



# Curso de NodeJS

Unidad Didáctica 02: Gestión de Paquetes NPM



Ayuntamiento  
de Vitoria-Gasteiz  
Vitoria-Gasteizko  
Udala

# Índice de contenidos

- Introducción
- Información y Búsqueda
- Instalación
- Desinstalación
- Global vs Local
- Actualización
- Conclusiones

<http://cursosdedesarrollo.com/>



# Introducción

¿Qué es NPM?



# Introducción

Node Package Manager permite la descarga y gestión de recursos Javascript (y de otros tipos) de una manera sencilla



# Introducción

NPM permite la publicación pública o privada de paquetes descargables, actualizables y versionables



# Introducción

La instalación de NPM se realiza conjunto con el instalador de NodeJS



# Introducción

Para comprobar que está correctamente instalado desde una terminal hay que ejecutar:

```
$ npm --version
```

```
salida: 3.10.10
```



# Introducción

Para actualizar npm hay que ejecutar:

```
$ sudo npm install npm@latest -g
```





# Introducción

Después de la actualización

```
$ npm --version
```

salida: 4.4.1



# Introducción

Para empezar a trabajar con los comandos de npm siempre es mejor iniciar el directorio principal del proyecto que queremos desarrollar con

```
npm init
```



# Información y Búsqueda

Una vez instalado NPM ya podemos empezar a buscar paquetes que estén disponibles a través de NPM



# Información y Búsqueda

Herramientas de búsqueda:

<https://npmsearch.com/>

<http://nipstr.com/>

<http://cursosdedesarrollo.com/>



# Información y Búsqueda

npm search paquete

Ejemplo:

npm search express



# Información y Búsqueda

Una vez localizado el nombre del paquete podemos buscar la información referente al mismo:

```
npm info paquete
```

ejemplo

```
npm info express
```



# Información y Búsqueda

La información que nos dará es:

tags de distribución: latest

versiones disponibles

autores, mantenedores, contribuidores

licencia

repositorio

dependencias

etc...

<http://cursosdedesarrollo.com/>



# Instalación

Para instalar un determinado paquete tenemos que ejecutar el comando

```
npm install paquete
```

ejemplo

```
npm install express
```





# Instalación

Esto descargará el paquete o paquetes seleccionados en el directorio donde se haya ejecutado el comando npm



# Instalación

Se creará una carpeta llamada node\_modules donde se descargará el paquete seleccionado y las dependencias de funcionamiento de dicho paquete



# Instalación

También se puede hacer una instalación global del paquete

```
npm install paquete -g
```

ejemplo

```
npm install express -g
```



# Instalación

También se puede especificar la versión concreta del paquete a descargar:

```
npm install paquete@version
```

ejemplo

```
npm install express@4.15.2
```

```
npm install express@latest
```



# Instalación

También se puede indicar descargarlo de un repositorio git:

```
npm install https://github.com/<user>/<repo>/
```

```
npm install https://github.com/expressjs/express
```

```
npm install expressjs/express
```

```
npm i expressjs/express
```



# Instalación

Se puede indicar que al instalar un paquete se guarde su dependencia con

`--save`

`--save-dev`

`npm install express --save`



# Instalación

`npm ls`

presenta los paquetes instalados actualmente en un determinado directorio



# Instalación

El algoritmo de instalación hace los siguientes pasos:

Carga el árbol de dependencias desde el node\_modules

Clona el árbol

Coge el packages.json y sus metadatos y lo clona

sobre el clon va añadiendo las dependencias que falten

Las dependencias se colocarán lo más arriba posible sin romper otros módulos

Compara el árbol original con el clonado y hace una lista de las acciones que debería hacer para que fueran iguales

Ejecuta todas las acciones pendientes, empezando por la más profunda ( por lo tanto la más simple)

Las acciones serán install, update, remove y move

<http://cursosdedesarrollo.com/>





# Desinstalación

Para desinstalar un paquete deberemos ejecutar:

```
npm uninstall paquete
```

ó

```
npm uninstall paquete -g
```



# Global vs Local

Existen dos maneras de realizar una instalación:

Global: disponible para todos los proyectos

Local: Versiones descargadas para un proyecto concreto



# Global vs Local

La ventaja principal de usar una instalación global es que una vez instalada globalmente se puede enlazar al directorio del proyecto sin la necesidad de descargar de nuevo dicha versión otra vez localmente



# Global vs Local

npm link paquete

ejemplo

npm link express



# Global vs Local

Aunque el enlazado de paquete globales puede ser útil se desaconseja su utilización en producción



# Global vs Local

Para desenlazar puede usarse:

```
npm unlink paquete
```

```
npm unlink express
```



# Actualización

Siempre vamos a poder actualizar un determinado paquete:

```
npm update paquete
```

```
npm update paquete -g
```

ejemplo

```
npm update express -g
```



# Conclusiones

Hemos visto cómo buscar,  
instalar y actualizar las  
dependencias de un  
proyecto con NPM





# Datos de Contacto

<http://www.cursosdedesarrollo.com>  
[david@cursosdedesarrollo.com](mailto:david@cursosdedesarrollo.com)

<http://cursosdedesarrollo.com/>



# Licencia



David Vaquero Santiago

Esta obra está bajo una  
Licencia Creative Commons  
Atribución-NoComercial-  
CompartirIgual 4.0 Internacional

