

# NodeJS

lunes, 24 de febrero de 2014 04:08 p.m.

## Windows

Paso 1: Ir a <http://nodejs.org>, descargar el archivo e instalar  
Paso 2: Buscar en sus programas instalados: "Node.js Command Prompt"  
Paso 3: Teclear `node -v` para comprobar que se instaló d(eberan tener v0.8.x)

## Mac

Paso 1: Ir a <http://nodejs.org>  
Paso 2: Descargar el archivo  
Paso 3: Instalar NodeJs  
Paso 4: Abrir el programa "Terminal"  
Paso 5: Teclear `node -v` para comprobar que se instaló  
Deberán tener v0.8.x

## Ubuntu

Paso 1: Abrir la terminal  
Paso 2: Instalar dependencias así: `sudo apt-get install build-essential git-core libssl-dev libssl0.9.8`  
Paso 3: Descargar Node: `git clone git://github.com/joyent/node`  
Después entra al folder que acabas de descargar con: `cd node`  
Paso 4: Pasar a NodeJs 0.8.4: `git checkout v0.8.4`  
Paso 5: Compilar e instalar NodeJs:  
`./configure`  
`make`  
`sudo make install`  
Paso 6: Teclear `node -v` para comprobar que se instalo deberán tener v0.8.x

## Pasando de PHP a NodeJS

Cuando programamos en PHP el ambiente normal es XAMP donde tenemos que compartir las tareas de servidor web y programación entre Apache y PHP. Es un ambiente donde todo tiene lugar en un folder y sabemos dónde se encuentra el root de nuestro proyecto, dónde deben ir nuestros archivos (como `index.php`) y nuestro `.htaccess`. Generalmente todo es cuesta abajo.

En node no existe un root del proyecto ni un archivo `.htaccess`, lo cual suele confundir mucho a primera vista.

Otro factor que confunde mucho la primera vez que entras a node, es que casi todo lo tienes que hacer en la terminal.

**Nota importante:** Cuando inicies su terminal de node, naveguen al folder que usarán para su proyecto. Nodejs podrá vivir en cualquier folder que deseen de su sistema operativo y en ese folder requieren 2 archivos para empezar su proyecto:

- 1) Un archivo de Javascript al que pueden ponerle el nombre que deseen.
- 2) Un archivo JSON llamado `package.json`.

En el archivo de JSON definirán el nombre de su proyecto, la versión de su proyecto, qué librerías están usando y muchas cosas más.

```
{
  "name": "NombreDeTuApp",
  "version": "0.0.1",
  "dependencies": {
    "express": "3.0.3",
    "jade": "0.27.7",
    "consolidate": "0.5.0",
    "socket.io": "0.9.11"
  }
}
```

Una vez que hayan hecho su archivo `package.json` podrán usar `npm` para instalar todas sus librerías (llamadas dependencias) con un manejador de paquetes que viene instalado por default con Node llamado NPM (node package manage) el cuál será su fiel acompañante el resto de su desarrollo con Node.

Para instalar las dependencias del proyecto tienen que usar la siguiente instrucción:

```
npm install
```

Deberán de tener un resultado como este:

```
[ws v0.4.22] Attempting to compile blazing fast native extensions.
[ws v0.4.22] Native extension compilation successful
consolidate@0.5.0 node_modules/consolidate
├── jade@0.27.7 node_modules/jade
│   ├── commander@0.5.1
│   ├── mkdirp@0.3.4
│   └── coffee-script@1.4.0
└── express@3.0.3 node_modules/express
    ├── method@0.0.1
    ├── rx@0.1.0
    ├── range-parser@0.0.4
    └── cookie-signature@0.0.1
```

```

├─ cookie@0.6.5
├─ core@2.0
├─ commander@0.6.1
├─ debug@0.7.0
├─ mkdirp@0.3.3
├─ send@1.8 (mime@1.2.6)
├─ connect@2.7.0 (pause@0.0.1, bytes@1.0, formidable@1.0.11, qs@0.5.1)
├─ socket.io@0.9.11 node_modules/socket.io
├─ policyfile@0.0.4
├─ redis@0.7.3
├─ socket.io-client@0.9.11 (xmlhttprequest@1.4.2, uglify-js@1.2.5, active-x-obliterator@0.0.1, ws@0.4.23)
└─ sitedrix-MacBook-Pro-socket.io sitedrix

```

Y con eso podremos empezar a trabajar.

Lo siguiente que necesitamos es tener un archivo que inicie nuestra aplicación. En este caso le llamaré "server.js" porque es un servidor web. Pondremos el "Hello world" de ExpressJs:

```
var express = require('express');var app = express();
```

```
app.get('/', function(req, res){
  res.send('hello world');
});
```

```
app.listen(3000);console.log('Si ves esto ve a http://localhost:3000 \n
Oprime CTRL + C para apagar el servidor');
```

Una vez que esto esté en nuestro archivo "server.js" pasaremos a nuestra terminal y escribiremos:

```
node server.js
```

Y tendremos en pantalla:

Si ves esto ve a <http://localhost:3000/>  
Oprime CTRL + C para apagar el servidor

Podemos checar en nuestro browser en la dirección <http://localhost:3000/> que aparece un bonito "hello world".

Sin importar qué tan grandes sean sus proyectos de node, siempre seguirán este mismo proceso.

## Enlaces

[NodeJS](#). La página oficial del proyecto de NodeJS. Aquí encuentras toda la documentación, comunidad, su repositorio en GitHub, así como varios manuales que pudieran ayudarte a estudiar.

[Node GitHub](#). Te recomendamos que conozcas el Wiki y el proyecto. Puedes entrar a la sección de módulos donde se te invita a extender la funcionalidad de Node y crear proyectos más completos.

[NPM](#). También conocido como Node Package Manager. Es un gestor para NodeJS que te permite instalar paquetes de software de forma sencilla y fácil. Normalmente cuando instalas NodeJS, ya viene incluido por defecto.

## Comunidades y Blogs

[Foro NodeJS](#). Grupo de Google dónde se promueven discusiones de Node.

[How To Node](#). Una comunidad de Tim Caswell, el cual tiene como propósito debatir y enseñar conceptos fundamentales de la tecnología de Node.

[NodeJitsu](#). Un blog que se enfoca, además de buenas prácticas, en la escalabilidad de Node sobre las aplicaciones Web, incluyendo los denominados servidores en la nube.

## Guías y Tutoriales

[El libro de principiantes de Node](#). Una guía que te introduce a la plataforma de Node de forma empática y sencilla.

[Node CheatSheet](#). Un pequeño resumen de los principales comandos de Node.

[NodeGuide](#). La guía NodeJS para iniciados con Felix Geisendörfer.

[NodeJS First Look](#). Un curso de Lynda.com donde te enseña a través de videos cuál es el funcionamiento de NodeJS, además de sus conceptos fundamentales.

[Node Tuts](#). Video tutoriales de Node para iniciados.

[Mixu's Node Book](#). Un pequeño libro electrónico de Node. Recomendado si ya conoces las bases.

[Node: Up and Running](#). El libro de Tom Hughes para trabajar con Node de forma totalmente práctica.

## Herramientas Node

[Express](#). Un framework que te provee un robusto conjunto de funcionalidades para construir páginas individuales y múltiples, generando aplicaciones web híbridas.

[Socket.IO](#). Los famosos "Sockets", que te permiten crear aplicaciones en tiempo real a través de la gestión de conexiones, funcionales en cualquier dispositivo y navegador, incluyendo móviles.

[Stylus](#). Un compilador CSS que te permite trabajar de forma más sintetizada y práctica. Recomendada por Mejorándola.

[Jade](#). Un motor de plantillas para aplicaciones NodeJS. Combina la potencia y flexibilidad del código

HTML con una sintaxis agradable y limpia.

[Meteor](#). Una plataforma que va construyendo la aplicación en tiempo real con el navegador.

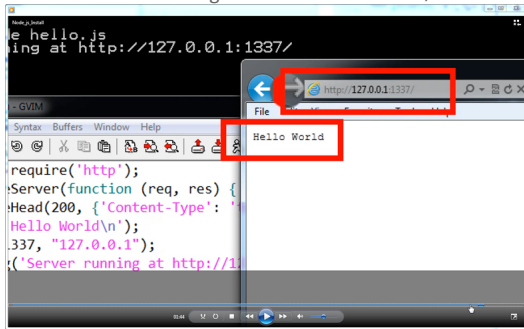
ERROR DE SSL EN LINUX

`npm config set registry http://registry.npmjs.org/`

ERROR DE SSL EN WINDOWS

```
npm install express
npm install socket.io
npm install mongojs
etc.
```

Sobre Windows se ingresa 127.0.0.1:3000/



//Respuesta HTML

```
app.get('/conexiones', function(req, res) {
  res.writeHead(200, { 'Content-Type': 'text/html' });
  res.write('PRUEBA');
  res.end();
});
```

//Respuesta JSON

```
app.get('/conexiones', function(req, res) {
  cantidad= Object.keys(conexiones).length;
  res.json({ Cantidad: cantidad, Conexiones: conexiones, });
});
```