

Presentación del curso



Hola!

Soy José Manuel Rosa Moncayo

Desarrollo de distintas APIs y microservicios a nivel personal y profesional

Actualmente trabajo como Frontend Engineer en Spotahome

Amante de JavaScript, la tecnología, las series y el Open Source!

Inquieto por naturaleza

REQUISITOS

- ▶ Conocer el lenguaje, Javascript, y algo del nuevo standard
- ▶ En caso de no saber el lenguaje, al menos, conocer otro lenguaje.
- ▶ Conocimientos de la terminal de comandos

¿POR QUÉ HACER ESTE CURSO?

- ▶ Alta demanda laboral
- ▶ Trabajar como desarrollador fullstack
- ▶ Uno de las tecnología más utilizadas
- ▶ Porque mola !

¿QUÉ VEREMOS EN EL CURSO?

- ▶ Introducción a Node
- ▶ Repaso por algunas de las features nuevas de ES6
- ▶ Sockets, programación asíncrona, eventos, ...
- ▶ Crearemos nuestro servidor web, sin librerías
- ▶ Conoceremos Express, y haremos nuestro servidor web
- ▶ Crearemos una API Rest con Express
- ▶ Lo prepararemos para producción
- ▶ Dockerizamos nuestro servidor web

Pero sobre todo harás ...

VARIOS SERVIDORES WEB CON DISTINTAS TECNOLOGÍAS



express

QUÉ VAS A SABER A FINALIZAR EL CURSO?



express



PREPARADO PARA DAR UN SALTO PROFESIONAL?

A qué esperas !

¿Qué es NodeJS?



ENTORNO DE EJECUCIÓN PARA JAVASCRIPT



Node.js® es un **entorno de ejecución** para **JavaScript** construido con el motor de JavaScript V8 de Chrome.

<https://nodejs.org/es/>



1.

Características

DESARROLLO EN JAVASCRIPT

Al ser un entorno de ejecución para JS, nos permite desarrollar servidores, herramientas, y demás con este lenguaje.

BASADO EN EL MOTOR

V8 DE CHROME

Uno de los motores de más avanzado del mercado, aunque hay impletenzaciones con Chakra (Microsoft, ...)

OPERACIONES DE E/S SIN BLOQUEOS

ORIENTADO A EVENTOS

La asincronía es un efecto resultando de estos eventos, por lo que con NodeJS, podemos hacer nuestros desarrollo asíncronos.

POR TODO ELLO ES LIVIANO Y EFICIENTE

Esto es debidos a estar basado en eventos, y
sin bloqueos

POR LO QUE ES BUENA OPCIÓN PARA DESARROLLAR

Servidores Web

Con paquetes propios de NodeJS o de terceros, como Express, Koa, Hapi

Sockets

Desarrollos basados en sockets, como chats, y aplicaciones en tiempo real

IOT

Podemos programar nuestras placas Arduino, por ejemplo, con NodeJS, al ser liviano.



Gracias!

Preguntas?

Podeis encontrarme en twitter @jmanuel_rosa

¿Qué es NPM?



NODE PACKAGE MANAGER

“

npm is the **package manager** for JavaScript and the world's largest software registry. Discover packages of reusable code — and assemble them in powerful new ways.

<https://www.npmjs.com/>

A large, stylized, light gray double quote icon consisting of two separate, slightly curved strokes.

Use npm to install, share, and distribute code; manage dependencies in your projects; and share & receive feedback with others.

<https://www.npmjs.com/>

package.json

Es el fichero donde podemos almacenar todos los paquetes de NPM que utilizamos en nuestro proyecto

package-lock.json

Se encarga de presentar una snapshot del árbol de dependencias de los paquetes de nuestro proyecto, es decir, convierte nuestro proyecto en determinista



1.

Yarn

ES LA ALTERNATIVA DE FB A NPM

Survió antes de la versión 5 de NPM, que no era determinista. Yarn genera un fichero yarn.lock que es equivalente al fichero package-lock.json



Gracias!

Preguntas?

Podeis encontrarme en twitter @jmanuel_rosa

**¿Qué podemos hacer
con NodeJS?**



CHAT

- ▶ Forma más típica en tiempo real
- ▶ Ligera
- ▶ Alto tráfico de datos
- ▶ Funciona en dispositivos distribuidos

API

- ▶ Uso de objetos JSON (Sin transformaciones)
- ▶ MongoDB
- ▶ Fácil de implementar con paquetes como Express, Koa, Hapi.js,
- ▶ Las peticiones HTTP son eventos aislados, por lo que en Node, podríamos procesar ficheros, codificar videos, etc.

Entradas en espera

- ▶ Trabajo con Bases de datos es bloqueante
- ▶ Al no estar basado en bloqueos, podemos utilizar cualquier MQ o gestor de colas para estas acciones
- ▶ Por ello, es bueno para la concurrencia

Proxy

- ▶ Como servidor proxy
- ▶ Puede manejar gran cantidad de conexiones simultáneas con modo no-bloqueo

IOT

- ▶ Consume pocos recursos
- ▶ Utilizar un solo lenguaje, Javascript
- ▶ Conexiones simultáneas
- ▶ E/S sin bloqueos

IOT

- ▶ Consume pocos recursos
- ▶ Utilizar un solo lenguaje, Javascript
- ▶ Conexiones simultáneas
- ▶ E/S sin bloqueos

Y TODO ELLO SABIENDO SOLO JAVASCRIPT!

Bueno, y algunas cositas de Node :P



Gracias!

Preguntas?

Podeis encontrarme en twitter @jmanuel_rosa

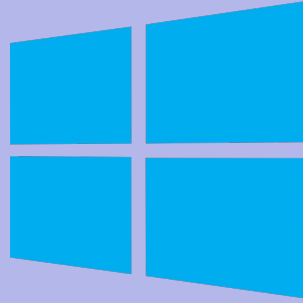
Instalación



¿Cómo?

DISTINTOS PAQUETES

Existen distintos paquetes o instaladores de NodeJS para los distintos Sistemas Operativos. Además, en linux, podemos instalarlo desde cualquier fuente de cualquier distribución





1.

NVM

NODE VERSION MANAGER

Nos permite gestionar distintas versiones de NodeJS en un mismo ordenador

.nvmrc

Por cada proyecto, nos permite definir qué versión de NodeJS queremos utilizar, lo que es bastante útil a la hora de desarrollar



2.

**Veamos cómo
funciona!**



Gracias!

Preguntas?

Podeis encontrarme en twitter @jmanuel_rosa

**Comunidad y
ecosistema**





1.

COMUNIDAD

UNA DE LAS MÁS GRANDES!

Gracias a NPM

CRECIENDO CADA VEZ MÁS

A veces, resulta abrumador la cantidad de paquetes que podemos utilizar, lo que convierte nuestros desarrollos en algo fácil.

TODO GRACIAS A JAVASCRIPT

Al ser en lenguaje en el que se desarrolla en NodeJS, es fácil que cualquiera con conocimientos, cree y comparta un paquete que soluciona un problema en concreto.

ALGUNOS DE LOS **PAQUETES** MÁS UTILIZADOS

Express

Nos permite crear nuestro servidor web, ya sea API Rest, o web renderizando nuestro servidor desde back.

React

Una de las librerías más utilizadas a la hora de crear Single Page Application

Vue

Una nueva experiencia a la hora de crear Single Page Applications. Muy recomendable

Babel

Nos permite transpilar o transformar nuestro código de la última versión de ECMAScript a una versión de JS que entienda el navegador o servidor

Lodash

Un conjunto de funciones y helpers que nos ayudan a desarrollar más rápido, y mejor

Webpack

Gestor de tareas automáticas, y generador de bundles para nuestra aplicación



2.

Ecosistema!

JAVASCRIPT

Y nada más !

¿En serio?

Pues si !



+2.760.000.000

Descarga de paquetes POR SEMANA !



Gracias!

Preguntas?

Podeis encontrarme en twitter @jmanuel_rosa

Ecmascript

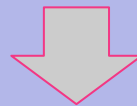
Nuevas features



CONSTANTES Y LET

```
const constante = 10
```

```
let variable = 15
```

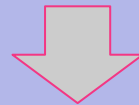


```
var constante = 10;
```

```
var variable = 15;
```

TEMPLATE STRING

```
const lugar = 'Cádiz'  
const NOMBRE = 'José Manuel'  
  
let frase = `Soy ${nombre} y nací en ${lugar}`
```

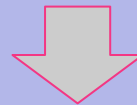


```
var lugar = 'Cádiz';  
var NOMBRE = 'José Manuel';  
  
var frase = 'Soy ' + nombre + ' y nac\xED en ' +  
lugar;
```

MEJORAS EN LOS OBJETOS

```
const lugar = 'Cádiz'

const datos = {
  nombre: 'José Manuel',
  edad: 35,
  lugar,
  random() {
    return 'randomNumber'
  }
}
```

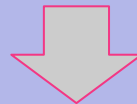


```
var lugar = 'Cádiz';

var datos = {
  nombre: 'José Manuel',
  edad: 35,
  lugar: lugar,
  random: function random() {
    return 'randomNumber';
  }
};
```

ARROW FUNCTIONS

```
const fn = (item) => {  
  console.log(item)  
  return item  
}  
  
const otraFn = item => console.log(item)
```



```
var fn = function fn(item) {  
  console.log(item);  
  return item;  
};  
  
var otraFn = function anotherFn(item) {  
  return console.log(item);  
};
```

DEFAULT PARAMETERS

```
function miFn(x, y, w = 10, h = 10) {  
  return x + y + w + h;  
}
```

```
miFn(1, 2) === 23
```



```
function miFn(x, y) {  
  var w = arguments.length > 2 && arguments[2]  
    !== undefined ? arguments[2] : 10;  
  var h = arguments.length > 3 && arguments[3]  
    !== undefined ? arguments[3] : 10;  
  
  return x + y + w + h;  
}
```

```
miFn(1, 2) === 23;
```

DESTRUCTURING ARRAY

```
var [a, , b] = [1, 2, 3];  
a === 1;  
b === 3;
```



```
var _ref = [1, 2, 3],  
    a = _ref[0],  
    b = _ref[2];  
  
a === 1;  
b === 3;
```

DESTRUCTURING EN FUNCIONES

```
function miFn({ nombre: x }) {  
  console.log(x);  
}  
  
miFn({ nombre: 5, random: 15 })
```



```
function miFn(_ref2) {  
  var x = _ref2.nombre;  
  
  console.log(x);  
}  
  
miFn({ nombre: 5, random: 15 });
```


DEFAULT PARAMETERS Y DESTRUCTURING

```
function miFn({ x, y, w = 10, h = 10 }) {  
  return x + y + w + h;  
}
```

```
miFn({ x: 1, y: 2 }) === 23
```

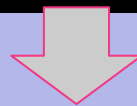


```
function miFn(_ref) {  
  var x = _ref.x,  
      y = _ref.y,  
      _ref$w = _ref.w,  
      w = _ref$w === undefined ? 10 : _ref$w,  
      _ref$h = _ref.h,  
      h = _ref$h === undefined ? 10 : _ref$h;  
  
  return x + y + w + h;  
}
```

```
miFn({ x: 1, y: 2 }) === 23;
```

ASYNC/AWAIT

```
function miFn () {  
  return miPromesa  
    .then(data => {  
      ....  
    })  
    .catch(data => {  
      ....  
    })  
}
```



```
async function miFn () {  
  try {  
    const data = await miPromesa  
    return data  
  } catch (error) {  
    return error  
  }  
}
```



Gracias!

Preguntas?

Podeis encontrarme en twitter @jmanuel_rosa

Babel



1.

BABEL

¿Qué es?
¿Para qué sirve?

A large, stylized, light gray double quote icon consisting of two separate, slightly curved strokes.

The compiler for writing next generation
JavaScript

TRANSFORMA ES6 A JS

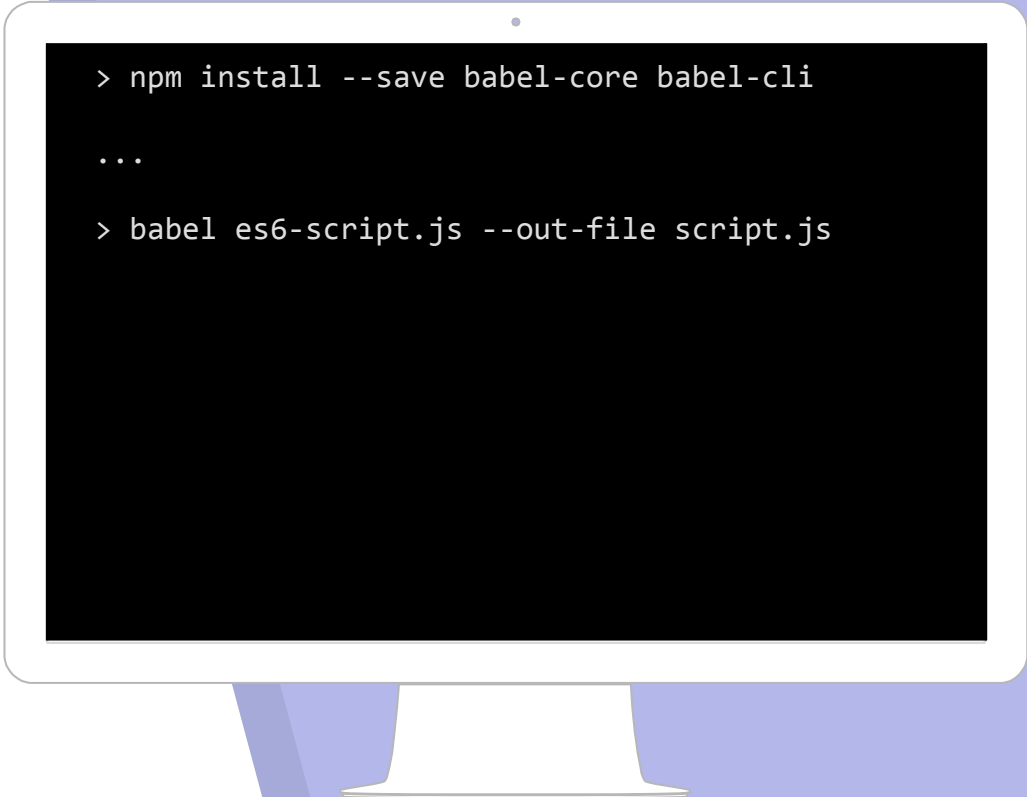
<https://babeljs.io/>



BABEL

INSTALACIÓN

Para poder instalar Babel, debemos tener instalado antes NodeJS, en una versión $\geq 6.x$, preferiblemente.



```
> npm install --save babel-core babel-cli  
  
...  
  
> babel es6-script.js --out-file script.js
```


SE BASA EN PRESETS

- ▶ env
- ▶ es2015
- ▶ es2016
- ▶ es2017 latest (deprecated in favor of env)
- ▶ react
- ▶ flow

Además, tiene los presets **experimentales Stage-X**

CONFIGURACIÓN

- ▶ .babelrc
- ▶ package.json
- ▶ env option



2.

Vamos a probar!



Gracias!

Preguntas?

Podeis encontrarme en twitter @jmanuel_rosa

MongoDB





1.

Qué es?

NoSQL

Base de datos orientada a documentos

NO ES NECESARIO SEGUIR UN ESQUEMA

Los documentos de una misma colección
pueden seguir esquemas distintos

POTENTE!

MongoDB esta escrita en C++



2.

Mongoose

ODM PARA NODEJS

Permite modelar datos de Mongo, para utilizarlos en nuestra aplicación, ofreciendo una interfaz de fácil uso

ES LA SOLUCIÓN MÁS UTILIZADA

Aunque hay otras alternativas, como el uso del driver nativo de NodeJS para Mongo

3.

Veamos un ejemplo!



Gracias!

Preguntas?

Podeis encontrarme en twitter @jmanuel_rosa

PostgreSQL





1.

Qué es?

RELACIONAL

Una de las bases de datos relacionales más potentes del mercado

LIBRE!

La licencia de PostgreSQL es Open Source!

RELACIONAL

Una de las bases de datos relacionales más
potentes del mercado

**DISPONIBLE EN
CUALQUIER
SISTEMA
OPERATIVO!**

Linux, Windows, Mac, Solaris,

SQL ESTÁNDAR !

Sigue el estándar de SQL!



2.

Sequelize

THE NODEJS ORM

Permite modelar datos de algunas de las bases de datos más conocidas como PostgreSQL, MySQL, SQLite, ...

A DESARROLLAR!

Utilizando un ORM, nos olvidamos un poco de las cuestiones propias de nuestra base de datos, ya que ofrece una interfaz común dando “igual” la base de datos que usemos.

3.

Veamos un ejemplo!



Gracias!

Preguntas?

Podeis encontrarme en twitter @jmanuel_rosa

¿Qué es REST?



REPRESENTATIONAL STATE TRANSFER

Y esto qué significa?

“

REST es cualquier interfaz entre sistemas que use HTTP para obtener datos o generar operaciones sobre esos datos en todos los formatos posibles, como XML y JSON

ALTERNATIVA A SOAP

Normal! Es muchísimo más sencilla !

¿POR QUÉ?

- ▶ Cada petición HTTP contiene toda la información necesaria
- ▶ Se apoya en HTTP
- ▶ POST, GET, PUT y DELETE
- ▶ Los objetos se manipulan a partir de URI
- ▶ Uso de Hypermedia

Existen 3 niveles de implementación de REST.

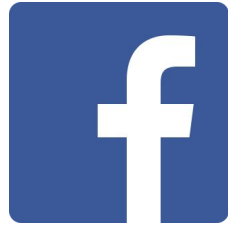
JSON!

La idea, es que por simplicidad, eficacia y tamaño, las URLs de nuestra Rest siempre devuelvan objetos JSON (aunque podríamos devolver XML)

VENTAJAS

- ▶ Separación entre cliente y servidor
- ▶ Visibilidad y escalabilidad
- ▶ Independiente de la plataforma
- ▶ Fácil de dividir en microservicios

¿QUIEN USA REST?



Google



LinkedIn

NETFLIX



Gracias!

Preguntas?

Podeis encontrarme en twitter @jmanuel_rosa