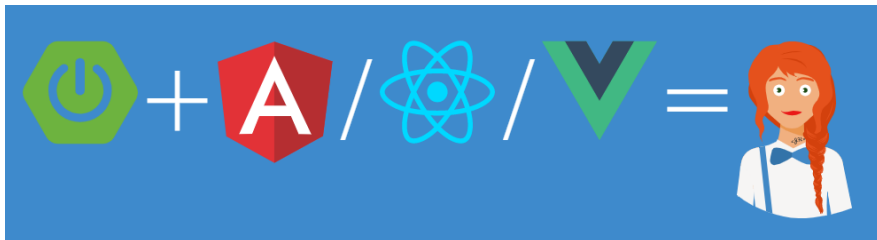


Sistemas Distribuidos e Internet

Seminario 4

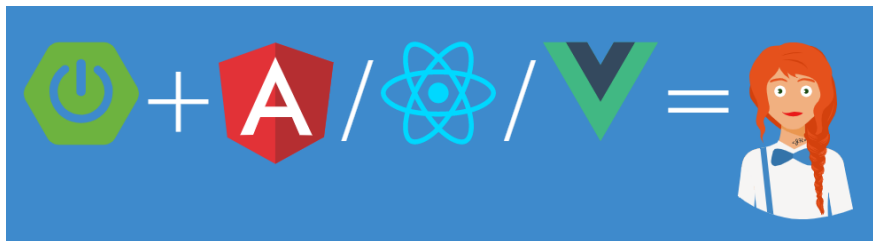
Desarrollo ágil con Spring Boot y JHipster



<https://www.jhipster.tech/artwork/>

¿Qué es JHispter?

- En una plataforma de desarrollo que se utiliza para **generar, desarrollar y desplegar aplicaciones Web Spring Boot + Angular/React/Vue y microservicios Spring** de una manera rápida y sencilla.
- Es una herramienta de *Scaffolding*



¿Qué es Node.js?

- Es una plataforma de software que permite ***ejecutar JavaScript del lado del servidor***
- Utiliza un diseño ***basado en eventos***
- Las operaciones ***Entrada/Salida*** en Node.js se llevan a cabo de ***forma asíncrona***.
- Usa un modelo de operaciones **Entrada/Salida sin bloqueo**
 - Utiliza un **único hilo de ejecución** que gestiona las entradas y salidas asíncronas de los clientes conectados
- Permite construir aplicaciones web rápidas y escalables



¿Qué es Angular?

- Angular es una plataforma y un Framework **para crear aplicaciones cliente** en HTML y TypeScript.
- Angular está escrito en TypeScript
- Proyecto de código abierto mantenido por Google y que tiene una gran popularidad
- Facilita el desarrollo de aplicaciones multiplataforma
- Permite el desarrollo de aplicaciones basadas en el patrón MVC
- La finalidad principal de Angular es facilitar el desarrollo de **aplicaciones web SPA (Single Page Application)**
 - No recargan el navegador
 - La navegación y carga de datos es dinámica y asíncrona (utiliza Ajax).



Ventajas de JHispter

- Permite generar aplicaciones web completas de forma ágil
- Código fuente muy bien estructurado y robusto
- Código fuente generado es fácil de modificar
- Incluye un sistema de autenticación y gestión de usuario
- Incluye internacionalización, métricas, logs, etc.
- JDL -> DSL para generar CRUD de nuevas entidades y sus relaciones
- Comunidad de desarrolladores muy activa y bastante grande
- Es de código abierto y todo el desarrollo está disponible en GitHub

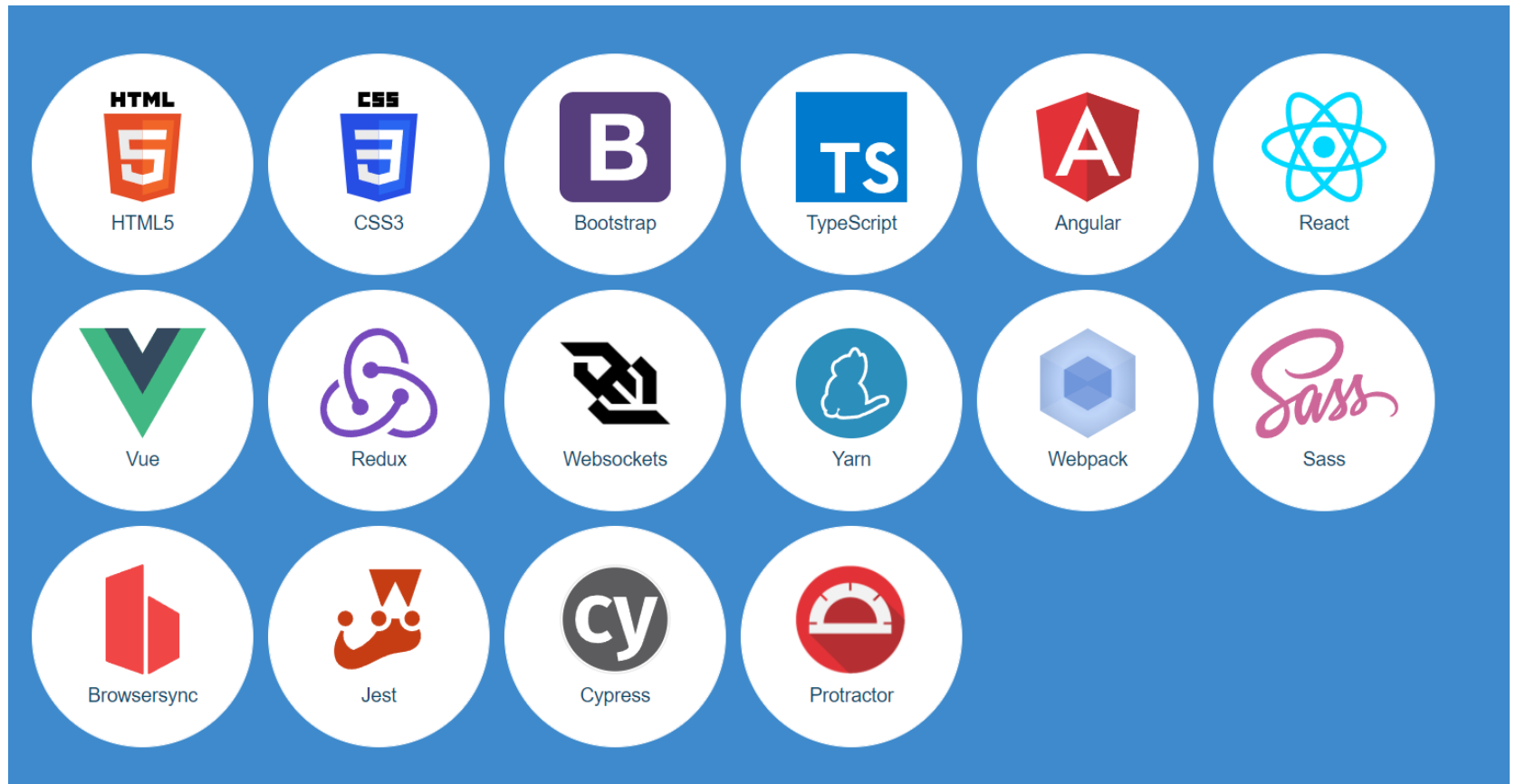
Desventajas de JHispter

- Hay que conocer las tecnología utilizadas para ser productivo
- Posibles conflictos al regenerar o modificar un proyecto existente
- Hay que tener cuidado cuando se cambia de versión de las tecnologías utilizadas

Tipos de aplicaciones con JHipster

- **Basadas en una arquitectura *'monolítica'*** utiliza una única aplicación para todo, que contiene tanto el código angular (frontend) como el código Spring Boot (backend).
- **Basadas en arquitectura de *'microservicios'*** divide el frontend y el backend, de modo que es más fácil para escalar una aplicación y sobrevivir a problemas de infraestructura.
- **Microservicios Gateway:** Es una aplicación que proporciona básicamente una interfaz API REST o WebSocket que sirve puerta de enlace "Gateway" o punto de entrada para comunicarse con grupo de microservicios y/o API de terceros.

Tecnologías soportadas (Frontend)



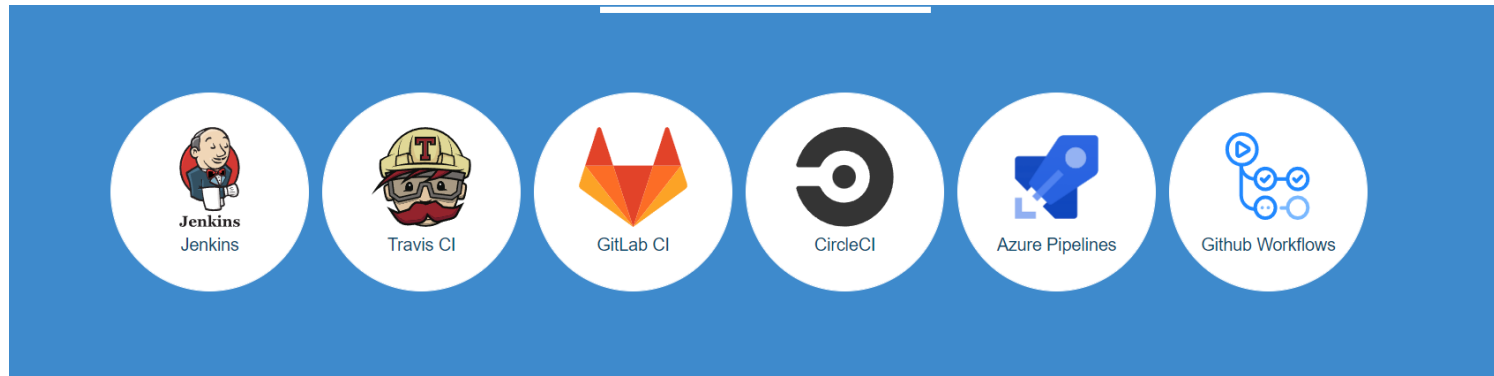
Tecnologías soportadas (Backend)





















Tecnologías soportadas (Despliegue)



Tecnologías soportadas (Integración continua)



¿Quién usa JHipster?

 Barclays Capital Finance United Kingdom	 Bosch Industry Germany	 Capgemini System Integrator France	 Carrefour Retail France	 CERN Research Switzerland	 CGI System Integrator Canada
 Conseil d'état Government France	 Consiglio Nazionale delle Ricerche Research	 Decathlon Retail France	 Direct Energie Energy France	 Docapost Information Technology France	 Engie Energy France
 Ericsson Information Technology Sweden	 Google Software Editor USA	 HBO Entertainment USA	 HCL Technologies System Integrator India	 Heroku Software Editor USA	 Hewlett Packard Enterprise Information Technology USA

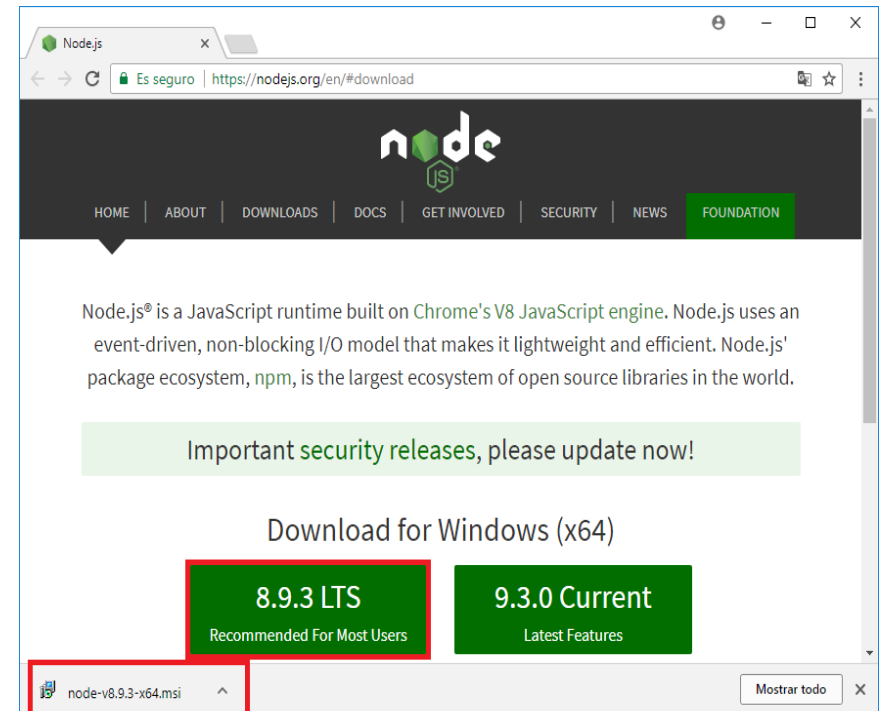
<https://www.jhipster.tech/companies-using-jhipster/>

Instalación

- Paso 1. Descargar e instalar Java y Maven
- Paso 2. Descargar e instalar Node.js
- Paso 3. Descargar e instalar Git.
- Paso 4. Instalar Jhipster

Instalación

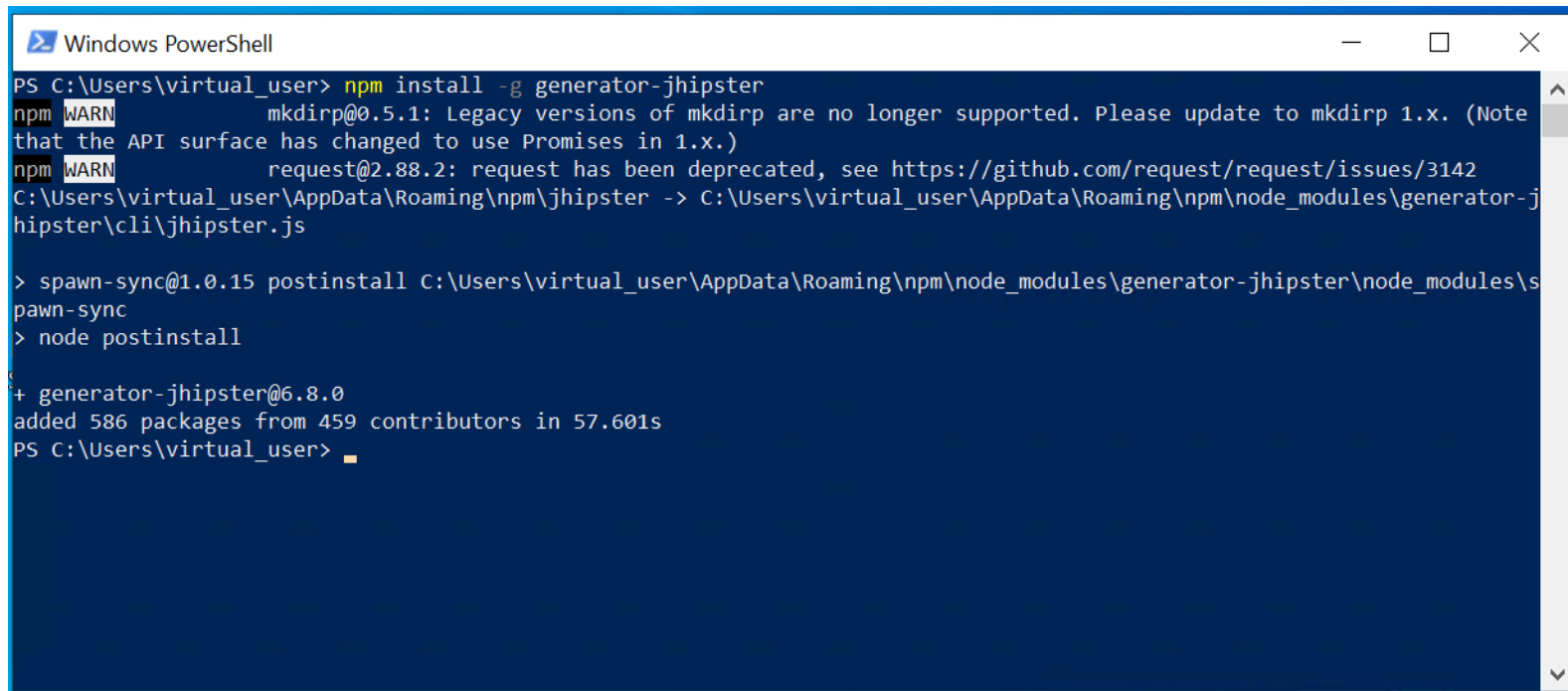
- Paso 1. Descargar e instalar Java
 - <http://www.oracle.com/tech/network/java/javase/downloads/index.html>
- Paso 2. Descargar e instalar Node.js
 - <https://nodejs.org/es/download/package-manager/>
- Paso 3. Descargar e Instalar Git
 - <https://git-scm.com/book/en/v2/Getting-Started-Installing-Git>



Instalación

■ Paso 4 - Instalar JHipster

npm install -g generator-jhipster



```
Windows PowerShell
PS C:\Users\virtual_user> npm install -g generator-jhipster
npm WARN mkdirp@0.5.1: Legacy versions of mkdirp are no longer supported. Please update to mkdirp 1.x. (Note
that the API surface has changed to use Promises in 1.x.)
npm WARN request@2.88.2: request has been deprecated, see https://github.com/request/request/issues/3142
C:\Users\virtual_user\AppData\Roaming\npm\jhipster -> C:\Users\virtual_user\AppData\Roaming\npm\node_modules\generator-j
hipster\cli\jhipster.js

> spawn-sync@1.0.15 postinstall C:\Users\virtual_user\AppData\Roaming\npm\node_modules\generator-jhipster\node_modules\s
pawn-sync
> node postinstall

+ generator-jhipster@6.8.0
added 586 packages from 459 contributors in 57.601s
PS C:\Users\virtual_user>
```

CREANDO UNA APLICACIÓN CON JHIPSTER

Creación de aplicación

- Crear directorio de la aplicación y movernos a dicho directorio

```
PS C:\Dev\workspaces\jhispter-projects> mkdir mysocialnetworkapp

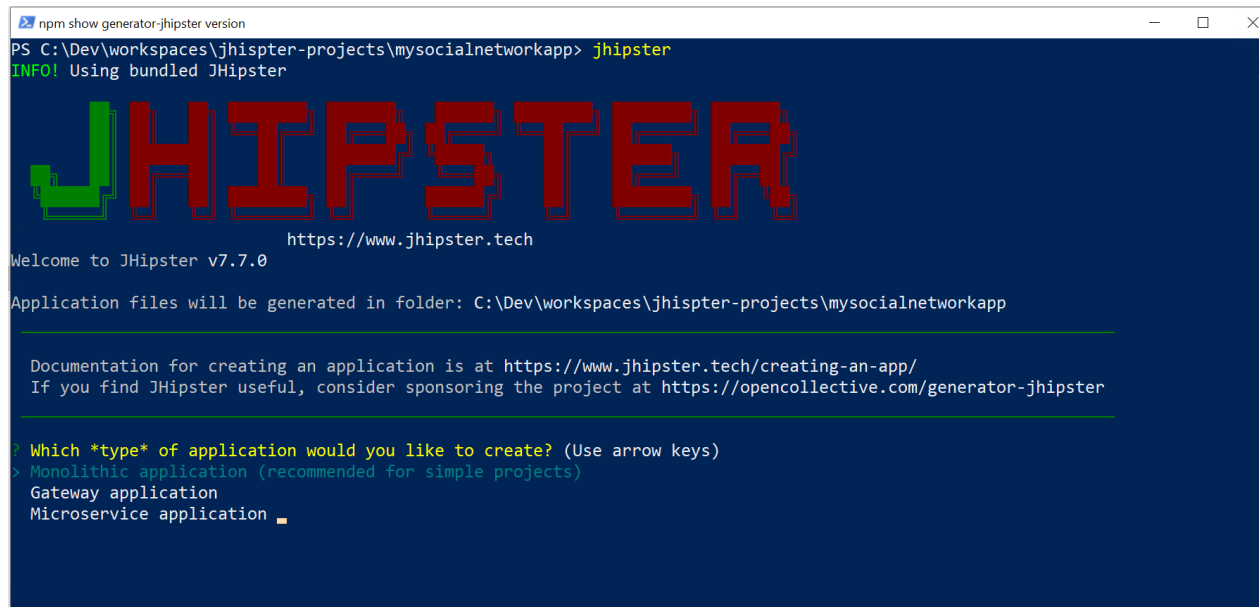
Directorio: C:\Dev\workspaces\jhispter-projects

Mode                LastWriteTime         Length Name
----                -
d-----         29/03/2022         9:43             mysocialnetworkapp

PS C:\Dev\workspaces\jhispter-projects> █
```

Creación de aplicación

- Generamos la aplicación con Jhipster, con el comando: *jhipster*



```
npm show generator-jhipster version
PS C:\Dev\workspaces\jhipster-projects\mysocialnetworkapp> jhipster
INFO! Using bundled JHipster

JHIPSTER

https://www.jhipster.tech

Welcome to JHipster v7.7.0

Application files will be generated in folder: C:\Dev\workspaces\jhipster-projects\mysocialnetworkapp

Documentation for creating an application is at https://www.jhipster.tech/creating-an-app/
If you find JHipster useful, consider sponsoring the project at https://opencollective.com/generator-jhipster

Which *type* of application would you like to create? (Use arrow keys)
> Monolithic application (recommended for simple projects)
Gateway application
Microservice application
```

Creación de aplicación

- JHipster realizará una serie de preguntas para generar la aplicación, como por ejemplo:
 - El tipo de aplicación a generar
 - Monolíticas, Basadas de microservicios o aplicaciones Gateway
 - Nombre de aplicación
 - Nombre del paquete base
 - Tipo de autenticación a utilizar
 - Herramienta para construcción del proyecto (Maven o grandle)
 - Bases de datos a utilizar (en producción y desarrollo)
 - Idiomas para la internacionalización
 - Framework de pruebas a utilizar (JUNIT y Karma, Gartling...)
 - Etc.

Creación de aplicación


■ Preguntas ...

```
? Which *type* of application would you like to create? Monolithic application (recommended for simple projects)
? What is the base name of your application? mysocialnetworkapp
? Do you want to make it reactive with Spring WebFlux? No
? What is your default Java package name? com.uniovi.sdi
? Which *type* of authentication would you like to use? HTTP Session Authentication (stateful, default Spring Security mechanism)
? Which *type* of database would you like to use? SQL (H2, PostgreSQL, MySQL, MariaDB, Oracle, MSSQL)
? Which *production* database would you like to use? MySQL
? Which *development* database would you like to use? H2 with disk-based persistence
? Which cache do you want to use? (Spring cache abstraction) Ehcache (local cache, for a single node)
? Do you want to use Hibernate 2nd level cache? Yes
? Would you like to use Maven or Gradle for building the backend? Maven
? Do you want to use the JHipster Registry to configure, monitor and scale your application? Yes
? Which other technologies would you like to use?
? Which *Framework* would you like to use for the client? Angular
? Do you want to generate the admin UI? Yes
? Would you like to use a Bootswatch theme (https://bootswatch.com/)? Default JHipster
? Would you like to enable internationalization support? Yes
? Please choose the native language of the application Spanish
? Please choose additional languages to install English, Portuguese
? Besides JUnit and Jest, which testing frameworks would you like to use? Cucumber
? Would you like to install other generators from the JHipster Marketplace? No
```

Creación de aplicación

- Soporte a diferentes tipos de bases de datos


```
? May JHipster anonymously report usage statistics to improve the tool over time? Yes
? Which *type* of application would you like to create? Monolithic application (recommended for simple projects)
? [Alpha] Do you want to make it reactive with Spring WebFlux? No
? What is the base name of your application? mysocialnetworkapp
? What is your default Java package name? sdi.com.uniovi
? Do you want to use the JHipster Registry to configure, monitor and scale your application? No
? Which *type* of authentication would you like to use? JWT authentication (stateless, with a token)
? Which *type* of database would you like to use? (Use arrow keys)
> SQL (H2, MySQL, MariaDB, PostgreSQL, Oracle, MSSQL)
MongoDB
Cassandra
Couchbase
[BETA] Neo4j
No database
```




Creación de aplicación

■ Soporte a diferentes tecnologías


```
? Which other technologies would you like to use? (Press <space> to select, <a> to toggle all, <i> to invert selection)
( ) Search engine using Elasticsearch
( ) WebSockets using Spring Websocket
( ) Asynchronous messages using Apache Kafka
( ) API first development using OpenAPI-generator
```



```
? Which *Framework* would you like to use for the client? (Use arrow keys)
> Angular
  React
  Vue
  No client
```



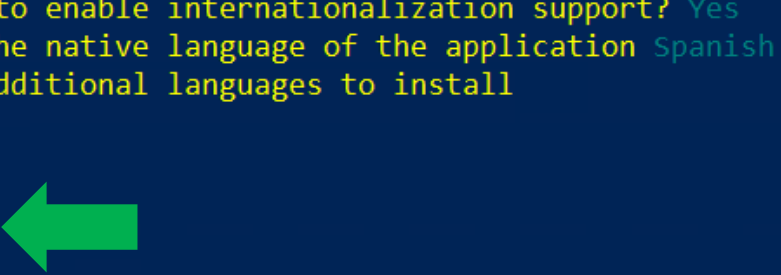
```
? Would you like to use Maven or Gradle for building the backend? (Use arrow keys)
> Maven
  Gradle
```



Creación de aplicación

- Internacionalización a múltiples idiomas

```
? Would you like to enable internationalization support? Yes
? Please choose the native language of the application Spanish
? Please choose additional languages to install
(*) English
( ) Estonian
( ) Farsi
>(*) French
( ) Galician
( ) German
( ) Greek
(Move up and down to reveal more choices)
```



Creación de aplicación

- Soporte a múltiples framework de testing

```
? Besides JUnit and Jest, which testing frameworks would you like to use?  
t>( ) Cypress  
r ( ) [DEPRECATED] Protractor  
i ( ) Gatling  
l ( ) Cucumber
```


Creación de aplicación

■ Generando código ...

```
create src\main\resources\config\application.yml
create src\main\docker\grafana\provisioning\datasources\datasource.yml
create src\main\resources\config\application-dev.yml
create src\main\resources\config\application-tls.yml
create src\main\docker\mysql.yml
create src\main\resources\config\application-prod.yml
```

Configuración

```
create src\main\java\com\uniovi\sdi\web\rest\errors\EmailAlreadyUsedException.java
create src\main\java\com\uniovi\sdi\security\jwt\TokenProvider.java
create src\main\java\com\uniovi\sdi\MySocialNetworkAppApp.java
create src\main\java\com\uniovi\sdi\config\JacksonConfiguration.java
create src\main\java\com\uniovi\sdi\repository\package-info.java
create src\main\java\com\uniovi\sdi\web\rest\errors\InvalidPasswordException.java
create src\main\java\com\uniovi\sdi\security\jwt\JWTFilter.java
create src\main\java\com\uniovi\sdi\ApplicationWebXml.java
create src\main\java\com\uniovi\sdi\config\LoggingAspectConfiguration.java
create src\main\java\com\uniovi\sdi\service\EmailAlreadyUsedException.java
create src\main\java\com\uniovi\sdi\web\rest\errors>LoginAlreadyUsedException.java
```

Backend

```
create src\main\webapp\app\home\home.route.ts
create src\main\webapp\app\home\home.component.ts
create src\main\webapp\app\home\home.component.html
create src\main\webapp\app\home\home.component.scss
```

Frontend

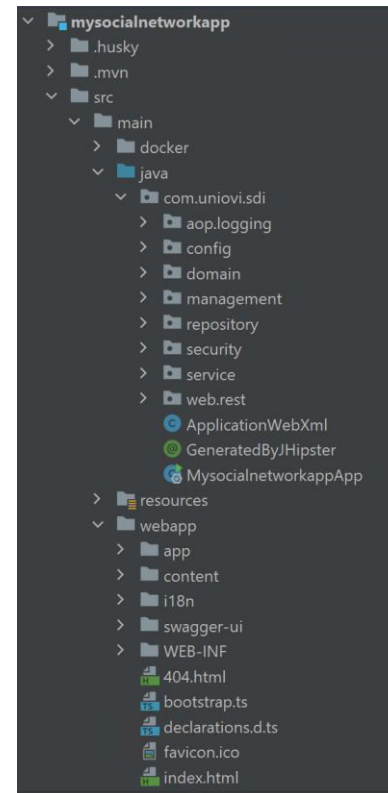
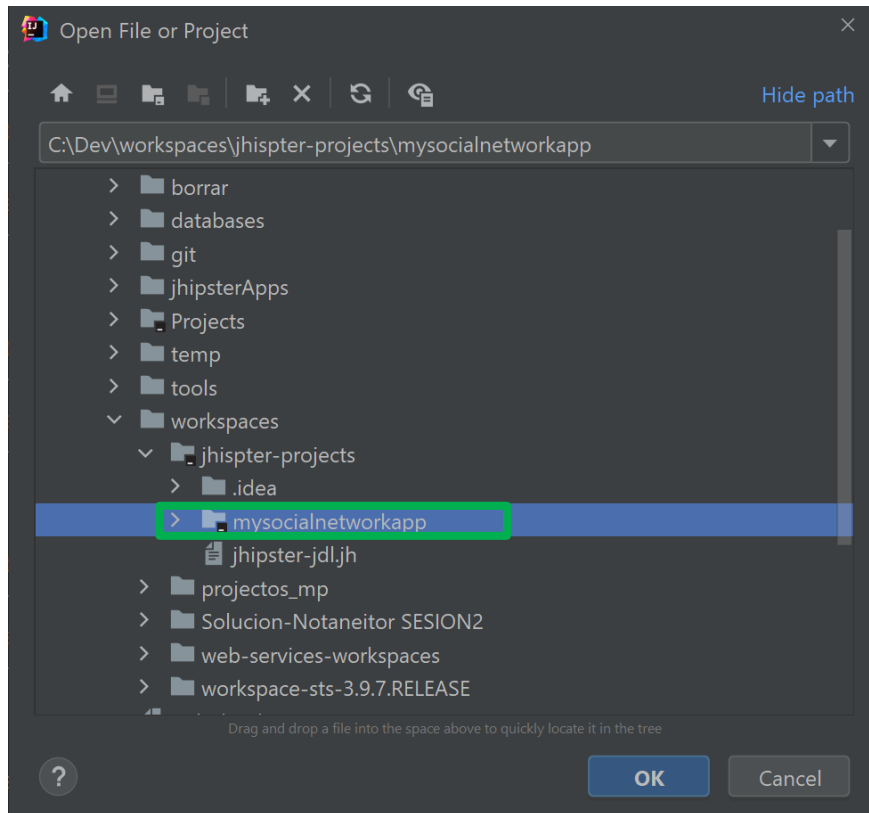
Creación de aplicación

- Una vez completadas las preguntas JHipster generará la aplicación completa y mostrará un mensaje de finalización.

```
Server application generated successfully.  
Run your Spring Boot application:  
./mvnw (mvnw if using Windows Command Prompt)  
Client application generated successfully.  
Start your Webpack development server with:  
npm start
```

Creación de aplicación

- Podemos importarla a IntelliJ IDEA u otro IDE



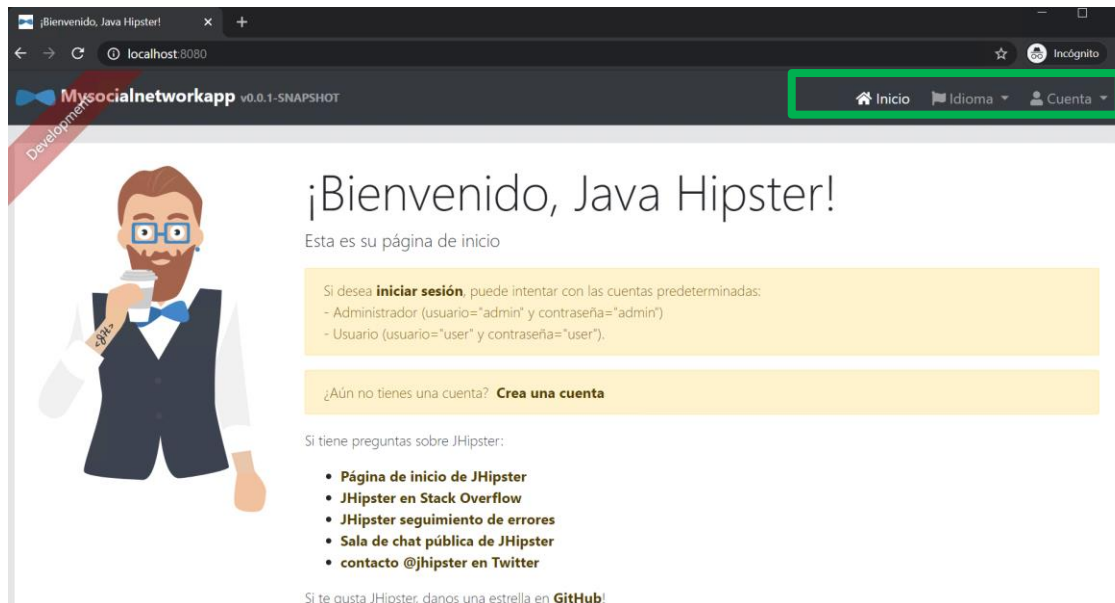
Compilando y desplegando aplicación

- Para compilar y desplegar la aplicación ejecutamos por consola:
 - *./mvnw (en macOS/Linux)*
 - *mvnw (en Windows)*
- Una vez compilado y desplegado

```
03-29 11:18:53.037 INFO 4508 --- [ restartedMain] com.uniovi.sdi.MysocialnetworkappApp
-----
Application 'mysocialnetworkapp' is running! Access URLs:
Local:      http://localhost:8080/
External:   http://172.24.64.170:8080/
Profile(s): [dev, api-docs]
-----
```

Aplicación Web

- Desde el navegador vamos a URL
 - <http://localhost:8080>
 - La pagina principal nos muestra varias opciones



Aplicación Web

■ Menú para administradores

Usuarios

localhost8080/admin/user-management

MySocialNetworkapp v0.0.1-SNAPSHOT

Inicio Entidades Administración Idioma Cuenta

Gestión de usuarios

- Métricas
- Salud
- Configuración
- Auditorías
- Logs
- API
- Base de datos

ID	Login	Email		Idioma	Perfiles	Fecha de creación	Modificado por	
1	system	system@localhost	Activado	es	ROLE_USER ROLE_ADMIN		system	Vista Editar Eliminar
3	admin	admin@localhost	Activado	es	ROLE_USER ROLE_ADMIN		system	Vista Editar Eliminar
4	user	user@localhost	Activado	es	ROLE_USER		system	Vista Editar Eliminar

Mostrando 1 - 3 de 3 elementos.

« « 1 » »

Aplicación Web

■ Gestión de usuarios

Usuarios

+ Crear un nuevo usuario

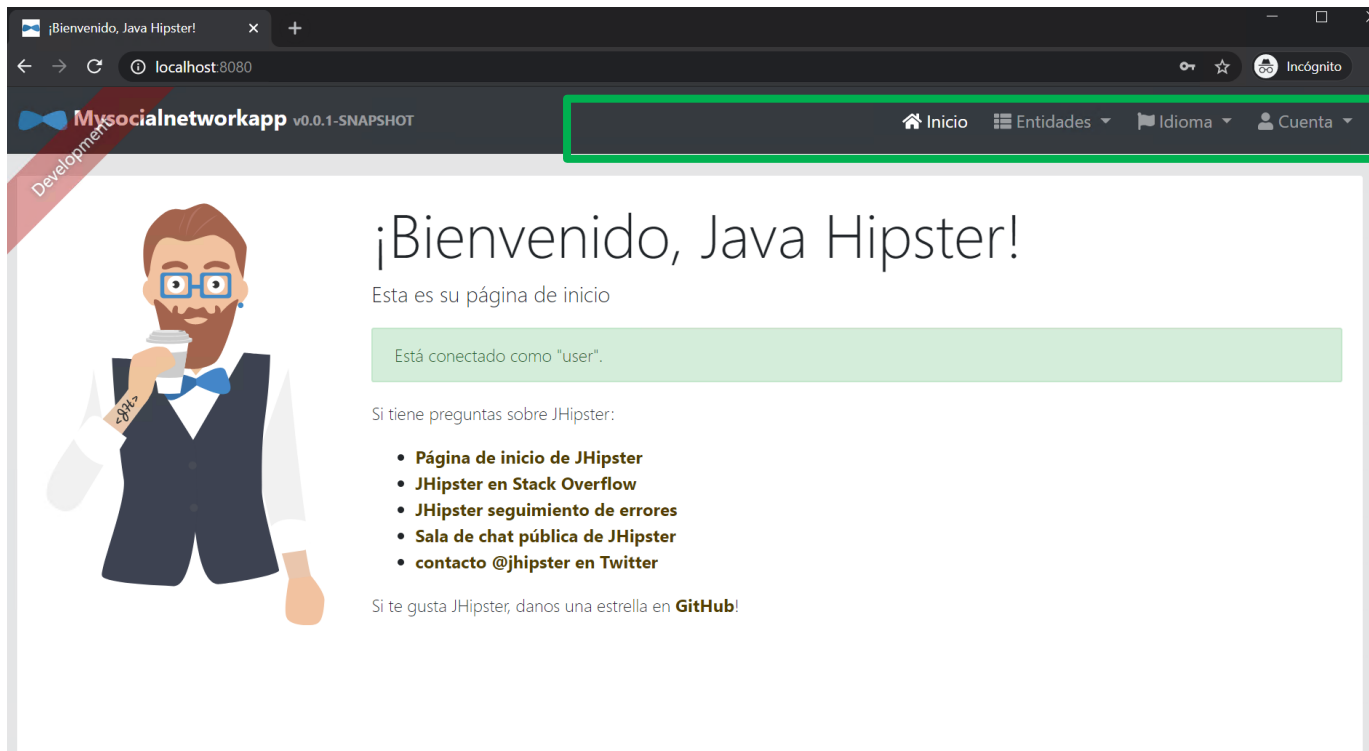
ID	Login	Email		Idioma	Perfiles	Fecha de creación	Modificado por	Fecha de modificación
1	system	system@localhost	Activado	es	ROLE_USER ROLE_ADMIN		system	Vista Editar Eliminar
3	admin	admin@localhost	Activado	es	ROLE_USER ROLE_ADMIN		system	Vista Editar Eliminar
4	user	user@localhost	Activado	es	ROLE_USER		system	Vista Editar Eliminar

Mostrando 1 - 3 de 3 elementos.

««« « 1 » »»»

Aplicación Web

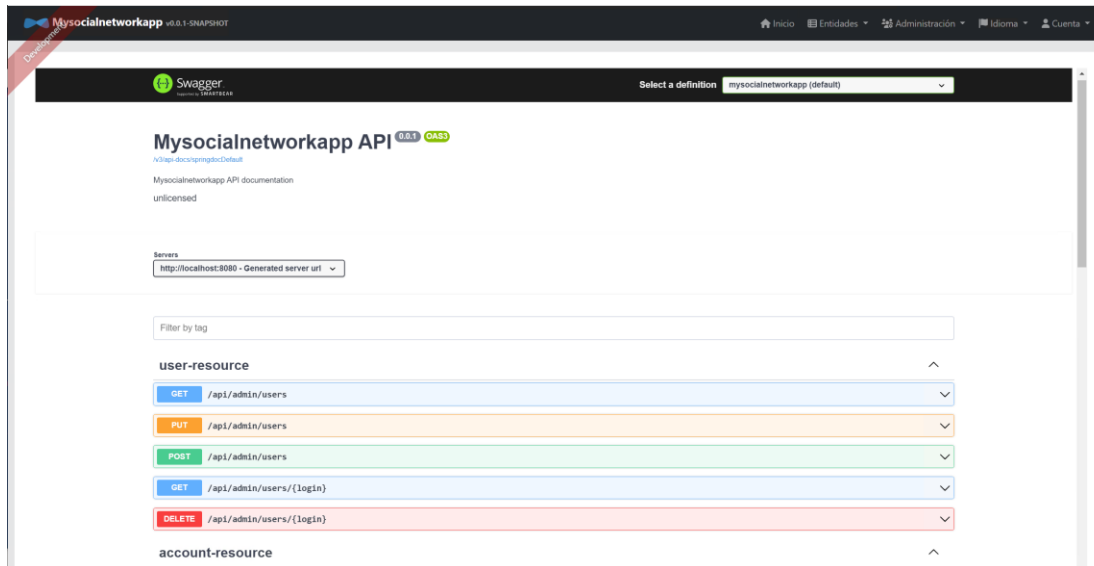
- Menú para usuarios registrados (no Administradores)



Aplicación Web

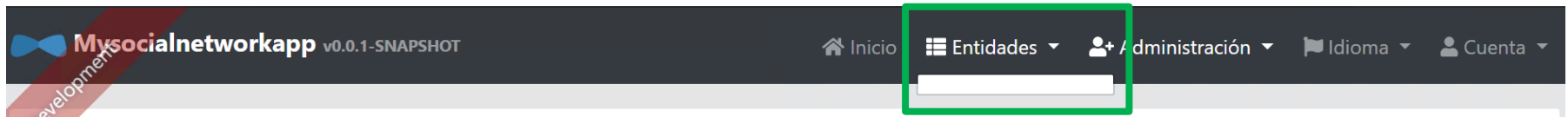
- API REST

- Muestra los endpoints de la API del sistema



JHipster Domain Language (JDL)

- Por defecto las aplicaciones generadas con Jhipster solo vienen con las entidades básicas definidas.
- Podemos utilizar JDL para definir las entidades de nuestra aplicación



JHipster Domain Language (JDL)

- JDL es un ***lenguaje de dominio específico*** de JHipster para describir todas ***las entidades y sus relaciones*** de una aplicación
- Con una sintaxis simple y fácil de usar
- Con el ***IDE JDL-Studio online*** se puede crear modelos JDL y su visualización UML.
- También se puede crear y exportar o compartir la URL de su modelo JDL.

JDL- Studio

- Definiendo entidades con JDL- Studio y sus relaciones
 - Entidad Publicaciones
 - Entidad Etiquetas
 - Relación entre publicaciones y etiquetas
 - Relación entre publicaciones y usuarios

JDL- Studio

- Definiendo entidades con JDL- Studio y sus relaciones

```
entity Publication {  
  title String required,  
  description String,  
  date Instant  
}  
  
entity Tag {  
  name String required minlength(4)  
}  
  
relationship ManyToOne {  
  Publication{user(login)} to User  
}  
  
relationship ManyToMany {  
  Publication{tag(name)} to Tag{publication}  
}  
  
// Set pagination options  
paginate Publication with pagination  
paginate Tag with infinite-scroll  
  
service all with serviceImpl
```

The diagram on the right shows the entities and their relationships:

- Publication** entity with attributes: title: String*, description: String*, date: Instant.
- User** entity.
- Tag** entity with attribute: name: String*.
- Relationships: Publication to User (ManyToOne, *..1), Publication to Tag (ManyToMany, *..*).

Generando Entidades

- Descargamos el fichero y ejecutamos el siguiente comando
 - ***Jhipster import-jdl .. \jhipster-jdl.jh***

```
PS C:\Dev\workspaces\jhipster-projects\mysocialnetworkapp> jhipster import-jdl ..\jhipster-jdl.jh
INFO! Switching to JHipster installed locally in current project's node repository (node_modules)

JHIPSTER
https://www.jhipster.tech

Welcome to JHipster v7.7.0

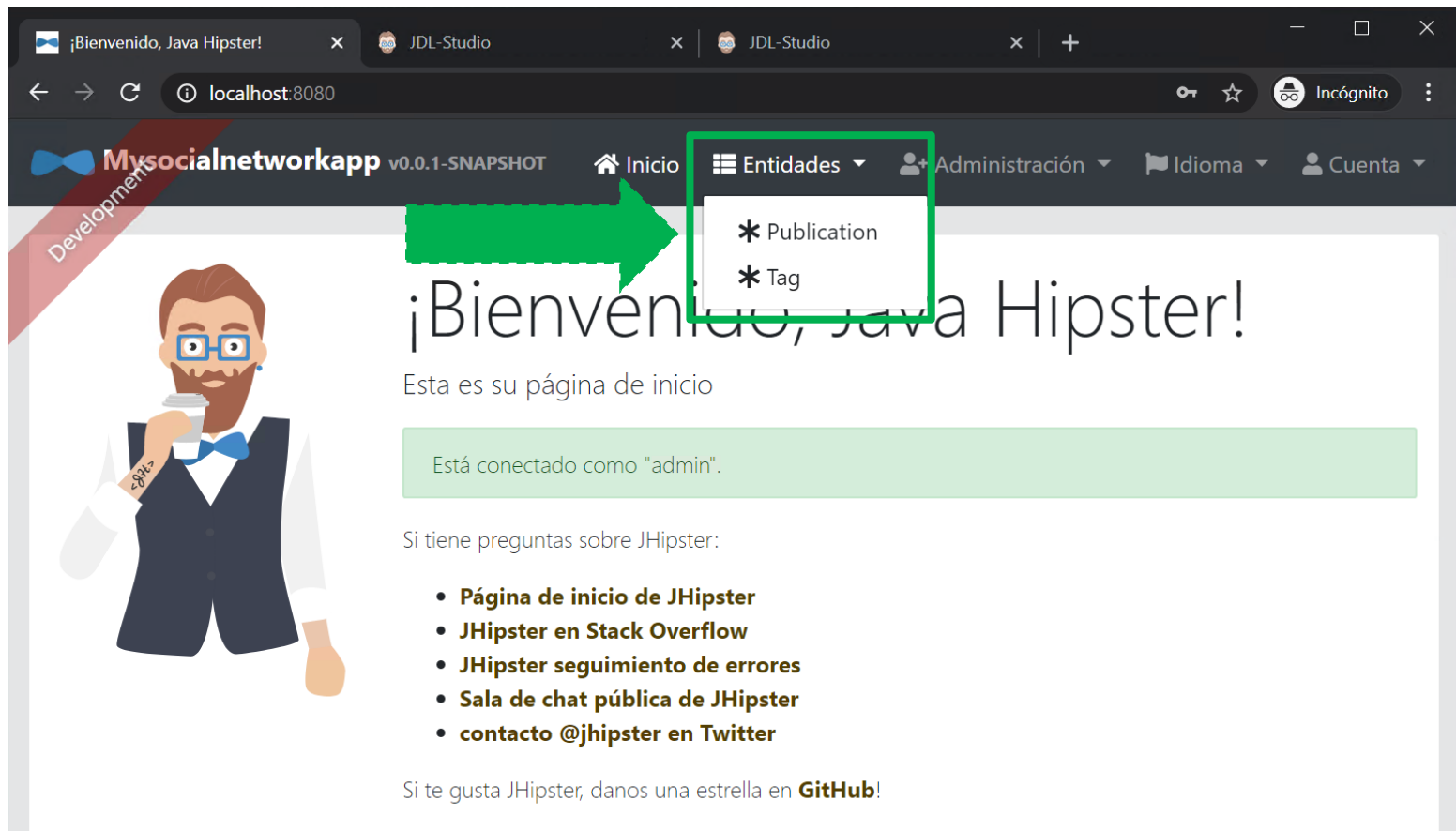
INFO! Executing import-jdl ..\jhipster-jdl.jh
INFO! The JDL is being parsed.
warn: An Entity name 'User' was used: 'User' is an entity created by default by JHipster. All relationships toward it will be kept but any attributes and relationships from it will be disregarded.
INFO! Found entities: Publication, Tag.
INFO! The JDL has been successfully parsed
INFO! Generating 0 applications.
INFO! Generating 2 entities.
INFO! Generating entities for application undefined in a new parallel process

Found the .jhipster\Publication.json configuration file, entity can be automatically generated!

Found the .jhipster\Tag.json configuration file, entity can be automatically generated!

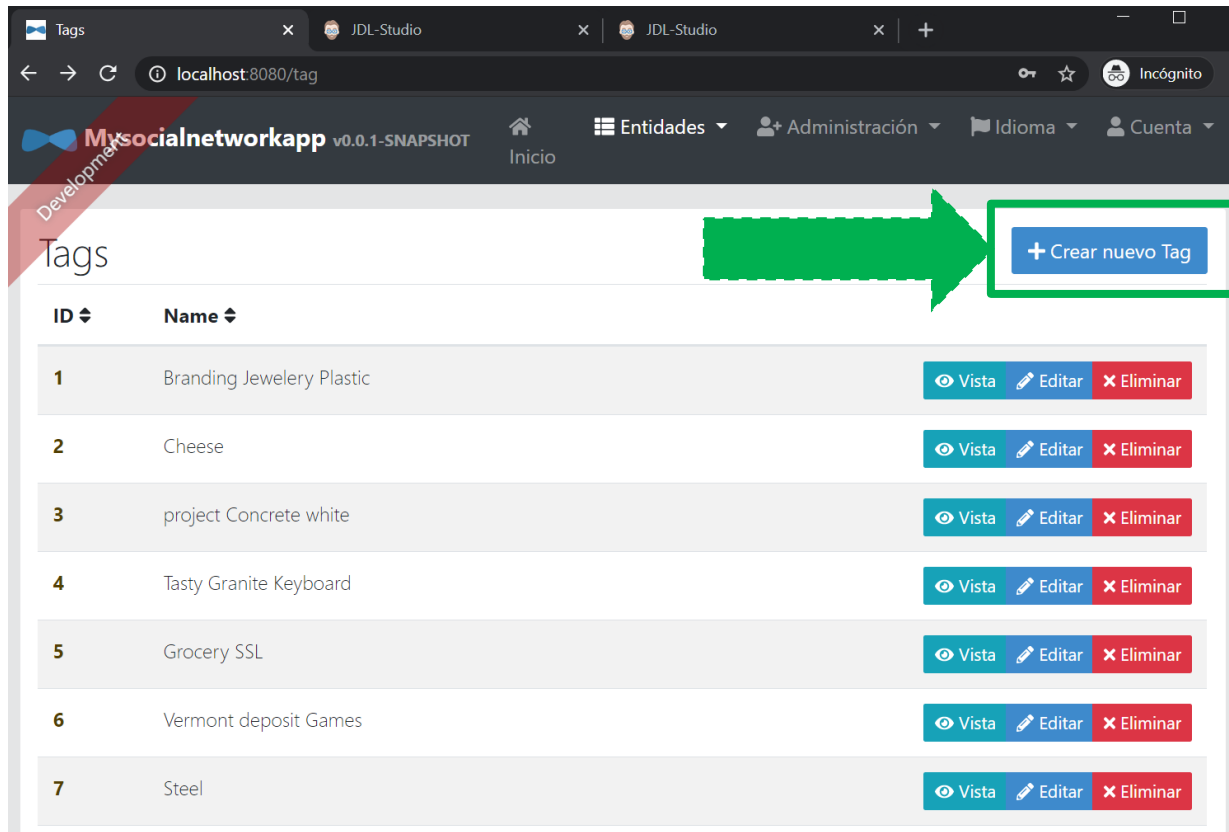
info Creating changelog for entities Publication,Tag
force .yo-rc.json
force .jhipster\Publication.json
force .jhipster\Tag.json
create src\main\resources\config\liquibase\changelog\20220329094111_added_entity_Publication.xml
create src\main\resources\config\liquibase\changelog\20220329094111_added_entity_constraints_Publication.xml
create src\main\resources\config\liquibase\fake-data\publication.csv
create src\main\resources\config\liquibase\changelog\20220329094112_added_entity_Tag.xml
create src\main\resources\config\liquibase\fake-data\tag.csv
```

Generando Entidades



Generando Entidades

- Operaciones CRUD con entidades



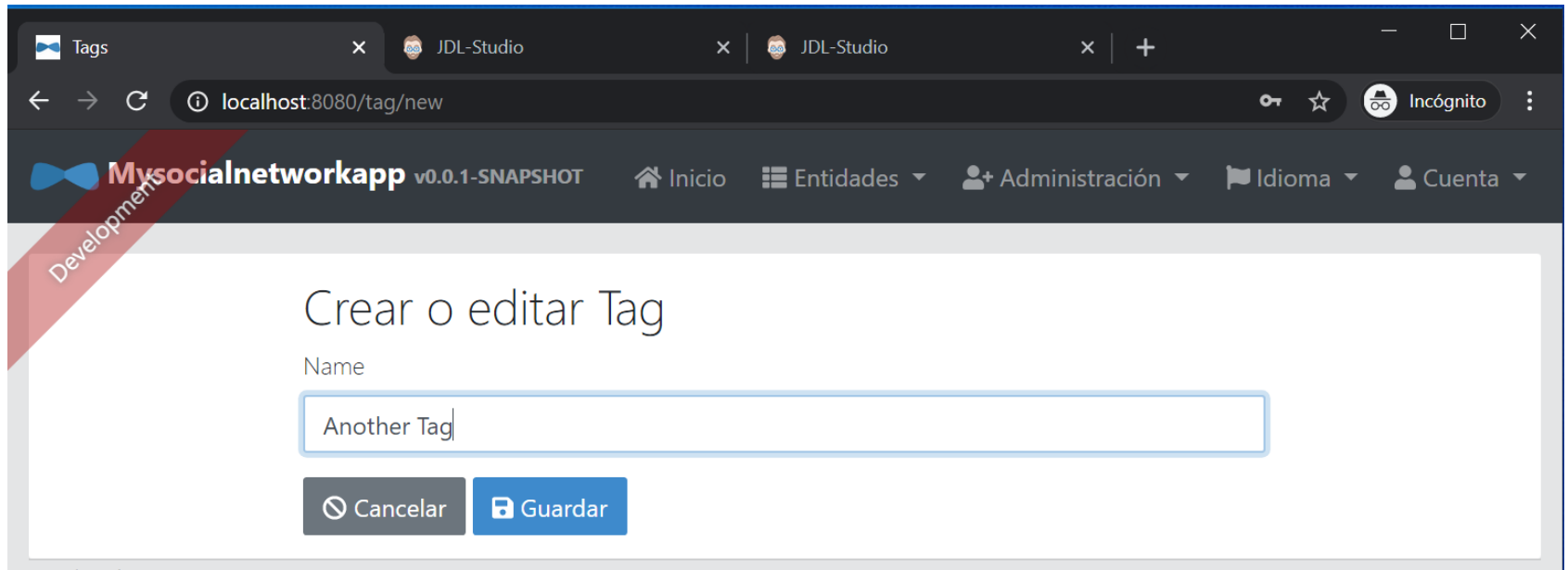
The screenshot shows a web application interface for managing tags. The browser address bar indicates the URL is `localhost:8080/tag`. The application header includes the logo 'My socialnetworkapp v0.0.1-SNAPSHOT', a 'Inicio' link, and navigation menus for 'Entidades', 'Administración', 'Idioma', and 'Cuenta'. A red diagonal banner on the left reads 'Development'. The main content area is titled 'Tags' and features a table with the following data:

ID	Name	Actions
1	Branding Jewellery Plastic	Vista Editar Eliminar
2	Cheese	Vista Editar Eliminar
3	project Concrete white	Vista Editar Eliminar
4	Tasty Granite Keyboard	Vista Editar Eliminar
5	Grocery SSL	Vista Editar Eliminar
6	Vermont deposit Games	Vista Editar Eliminar
7	Steel	Vista Editar Eliminar

A green arrow points to a blue button labeled '+ Crear nuevo Tag' located in the top right corner of the tags list area.

Generando Entidades

- Creando etiquetas



The screenshot shows a web browser window with the address bar at `localhost:8080/tag/new`. The page title is "Tags". The application header includes the logo "Mysocialnetworkapp v0.0.1-SNAPSHOT" and navigation links: "Inicio", "Entidades", "Administración", "Idioma", and "Cuenta". A red diagonal banner on the left side of the page reads "Development". The main content area is titled "Crear o editar Tag" and contains a form with a "Name" label and a text input field containing "Another Tag". Below the input field are two buttons: "Cancelar" (with a close icon) and "Guardar" (with a save icon).

Generando Entidades

- Creando publicaciones y sus relaciones

Crear o editar Publication

Title

Another Publication

Description

My Description

Date

18/03/2020 00:00


User

admin

Tag

Branding Jewellery Plastic
Cheese
project Concrete white
Tasty Granite Keyboard

Cancel Guardar



Generando Entidades

- Listando publicaciones y paginando

Publications					+ Crear nuevo Publication	
ID	Title	Description	Date	User		
11	Another Publication	My Description	18 mar. 2020 0:00:00	admin	Vista	Eliminar
10	Assurance	Practical Product e-commerce	18 mar. 2020 3:58:48		Vista	Eliminar
9	Accounts Computer	empowering	18 mar. 2020 15:51:28		Vista	Eliminar



Generando Entidades

■ Detalle de una publicación

Dev

Publication 11

Title
Another Publication


Description
My Description

Date
1584486000000

User
admin

Tag
Cheese, project Concrete white

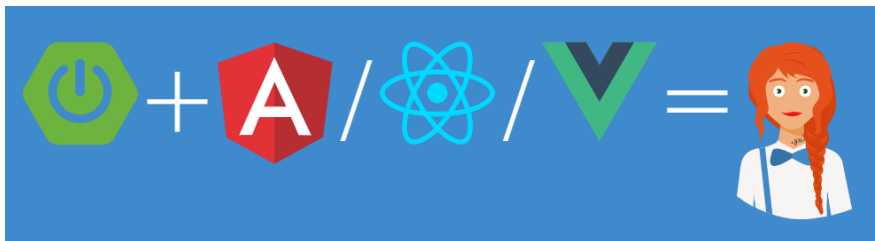
← Volver

 Editar

Sistemas Distribuidos e Internet

Seminario 4

Desarrollo ágil con Spring Boot y JHipster



<https://www.jhipster.tech/artwork/>