

# Jhipster: Herramienta para crear desarrollo web en java

Conoce Jhipster, una herramienta de generación de proyectos que te ayudará a configurar tu desarrollo Java de una manera rápida usando las últimas tecnologías en desarrollo web

## Compártelo:



Hola, buen día en esta ocasión hablaremos sobre una herramienta de **desarrollo web**, más específicamente para el desarrollo de aplicaciones web basadas en la tecnología Java, esta herramienta se conoce como Jhipster el cual es generador de proyectos de cierta estructura que vas definiendo tú

---

¿Cuantas veces o mejor dicho cuento tiempo dedicamos nosotros a levantar la arquitectura de proyecto de **desarrollo web**, específicamente de una aplicación web que corra en Java?, No sé ustedes, pero en nuestra experiencia y con los proyectos en los que hemos participado es una etapa que nos consume como mínimo un periodo de entre dos y dos semanas y media (partiendo del hecho de que hablamos de un desarrollo relativamente sencillo), claro que esta es una de las etapas del **desarrollo web** en la que es necesario invertir tiempo y esfuerzo ya que si no creamos y configuramos nuestra aplicación de la manera correcta más adelante nos puede costar mucho en tiempo y recursos.

Es aquí en donde entra *Jhipster*, pero seguramente se preguntan ¿Qué es y para qué sirve Jhipster?, bueno *Jhipster* como les mencionaba es una herramienta que nos ayuda a crear la estructura de un proyecto Java con ciertas tecnologías específicas que en la actualidad se están usando mucho para el **desarrollo web**, en específico para la plataforma Java, y que a través de una serie de pasos se crea el proyecto con la estructura y funcionalidades básicas que necesitas para arrancar lo más pronto posible tu proyecto y así reducir considerablemente el tiempo que inviertes en la configuración de un proyecto, un ejemplo si tu te tardas 1 semana en crear toda la configuración necesaria para echar a andar tu proyecto ahora con *Jhipster* lo puedes hacer en una hora.

*Jhipster* hace uso de ciertas tecnologías y patrones muy utilizados en la actualidad como *Spring*, *Spring Boot*, *Spring Security*, *Hibernate*, *Angular 1*, *Angular 2*, *Bootstrap*, *Docker*, etc., y divide el proyecto en dos ramas importantes el back-end y el front-end (ambas igual de personalizables dependiendo tus necesidades).

Vayamos paso a paso a través de una serie de pasos e imágenes en las que podamos apreciar con mejor detalle todo lo que podemos hacer con *Jhipster*, hay que aclarar que es necesario tener instalado todo lo siguiente antes de hacer uso de *jhipster*:

- Nodejs preferentemente en su versión 7.8 o superior
- Npm versión 4.2.0 o superior
- Java 8
- Git
- Yarn (el cual es un manejador de dependencias)
- Yeoman (el cual es un generador de proyecto que toma como base *jhipster*)
- Algún gestor de base de datos tanto sql como nosql (ambos los soporta *jhipster*)

## Desarrollo Web con **Jhipster**

como instalarlo en otros S.O.

```
Marcos-MacBook-Pro:jhipster marco$ brew install yarn
```

El segundo paso será instalar Yeoman, es aquí en donde hacemos uso de nodejs y npm, lo instalaremos abriendo una consola e ingresando lo siguiente (Yeoman al instalarlo de manera global les pedirá que otorguen permisos de administrador):

```
Marcos-MacBook-Pro:jhipster marco$ sudo npm install -g yo
```

El tercer paso que vamos a realizar es la instalación de Jhipster, la cual realizaremos haciendo uso de yarn, lo que tenemos que hacer es abrir una consola e ingresar el siguiente comando para instalar Jhipster:

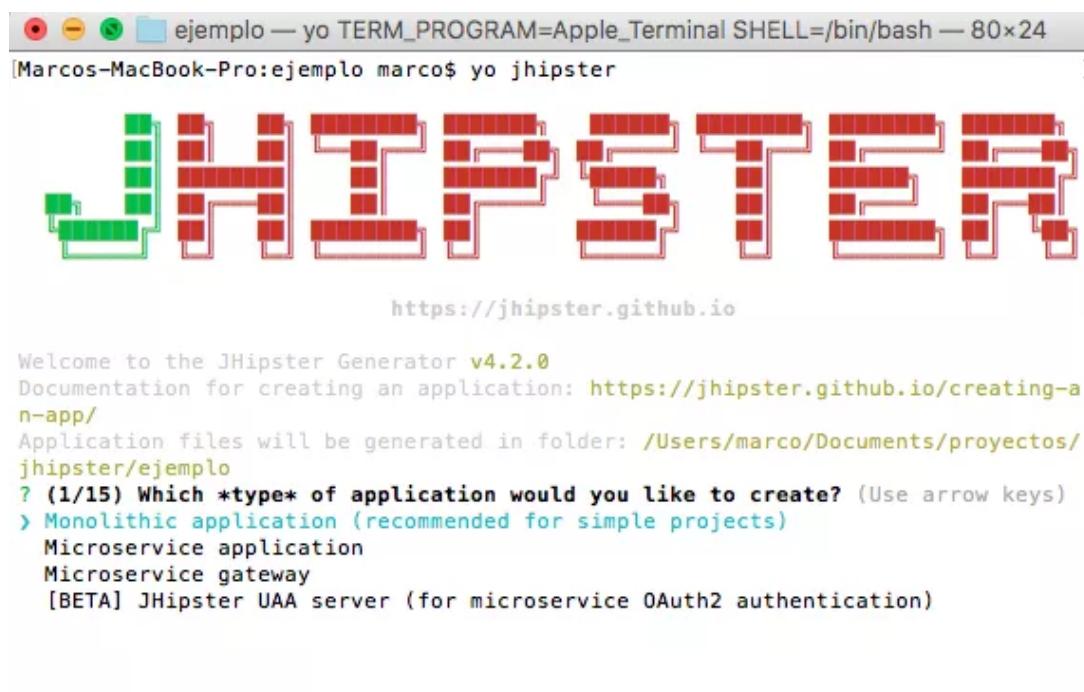
```
Marcos-MacBook-Pro:jhipster marco$ yarn global add generator-jhipster
```

Una vez realizado estos pasos lo siguiente será a través de una terminal crear una carpeta y entrar en ella, en esta carpeta es en donde se instalará nuestro proyecto que crearemos con jhipster, esto lo hacemos de la siguiente manera, (en s.o. linux y macOS):

## **mkdir ejemplo && cd ejemplo**

Una vez que hemos creado nuestra carpeta y estamos dentro de ella en una terminal ingresaremos el siguiente comando para comenzar la creación de nuestro proyecto:

Lo primero que nos va a aparecer es una pantalla con la siguiente información, en la cual nos va a preguntar que tipo de aplicación vamos a crear, nos da a elegir entre 4 diferentes opciones, en este caso para fines del ejemplo escogeremos la primera opción en la cual va a crear la estructura para un proyecto simple



A screenshot of a Mac OS X terminal window titled "ejemplo — yo TERM\_PROGRAM=Apple\_Terminal SHELL=/bin/bash — 80x24". The window shows the JHipster generator's welcome screen and a list of application types. The JHipster logo is displayed at the top, followed by the URL "https://jhipster.github.io". The text in the terminal is as follows:

```
[Marcos-MacBook-Pro:ejemplo marco$ yo jhipster
]
[  JHIPSTER
[  https://jhipster.github.io
Welcome to the JHipster Generator v4.2.0
Documentation for creating an application: https://jhipster.github.io/creating-a
n-app/
Application files will be generated in folder: /Users/marco/Documents/proyectos/
jhipster/ejemplo
? (1/15) Which *type* of application would you like to create? (Use arrow keys)
> Monolithic application (recommended for simple projects)
  Microservice application
  Microservice gateway
  [BETA] JHipster UAA server (for microservice OAuth2 authentication)
```

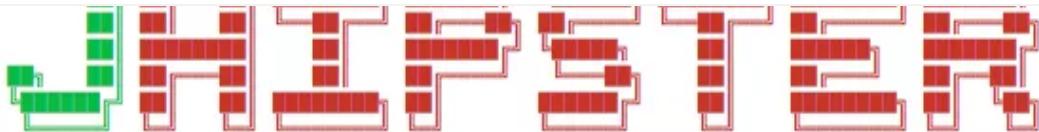
La siguiente pregunta que nos va a realizar es como se va llamar nuestro proyecto, por default tiene el mismo nombre de la carpeta en la que nos encontramos, aunque lo podemos cambiar a nuestras necesidades



La siguiente pregunta que nos hace es si queremos instalar otros generadores desde el Jhipster Marketplace, en esta ocasión le diremos que no para hacer la configuración más simple posible



El siguiente paso es ingresar el nombre del paquete base de nuestra aplicación java, para el ejemplo usaremos el siguiente app.jhipster.ejemplo



<https://jhipster.github.io>

```
Welcome to the JHipster Generator v4.2.0
Documentation for creating an application: https://jhipster.github.io/creating-an-app
/
Application files will be generated in folder: /Users/marco/Documents/proyectos/jhipster/ejemplo
? (1/15) Which *type* of application would you like to create? Monolithic application
(recommended for simple projects)
[?] (2/15) What is the base name of your application? ejemplo
[?] (3/15) Would you like to install other generators from the JHipster Marketplace? No
[

? (4/15) What is your default Java package name? (com.mycompany.myapp) app.jhipster.ejemplo
```

En el quinto paso nos pregunta que método de autenticación nos gustaría utilizar en nuestra aplicación, en esta ocasión seleccionaremos la segunda opción que es la JWT (Json Web Token) que ahora se está volviendo muy popular en el **desarrollo web** para dejar de lado la ya un poco anticuada Http Session Authentication

```
ejemplo — yo TERM_PROGRAM=Apple_Terminal SHELL=/bin/bash — 87x29
at process._tickCallback (internal/process/next_tick.js:104:9)
Marcos-MacBook-Pro:ejemplo marco$ clear

Marcos-MacBook-Pro:ejemplo marco$ yo jhipster

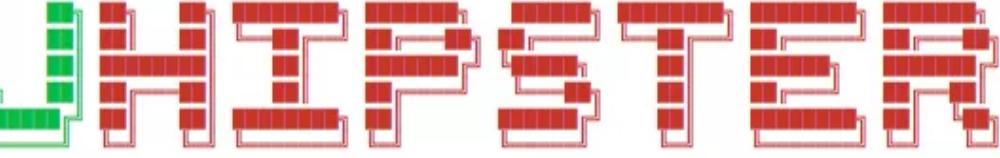
[ JHIPSTER

[ https://jhipster.github.io

Welcome to the JHipster Generator v4.2.0
Documentation for creating an application: https://jhipster.github.io/creating-an-app
Application files will be generated in folder: /Users/marco/Documents/proyectos/jhipster/ejemplo
? (1/15) Which *type* of application would you like to create? Monolithic application (recommended for simple projects)
? (2/15) What is the base name of your application? ejemplo
? (3/15) Would you like to install other generators from the JHipster Marketplace? No

? (4/15) What is your default Java package name? app.jhipster.ejemplo
? (5/15) Which *type* of authentication would you like to use?
  HTTP Session Authentication (stateful, default Spring Security mechanism)
  > JWT authentication (stateless, with a token)
  OAuth2 Authentication (stateless, with an OAuth2 server implementation)
```

En el sexto paso nos pregunta qué tipo de base de datos utilizaremos, en esta ocasión nos iremos por lo tradicional y elegiremos la primera opción que es la SQL, por supuesto que dependiendo de sus necesidades deberán usar la opción que necesiten de las que se muestran en la siguiente imagen

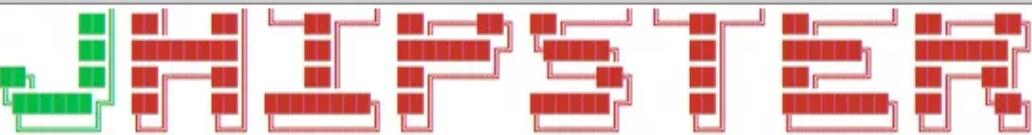


<https://jhipster.github.io>

```
Welcome to the JHipster Generator v4.2.0
Documentation for creating an application: https://jhipster.github.io/creating-an-app/
Application files will be generated in folder: /Users/marco/Documents/proyectos/jhipster/ejemplo
? (1/15) Which *type* of application would you like to create? Monolithic application (recommended for simple projects)
? (2/15) What is the base name of your application? ejemplo
? (3/15) Would you like to install other generators from the JHipster Marketplace? No

? (4/15) What is your default Java package name? app.jhipster.ejemplo
? (5/15) Which *type* of authentication would you like to use? JWT authentication (stateless, with a token)
? (6/15) Which *type* of database would you like to use? (Use arrow keys)
> SQL (H2, MySQL, MariaDB, PostgreSQL, Oracle, MSSQL)
  MongoDB
  Cassandra
```

En el séptimo paso nos preguntará que gestor de base de datos vamos a utilizar, en el ejemplo seleccionaremos MySQL, aunque en la siguiente imagen verán cómo pueden elegir entre más opciones, una de ellas es Oracle en la cual si eligen esta opción una vez finalizada la configuración tendrán que ingresar de manera manual el driver de Oracle



<https://jhipster.github.io>

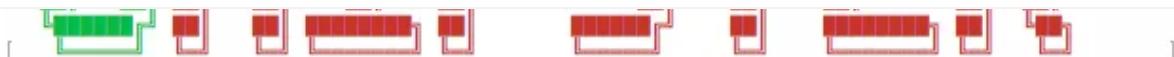
```
Welcome to the JHipster Generator v4.2.0
Documentation for creating an application: https://jhipster.github.io/creating-an-app/
Application files will be generated in folder: /Users/marco/Documents/proyectos/jhipster/ejemplo
? (1/15) Which *type* of application would you like to create? Monolithic application (recommended for simple projects)
? (2/15) What is the base name of your application? ejemplo
? (3/15) Would you like to install other generators from the JHipster Marketplace? No

? (4/15) What is your default Java package name? app.jhipster.ejemplo
? (5/15) Which *type* of authentication would you like to use? JWT authentication (stateless, with a token)
? (6/15) Which *type* of database would you like to use? SQL (H2, MySQL, MariaDB, PostgreSQL, Oracle, MSSQL)
? (7/15) Which *production* database would you like to use? (Use arrow keys)
> MySQL
  MariaDB
  PostgreSQL
  Oracle (Please follow our documentation to use the Oracle proprietary driver)
  Microsoft SQL Server
```

duro, la segunda opción se trata de H2 con persistencia en la memoria, en esta opción los datos se borrarán una vez que apaguemos o reiniciemos nuestra computadora y la tercera opción nos pone directamente el gestor de base de datos que elegimos en el paso anterior, en esta ocasión elegiremos el propio gestor para hacer pruebas directamente con el

```
[ejemplo] exemplo — yo TERM_PROGRAM=Apple_Terminal SHELL=/bin/bash — 87x29
[| JHipster
[| https://jhipster.github.io
[| Welcome to the JHipster Generator v4.2.0
[| Documentation for creating an application: https://jhipster.github.io/creating-an-app/
[| Application files will be generated in folder: /Users/marco/Documents/proyectos/jhipster/ejemplo
[| ? (1/15) Which *type* of application would you like to create? Monolithic application
[| (recommended for simple projects)
[| ? (2/15) What is the base name of your application? ejemplo
[| ? (3/15) Would you like to install other generators from the JHipster Marketplace? No
[| ? (4/15) What is your default Java package name? app.jhipster.ejemplo
[| ? (5/15) Which *type* of authentication would you like to use? JWT authentication (stateless, with a token)
[| ? (6/15) Which *type* of database would you like to use? SQL (H2, MySQL, MariaDB, PostgreSQL, Oracle, MSSQL)
[| ? (7/15) Which *production* database would you like to use? MySQL
[| ? (8/15) Which *development* database would you like to use?
[|   H2 with disk-based persistence
[|   H2 with in-memory persistence
> MySQL
```

Una vez elegimos cualquiera de las opciones anteriores nos preguntará si queremos que Hibernate tenga un segundo nivel de caché, en esta ocasión le diremos que no para hacer más sencilla la configuración, pero repito esto puede variar dependiendo de sus necesidades

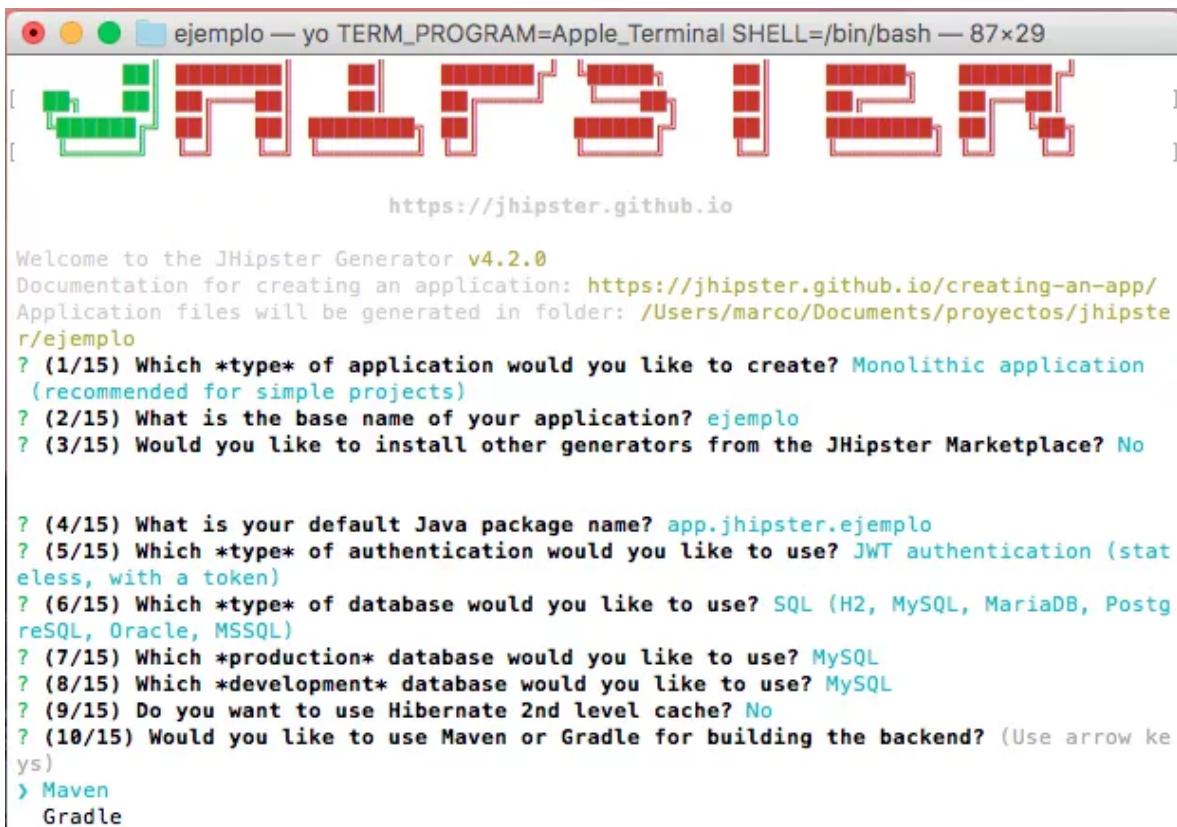


<https://jhipster.github.io>

```
Welcome to the JHipster Generator v4.2.0
Documentation for creating an application: https://jhipster.github.io/creating-an-app/
Application files will be generated in folder: /Users/marco/Documents/proyectos/jhipster/ejemplo
? (1/15) Which *type* of application would you like to create? Monolithic application (recommended for simple projects)
? (2/15) What is the base name of your application? ejemplo
? (3/15) Would you like to install other generators from the JHipster Marketplace? No

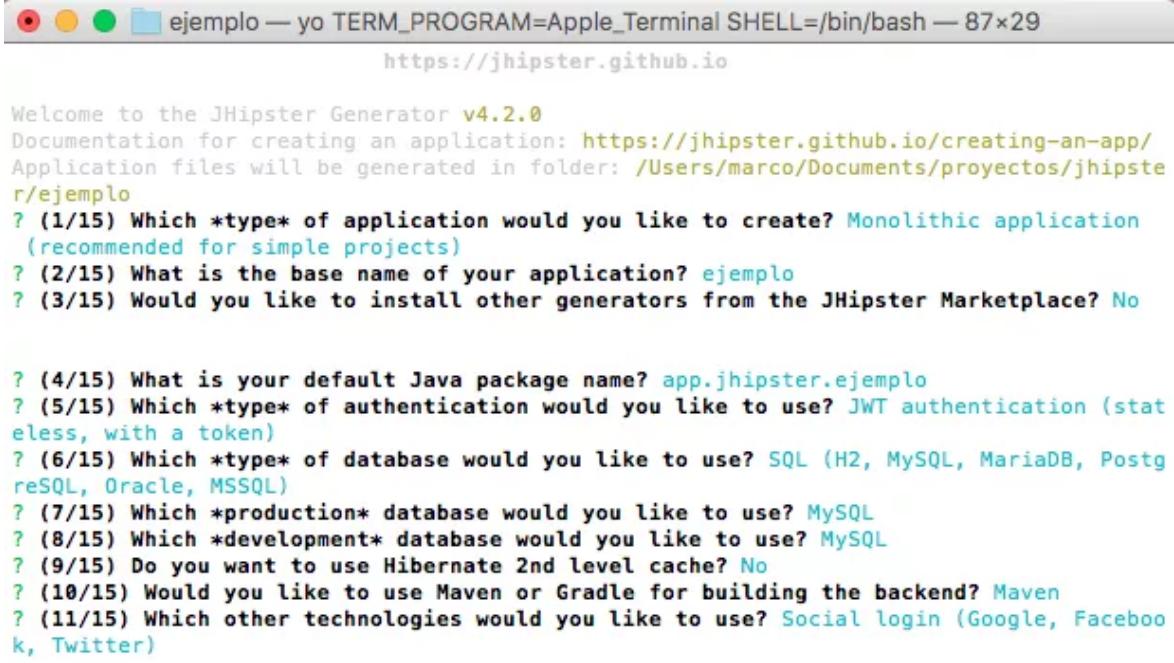
? (4/15) What is your default Java package name? app.jhipster.ejemplo
? (5/15) Which *type* of authentication would you like to use? JWT authentication (stateless, with a token)
? (6/15) Which *type* of database would you like to use? SQL (H2, MySQL, MariaDB, PostgreSQL, Oracle, MSSQL)
? (7/15) Which *production* database would you like to use? MySQL
? (8/15) Which *development* database would you like to use? MySQL
? (9/15) Do you want to use Hibernate 2nd level cache?
> No
    Yes, with ehcache (local cache, for a single node)
    Yes, with HazelCast (distributed cache, for multiple nodes)
```

En el paso número 10 de la configuración nos preguntará que herramienta de gestión de dependencias queremos usar en el back-end, en esta ocasión usaremos Maven, como se puede apreciar en la siguiente imagen



En el paso 11 nos pregunta si queremos utilizar otras tecnologías en nuestro **desarrollo web**, como puede ser la integración de un sistema de login con redes sociales, el uso de un motor de búsqueda como

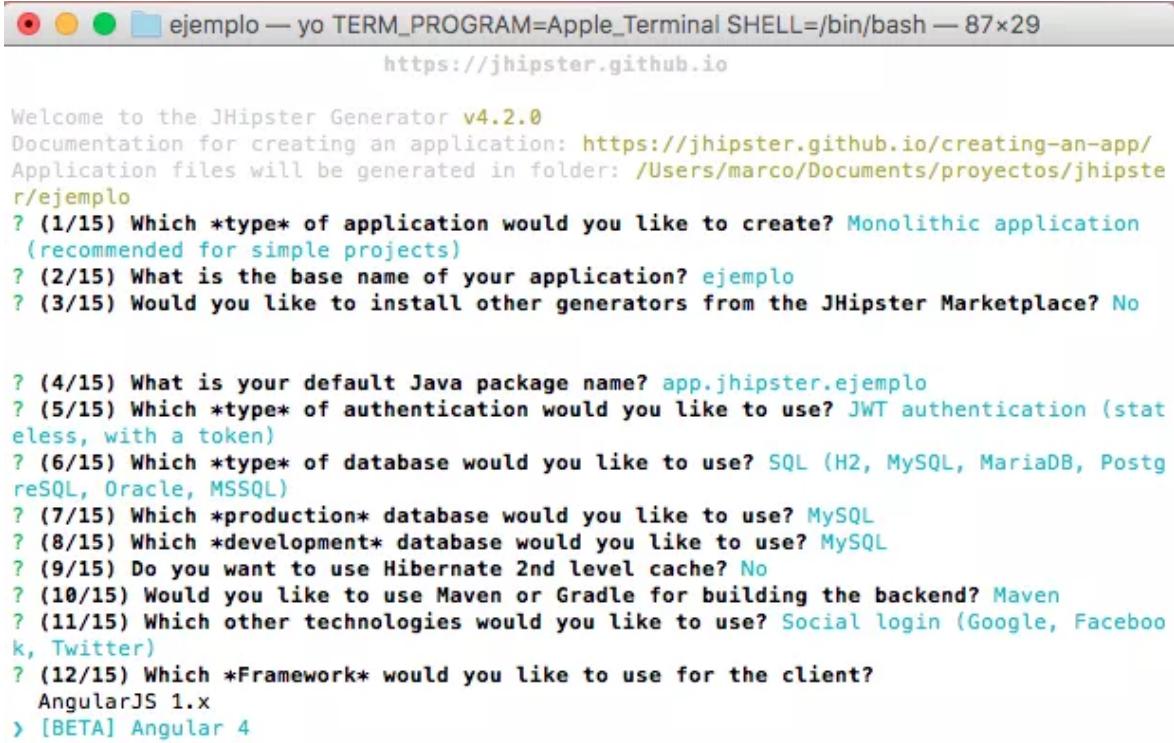
## login con redes sociales



```
Welcome to the JHipster Generator v4.2.0
Documentation for creating an application: https://jhipster.github.io/creating-an-app/
Application files will be generated in folder: /Users/marco/Documents/proyectos/jhipster/ejemplo
? (1/15) Which *type* of application would you like to create? Monolithic application (recommended for simple projects)
? (2/15) What is the base name of your application? ejemplo
? (3/15) Would you like to install other generators from the JHipster Marketplace? No

? (4/15) What is your default Java package name? app.jhipster.ejemplo
? (5/15) Which *type* of authentication would you like to use? JWT authentication (stateless, with a token)
? (6/15) Which *type* of database would you like to use? SQL (H2, MySQL, MariaDB, PostgreSQL, Oracle, MSSQL)
? (7/15) Which *production* database would you like to use? MySQL
? (8/15) Which *development* database would you like to use? MySQL
? (9/15) Do you want to use Hibernate 2nd level cache? No
? (10/15) Would you like to use Maven or Gradle for building the backend? Maven
? (11/15) Which other technologies would you like to use? Social login (Google, Facebook, Twitter)
```

En el siguiente paso que es el número 12 es uno de los más importantes ya que nos pregunta que versión de angular usaremos en nuestro proyecto, en este caso escogeré la versión de angular 2



```
Welcome to the JHipster Generator v4.2.0
Documentation for creating an application: https://jhipster.github.io/creating-an-app/
Application files will be generated in folder: /Users/marco/Documents/proyectos/jhipster/ejemplo
? (1/15) Which *type* of application would you like to create? Monolithic application (recommended for simple projects)
? (2/15) What is the base name of your application? ejemplo
? (3/15) Would you like to install other generators from the JHipster Marketplace? No

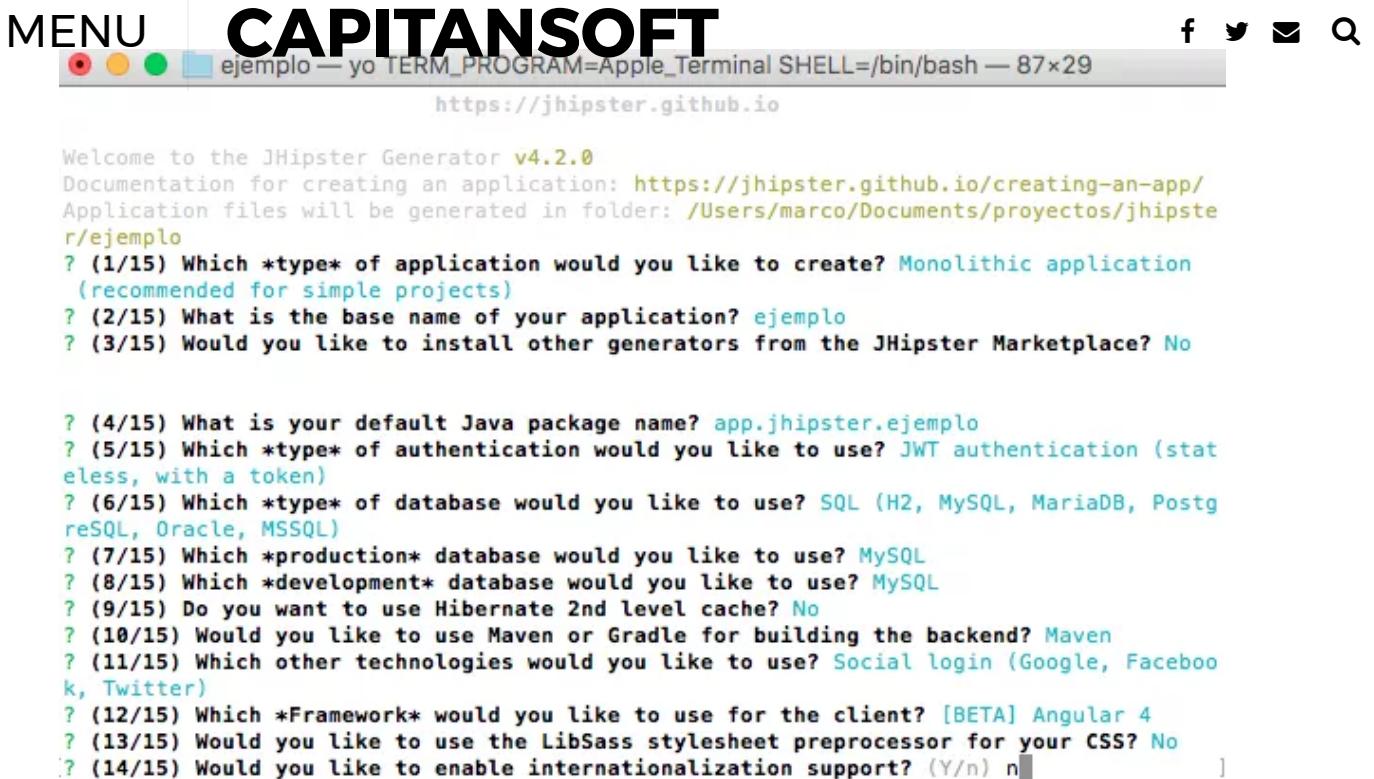
? (4/15) What is your default Java package name? app.jhipster.ejemplo
? (5/15) Which *type* of authentication would you like to use? JWT authentication (stateless, with a token)
? (6/15) Which *type* of database would you like to use? SQL (H2, MySQL, MariaDB, PostgreSQL, Oracle, MSSQL)
? (7/15) Which *production* database would you like to use? MySQL
? (8/15) Which *development* database would you like to use? MySQL
? (9/15) Do you want to use Hibernate 2nd level cache? No
? (10/15) Would you like to use Maven or Gradle for building the backend? Maven
? (11/15) Which other technologies would you like to use? Social login (Google, Facebook, Twitter)
? (12/15) Which *Framework* would you like to use for the client?
AngularJS 1.x
> [BETA] Angular 4
```

En el paso 13 nos pregunta si queremos hacer uso del preprocesador de CSS Sass en nuestro proyecto, en esta ocasión le diré que no y seguiremos con nuestra configuración

```
Documentation for creating an application: https://jhipster.github.io/creating-an-app/
Application files will be generated in folder: /Users/marco/Documents/proyectos/jhipster/ejemplo
? (1/15) Which *type* of application would you like to create? Monolithic application (recommended for simple projects)
? (2/15) What is the base name of your application? ejemplo
? (3/15) Would you like to install other generators from the JHipster Marketplace? No

? (4/15) What is your default Java package name? app.jhipster.ejemplo
? (5/15) Which *type* of authentication would you like to use? JWT authentication (stateless, with a token)
? (6/15) Which *type* of database would you like to use? SQL (H2, MySQL, MariaDB, PostgreSQL, Oracle, MSSQL)
? (7/15) Which *production* database would you like to use? MySQL
? (8/15) Which *development* database would you like to use? MySQL
? (9/15) Do you want to use Hibernate 2nd level cache? No
? (10/15) Would you like to use Maven or Gradle for building the backend? Maven
? (11/15) Which other technologies would you like to use? Social login (Google, Facebook, Twitter)
? (12/15) Which *Framework* would you like to use for the client? [BETA] Angular 4
? (13/15) Would you like to use the LibSass stylesheet preprocessor for your CSS? (y/N)
n
```

En el paso 14 nos pregunta si queremos que nuestro proyecto tenga la disponibilidad de poder verlo en diferentes idiomas, en este caso le diremos que no, aunque si ustedes lo necesitan Jhipster nos permite escoger entre una lista de idiomas en los cuales podrá verse el front-end de nuestro proyecto



Y por último en el paso 15 nos pregunta que con que framework de testing queremos probar nuestro proyecto, en este ejemplo usaremos Cucumber, aunque ustedes pueden elegir el que más les guste o se

```
Welcome to the JHipster Generator v4.2.0
Documentation for creating an application: https://jhipster.github.io/creating-an-app/
Application files will be generated in folder: /Users/marco/Documents/proyectos/jhipster/ejemplo
? (1/15) Which *type* of application would you like to create? Monolithic application (recommended for simple projects)
? (2/15) What is the base name of your application? ejemplo
? (3/15) Would you like to install other generators from the JHipster Marketplace? No

? (4/15) What is your default Java package name? app.jhipster.ejemplo
? (5/15) Which *type* of authentication would you like to use? JWT authentication (stateless, with a token)
? (6/15) Which *type* of database would you like to use? SQL (H2, MySQL, MariaDB, PostgreSQL, Oracle, MSSQL)
? (7/15) Which *production* database would you like to use? MySQL
? (8/15) Which *development* database would you like to use? MySQL
? (9/15) Do you want to use Hibernate 2nd level cache? No
? (10/15) Would you like to use Maven or Gradle for building the backend? Maven
? (11/15) Which other technologies would you like to use? Social login (Google, Facebook, Twitter)
? (12/15) Which *Framework* would you like to use for the client? [BETA] Angular 4
? (13/15) Would you like to use the LibSass stylesheet preprocessor for your CSS? No
[?] (14/15) Would you like to enable internationalization support? No
? (15/15) Besides JUnit and Karma, which testing frameworks would you like to use?
  o Gatling
  o Cucumber
  o Protractor
```

Después de toda esta serie de preguntas de configuración Jhipster comenzará a crear nuestro proyecto con base en las especificaciones que le hemos dado, este proceso puede tardar algunos minutos, pero como se habrán dado cuenta es muy fácil y rápido el poder crear un proyecto con Jhipster, el cual nos ahorrará muchísimo tiempo de configuración el cual podremos invertir en el desarrollo de la funcionalidad de nuestro proyecto.

Como podemos ver ahora en el mundo del **desarrollo web** existen muchas opciones y herramientas que nos ayudarán a crear la base de nuestros proyectos de una forma rápida y confiable, una de estas opciones es **Jhipster**, en este caso hablando específicamente de la plataforma Java, en entradas posteriores estaremos hablando de algunos otros generadores de proyectos en diferentes plataformas, como **Nodejs**, **Python**, etc.

Esperemos les haya gustado este post y les sea de mucha utilidad, si ha sido así compártelo en las redes sociales, por email o como quieras a las personas que estén interesadas en el **desarrollo web**, si tienes alguna duda en cómo utilizar **Jhipster** escríbenos en los comentarios o en nuestras redes sociales y con mucho gusto te ayudaremos, nos vemos en el siguiente post.

Jhipster: <https://jhipster.github.io/>

Yeoman: <http://yeoman.io/>

Yarn: <https://yarnpkg.com/lang/en/>

#### Compartelo:



#### Relacionado

[Certificaciones Java, ¿Cuáles son?](#)

abril 6, 2017

En "Java"

[Desarrollo de software en un solo](#)

[lugar, Bienvenida al blog](#)

marzo 28, 2017

En "Sin categoría"

[Angular 4 esta entre nosotros,](#)

[conoce que hay de nuevo](#)

abril 18, 2017

En "Angular"

Tags

[Angular](#)

[Bootstrap](#)

[Hibernate](#)

[java](#)

[Jhipster](#)

[MVC](#)

[Spring](#)

– Avertisement –



Global Versión Xiaomi  
mi A...

**193,54 €**

[Shop now](#)



#### You might also like

**Netflix OSS la API para implementar microservicios que te salvará la vida**

Tecnología ⏲ 5 meses ago

**Angular 4 esta entre nosotros, conoce que hay de nuevo**

Angular ⏲ 5 meses ago

**Certificaciones Java, ¿Cuáles son?**

Java ⏲ 6 meses ago

## Leave a Reply

Your email address will not be published. Required fields are marked \*

*Comment*

Name \*

Email \*

Website

POST COMMENT

- Recibir un email con los siguientes comentarios a esta entrada.
- Recibir un email con cada nueva entrada.















## NEWSLETTER

Suscríbete a este blog y recibe notificaciones de nuevas entradas.

Dirección de email

SUSCRIBIR

## SÍGUENOS EN FACEBOOK



Me gusta esta página

Compartir

Sé el primero de tus amigos en indicar que le gusta esto.



## SÍGUENOS EN TWITTER

Tweets por @CapitanSoft



Netflix OSS la API para implementar microservicios  
que te salvará la vida  
[capitansoft.com/blog/2017/05/1...](http://capitansoft.com/blog/2017/05/1...)



11 may. 2017



**Capitan Soft**  
@CapitanSoft

Arquitectura de Microservicios, olviate de la  
tradicional estructura monolítica  
[capitansoft.com/blog/2017/05/0...](http://capitansoft.com/blog/2017/05/0...)

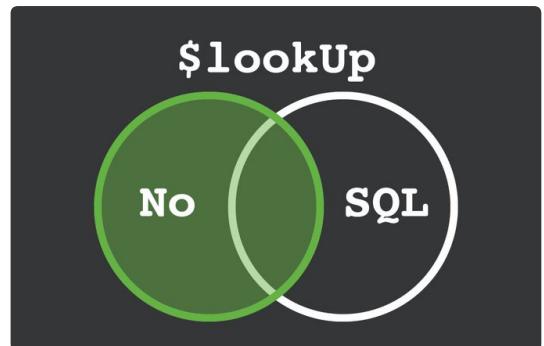


8 may. 2017



**Capitan Soft**  
@CapitanSoft

Base de datos noSQL, las bases de datos no  
volverán a ser las mismas  
[capitansoft.com/blog/2017/05/0...](http://capitansoft.com/blog/2017/05/0...)



3 may. 2017



**Capitan Soft**  
@CapitanSoft

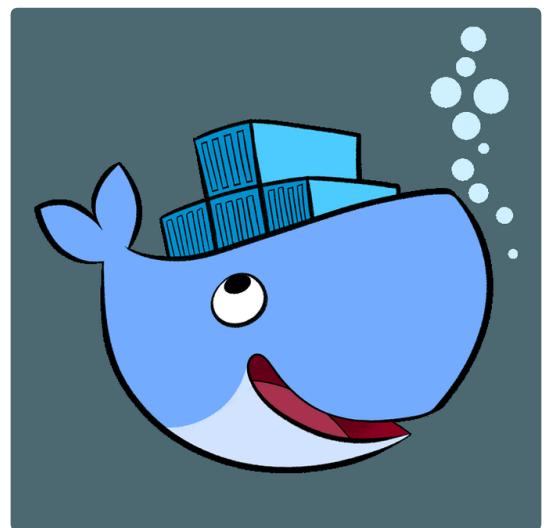
Certificaciones de Oracle, el camino a seguir  
[capitansoft.com/blog/2017/05/0...](http://capitansoft.com/blog/2017/05/0...)

---

2 may. 2017**Capitan Soft**

@CapitanSoft

Docker Tutorial de Instalación y creación de nuestro primer contenedor [capitansoft.com/blog/2017/04/2...](http://capitansoft.com/blog/2017/04/2...)

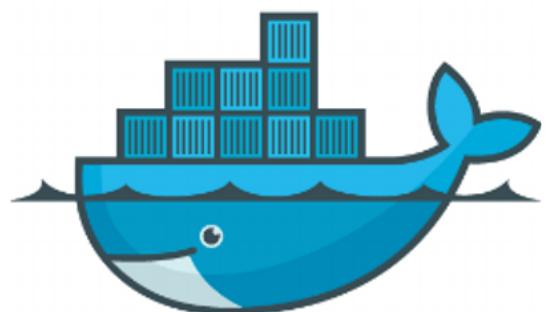


27 abr. 2017

**Capitan Soft**

@CapitanSoft

Docker, una opción contra las máquinas virtuales tradicionales  
[capitansoft.com/blog/2017/04/1...](http://capitansoft.com/blog/2017/04/1...)



19 abr. 2017

**Capitan Soft**

@CapitanSoft



18 abr. 2017

 **Capitan Soft**  
@CapitanSoft

Jhipster: Herramienta para crear desarrollo web en java [capitansoft.com/blog/2017/04/1...](http://capitansoft.com/blog/2017/04/1...)



12 abr. 2017

 **Capitan Soft**  
@CapitanSoft

Aplicaciones Móviles: Nativas vs Híbridas  
[capitansoft.com/blog/2017/04/0...](http://capitansoft.com/blog/2017/04/0...)



 @CapitanSoft

Desarrollo de software en un solo lugar, Bienvenida al blog [capitansoft.com/blog/2017/03/2...](http://capitansoft.com/blog/2017/03/2...)



28 mar. 2017

**Capitan Soft**

@CapitanSoft

Bienvenidos a CapitanSoft, noticias de desarrollo de software en un solo sitio [#CapitanSoft](#)

20 mar. 2017

[Insertar](#)[Ver en Twitter](#)

## ÚLTIMOS POSTS

- › Netflix OSS la API para implementar microservicios que te salvará la vida
- › Arquitectura de Microservicios, olviate de la tradicional estructura monolítica
- › Base de datos noSQL, las bases de datos no volverán a ser las mismas
- › Certificaciones de Oracle, el camino a seguir
- › Docker Tutorial de Instalación y creación de nuestro primer contenedor

– Avertisement –

## REDES SOCIALES



## SUSCRÍBETE AL BLOG

Introduce tu correo electrónico para suscribirte a este blog y recibir notificaciones de nuevas entradas.

*Dirección de email*

SUSCRIBIRSE