

Cómo crear una nueva aplicación con Express y nodejs 6.x

Desde octubre del 2016 en guebs.com ya soportamos la versión 6.x.x de NodeJs «Boron» LTS. Junto con ella puedes usar el conocido framework para NodeJs Express. Para poner en marcha una aplicación mínima de tipo «Hola Mundo» con Express sólo debes seguir estos pasos mínimos:

- En primer lugar debes crear la aplicación en el panel de hosting

Como acceder al Panel de Hosting

- Hecho esto debes acceder via ssh y crear la carpeta, si no existe, **nodejs/nombre_que_he_dado_a_mi_aplicacion**

Como acceder vía SSH a tu cuenta

- Ejecutamos la creación de esqueleto inicial de la aplicación con **npm init**. Hay que tener en cuenta que el punto de entrada es **app.js** y no index.js en nuestros servicios.

```
cd nodejs/miaplicacion
npm init
```

This utility will walk you through creating a package.json file.

It only covers the most common items, and tries to guess sensible defaults.

See `npm help json` for definitive documentation on these fields

and exactly what they do.

Use ``npm install <pkg> --save`` afterwards to install a package and save it as a dependency in the `package.json` file.

Press `^C` at any time to quit.

name: (miapp)

version: (1.0.0)

description: Mi aplicación

entry point: (index.js) app.js

test command:

git repository:

keywords:

author:

license: (ISC)

About to write to `/home/node6/nodejs/miapp/package.json`:

```
{
  "name": "miapp",
  "version": "1.0.0",
  "description": "Mi aplicación",
  "main": "app.js ",
  "scripts": {
    "test": "echo \"Error: no test specified\" && exit 1"
  },
  "author": "",
  "license": "ISC"
}
```

Is this ok? (yes) yes

- Ahora instalamos `express` y lo añadimos como dependencia a la aplicación:

```
npm install express --save
```

- Ahora sólo tenemos que crear un archivo **app.js** con el siguiente contenido básico por ejemplo

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

app.get('/', function (req, res) {
  res.send('Hola Mundo!');
});

app.listen(3000, function () {
  console.log('Example app listening on port 3000!');
});
```

Recuerda que en nuestros servicios no es necesario llamar al puerto. Automáticamente encontrarás tu aplicación al llamar a tu dirección web: <http://www.dominio.com>

- Finalmente sólo es necesario que crees el directorio **public** para tus recursos estáticos:

```
mkdir nodejs/miapp/public
```

Usando express-generator

Existe un módulo llamado express-generator que también permite instalar Express de un modo rápido. La documentación de express-generator se encuentra en la siguiente dirección web:

<http://expressjs.com/es/starter/installing.html>

Para instalarlo, desde ssh:

```
npm install express-generator
```

Una vez lo tengas instalado puedes crear tu aplicación ejecutando el siguiente comando:

```
./node_modules/express-generator/bin/express --view=pug  
nombredemiaplicacion
```

En el caso de este método nos encontraremos con que tenemos un archivo

bin/www . Este archivo podemos usarlo para ponerlo en el lugar del app.js.

```
cp app.js app-file.js ; cd nombredemiaplicacion; cp -fv  
bin/www app.js
```