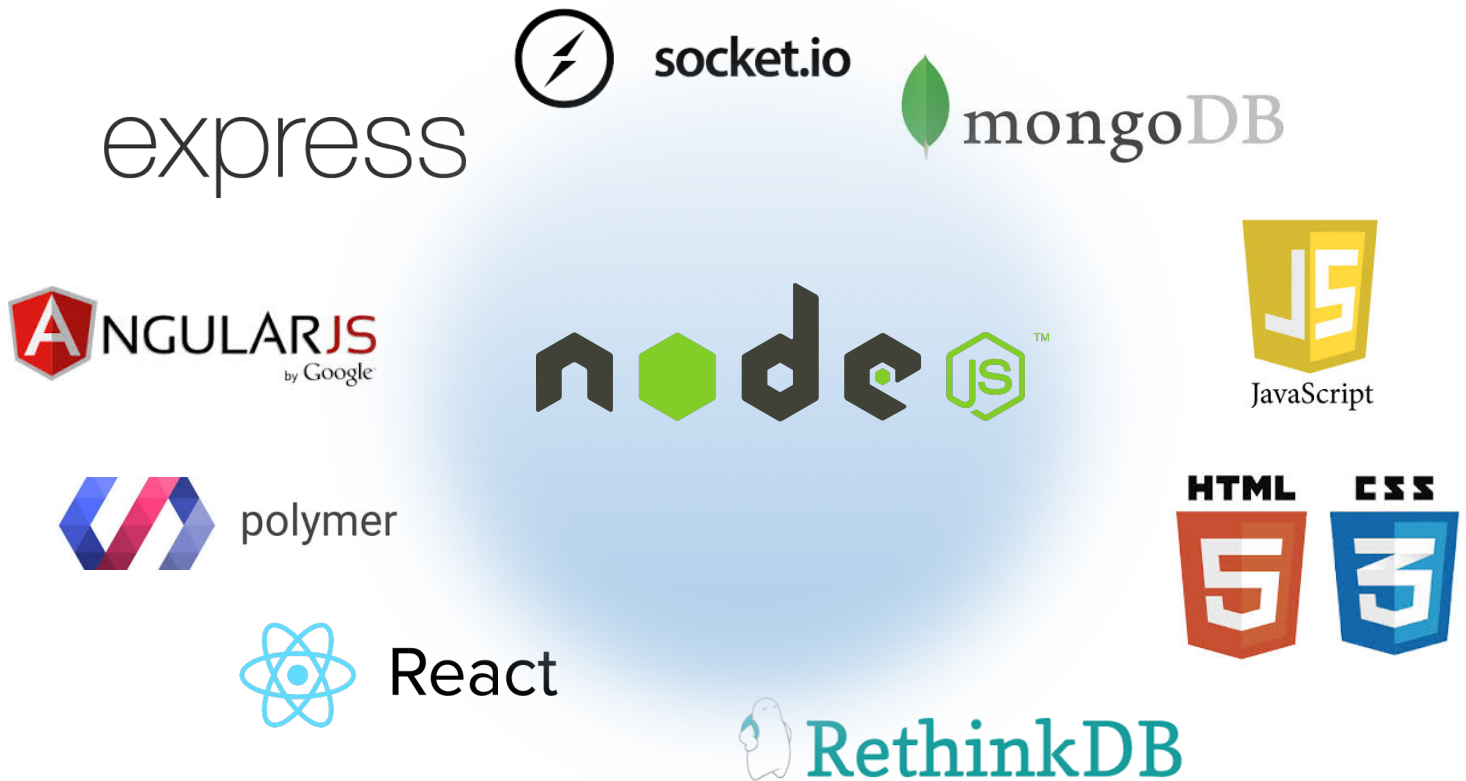
<https://www.paypal-engineering.com/2013/11/22/node-js-at-paypal/>

<http://venturebeat.com/2011/08/16/linkedin-node/>

<http://venturebeat.com/2012/01/24/node-at-google-mozilla-yahoo/>

<http://highscalability.com/blog/2012/10/4/linkedin-moved-from-rails-to-node-27-servers-cut-and-up-to-2.html>

Tecnologías relacionadas con node.js



En conclusión node.js es una plataforma basada en el motor de JavaScript V8 de Google que es utilizado en el navegador Chrome. Está pensada para facilitar el desarrollo de aplicaciones basadas en red, rápidas y fiables. Node.js utiliza un modelo I/O (entrada/salida) orientado a eventos y basado en el 'no-bloqueo', que lo hace ligero y eficiente, ideal para aplicaciones en tiempo real que hagan uso de datos intensivos y que se ejecuten a través de dispositivos distribuidos.



Dirección de Sistemas Microelectrónicos



Instalación de node.js