

[Inicio](#) | [Quiénes somos](#) | [Formación](#) | [Publicaciones](#)

Tutoriales.

[adictosaltrabajo](#) / [Tutoriales](#) / ¿Por qué todo el mundo habla de Jhipster?**Alberto Martínez**

Consultor tecnológico de desarrollo de proyectos informáticos.



¿Por qué todo el mundo habla de Jhipster?

octubre 19, 2016

[Alberto Martínez](#)[2 comentarios](#)[Tutoriales](#)[1208 visitas](#)

En este tutorial vamos a aprender el uso básico de jhipster y por qué la comunidad java tiene el punto de mira en él.

Índice de contenidos

- 1. Introducción
- 2. Entorno
- 3. ¿Que es Jhipster?
- 4. Requisitos previos
- 5. Comenzando con la instalación
- 6. Creando el proyecto
- 7. Comprobando y arrancando nuestra aplicación
- 8. Conclusiones
- 9. Referencias

1. Introducción

Ultimamente en la comunidad de desarrolladores se oye hablar mucho de Jhipster, y en este artículo voy a hacer una pequeña introducción sobre que es Jhipster, en que se basa y vamos a hacer un pequeño ejemplo para ver como funciona.

2. Entorno

El tutorial está escrito usando el siguiente entorno:

Uso de cookies

Este sitio web utiliza cookies para que usted tenga la mejor experiencia de usuario. Si continúa navegando está dando su consentimiento para la aceptación de las mencionadas cookies y la aceptación de nuestra política de cookies, pinche el enlace para mayor información

[ACEPTAR](#)

Los más visitados de la semana

[Page Analytics](#)[Instalación y uso del plugin de comentarios de Facebook en n...](#)
[Monitorizando equipos y servicios con Nagios + NagiosQI + PN...](#)[Javassist: manipulando el bytecode de una aplicación Java.](#)
[GIF animado con Photoshop](#)

3. ¿Qué es Jhipster?

En la propia web del proyecto dicen: "Jhipster is a Yeoman generator, used to create a Spring Boot + AngularJS project" Y realmente es básicamente eso, un generador de proyectos basado en Yeoman que nos creará la base para continuar un proyecto basado en el siguiente stack de tecnologías:

Lado cliente:

Modelo SPA (Single page application):

- Responsive Web Design
- HTML5 Boilerplate
- Twitter Bootstrap
- AngularJS
- Compatible con IE9+ y navegadores modernos
- Soporte completo para internacionalización con Angular Translate
- Soporte opcional Sass para diseño CSS
- Soporte opcional WebSocket con Spring WebSocket

Con el framework de yeoman:

- Fácil instalación de nuevas librerías JavaScript con Bower
- Construcción, optimización y recarga en tiempo real con Gulp.js
- Testing con Karma y PhantomJS

¿Y si el modelo de aplicación de una sola página web no es suficiente para nuestras necesidades?

- Soporte para el motor de plantillas Thymeleaf, para generar páginas web en el lado del servidor

Lado del servidor:

A complete Spring application:

- Spring Boot para la configuración de la aplicación de forma sencilla
- Configuración con Maven o Gradle para la construcción, test y lanzamiento de la aplicación con perfiles de desarrollo y producción (out of the box).
- Spring Security
- Spring MVC REST + Jackson
- Soporte opcional WebSocket con Spring WebSocket
- Spring Data JPA + Bean Validation
- Actualizaciones de base de datos con Liquibase
- Soporte para Elasticsearch si deseas tener capacidades de búsqueda por encima de la base de datos
- Soporte para MongoDB si deseas usar una base de datos NoSQL orientada a documentos en vez de JPA.
- Soporte para Cassandra si deseas usar una base de datos NoSQL orientada a columnas en vez de JPA.

Preparado para ir a producción:

- Monitorización con Metrics
- Cacheo con ehcache (cache local) o hazelcast (cache distribuido)
- Clusterización de sesiones HTTP opcional con hazelcast
- Recursos estáticos optimizados (filtro gzip, HTTP headers cacheados)
- Gestión de logs con Logback, configurable en tiempo de ejecución
- Pooling de conexiones con HikariCP para un rendimiento óptimo
- Creación de un WAR estándar o un JAR ejecutable

Como podéis ver es realmente impresionante lo que se promete, así que vamos a ver como lo hace y vamos a pasar a realizar un pequeño ejemplo.

Tweets por @adictosaltrabaj



Archivos

Archivos

Elegir mes

4. Requisitos previos

Uso de cookies

Este sitio web utiliza cookies para que usted tenga la mejor experiencia de usuario. Si continúa navegando está dando su consentimiento para la aceptación de las mencionadas cookies y la aceptación de nuestra política de cookies, pinche el enlace para mayor información

ACEPTAR

- Git
- Node.js que también nos instala npm que será el gestor de paquetes que utilizaremos para el resto de la instalación

5. Comenzando con la instalación

Lo primero va a ser instalar yeoman con:

```
1 sudo npm install -g yo
```

Tras la instalación de yeoman, pasaremos a instalar Bower, Gulp y el propio Jhipster

```
1 sudo npm install -g bower
2 sudo npm install -g gulp
3 sudo npm install -g generator-jhipster
```

Tras instalar la infraestructura básica, ya estamos en disposición de comenzar con nuestro pequeño piloto de jhipster

6. Creando el proyecto

Lo primero que vamos a hacer es crear una carpeta donde tener nuestro proyecto con:

```
1 mkdir books_jhipster
```

Con esto ya tenemos nuestro directorio y creamos el esqueleto del proyecto con:

```
1 cd books_jhipster
2 yo jhipster
```

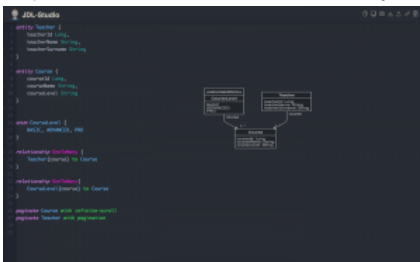
Lo que vamos a conseguir con esto es lanzar el asistente del generador de jhipster que nos hará unas preguntas para poder configurar de forma correcta nuestro proyecto.



Las preguntas, son bastante claras si sabes que quieres en tu aplicación, en todo caso en la web oficial de jhipster tenemos una descripción de todas y cada una de las opciones del generador. Ok, en un principio ya tenemos la base de nuestra aplicación. Ahora, vamos a añadir unas entidades al modelo de base de datos. Podemos hacerlo mediante un generador de entidades,

```
1 yo jhipster:entity [nombre_de_la_entidad]
```

Que al igual que el generador del proyecto, nos irá pidiendo información sobre la entidad para acabar creándola. De esta forma podríamos crear todas las entidades y relaciones, pero para este tutorial voy a utilizar una herramienta que también proporciona la gente de jhipster que es [JDL Studio](#), que nos permite crear un esquema de entidades de una forma muy sencilla. Vamos a verlo:



Como se puede ver he definido dos entidades y una enumeración, sus relaciones e indicado como quiere que se paginen dichas entidades dentro de la aplicación, ahora se exporta a un archivo ".jh" que es el que

Uso de cookies

Este sitio web utiliza cookies para que usted tenga la mejor experiencia de usuario. Si continúa navegando está dando su consentimiento para la aceptación de las mencionadas cookies y la aceptación de nuestra política de cookies, pinche el enlace para mayor información

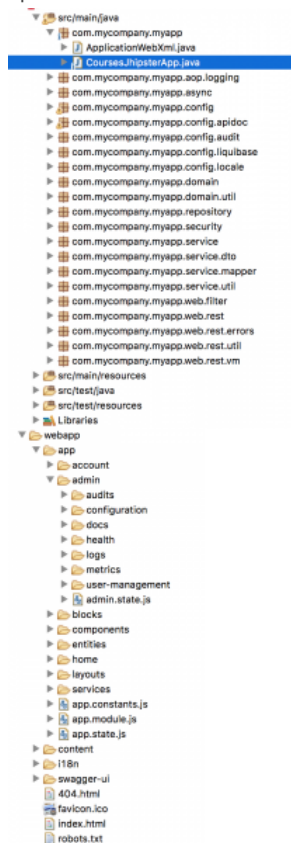
ACEPTAR

Jhipster también nos proporciona mecanismos para crear servicios cuando sea necesario añadir funcionalidad por encima de las necesidades básicas del CRUD con:

```
1 yo jhipster:service [nombre_del_servicio]
```

7. Comprobando y arrancando nuestra aplicación

Con todo lo hecho anteriormente, ya tenemos nuestra aplicación funcional para empezar a trabajar sobre ella, veamos que nos ha generado jhipster y como se arranca la aplicación. Jhipster nos genera todos los elementos básicos de la aplicación tanto del backend como del frontend y los deja preparados para que funcione la aplicación out of the box



Como se puede ver ya está todo preparado para lanzar nuestra aplicación, para lo cual utilizaremos el wrapper de maven que tenemos en el directorio de nuestro proyecto con el comando:

```
1 ./mvnw
```

Tras unos instantes ya tendremos nuestra aplicación preparada y dispuesta para ser utilizada



8. Conclusiones

Como hemos podido ver en este rápido tutorial, jhipster nos da la habilidad de tener aplicaciones funcionales con múltiples características que siempre nos gusta tener en nuestras aplicaciones como swagger para el testeo y documentación del api, metrics para el monitoreo de la aplicación... disponibles en muy poco tiempo. Además nos proporciona herramientas para el modelado de entidades, test de integración y unitarios pregenerados y muchas cosas más casi sin esfuerzo, hasta aquí bien. Ahora es cuando viene el pero (no demasiado grande). Lo que viene después es arduo (o no tanto), puesto que nos queda hacer la aplicación

Uso de cookies

Este sitio web utiliza cookies para que usted tenga la mejor experiencia de usuario. Si continúa navegando está dando su consentimiento para la aceptación de las mencionadas cookies y la aceptación de nuestra política de cookies, pinche el enlace para mayor información

ACEPTAR

una arquitectura básica que se cimienta en las últimas y más cacareadas tecnologías que existen en la actualidad. ¿Que os parece este enfoque?

9. Referencias

- [Página oficial de jhipster](#)

[angularjs](#)[fullstack](#)[jhipster](#)[Spring Boot](#)

Comparte este artículo!



Esta obra está licenciada bajo [licencia Creative Commons de Reconocimiento-No comercial-Sin obras derivadas 2.5](#)

Entradas relacionadas

[AngularJS: primeros pasos.](#) (febrero 21, 2013)

Respuestas (2)



Jesús Perales

octubre 20, 2016 at 1:40 am · Responder

Excelente, lo unico que me molesta es que no existe ningun tutorial que diga como hacer un login social, paso a paso con facebook, igual podrian hacer uno ! 😊



Nicolas

septiembre 8, 2017 at 9:47 pm · Responder

Llevo mas de un año utilizando esta tecnologia y la verdad que me cambio la vida, a veces tengo algunos problemas con liquibase, pero el resto es excelente, eh desarrollado aplicaciones en tiempo record y de excelente calidad! Aun me falta mucho por aprender, pero ya estoy muy avanzado con el framework y por el momento pienso seguir por este camino. Saludos, si tienen dudas puedo ayudar en lo que sea en cuanto a la utilizacion de JHipster.
Saludos.

Deja un comentario

Uso de cookies

Este sitio web utiliza cookies para que usted tenga la mejor experiencia de usuario. Si continúa navegando está dando su consentimiento para la aceptación de las mencionadas cookies y la aceptación de nuestra [política de cookies](#), pinche el enlace para mayor información

[ACEPTAR](#)

Lo último

Introducción a StencilJS y demo de integración con Angular
Javassist: manipulando el bytecode de una aplicación Java.
Firebase Cloud Messaging: mensajería gratuita en la nube
Disminuir tiempos de carga del Discover en Kibana 5
Gestión del estado en Angular sin ngrx

Datos de contacto

Edificio BestPoint Avd. de Castilla, 1, Planta 2, Oficina 21B (San Fernando de Henares)
Phone: 916 75 33 06
E-Mail: adictos@adictosaltrabajo.com
Web: <https://www.autentia.com>

Powered by



Copyright 2003-2016 © All Rights Reserved | Texto legal y condiciones de uso

Uso de cookies

Este sitio web utiliza cookies para que usted tenga la mejor experiencia de usuario. Si continúa navegando está dando su consentimiento para la aceptación de las mencionadas cookies y la aceptación de nuestra política de cookies, pinche el enlace para mayor información

[ACEPTAR](#)