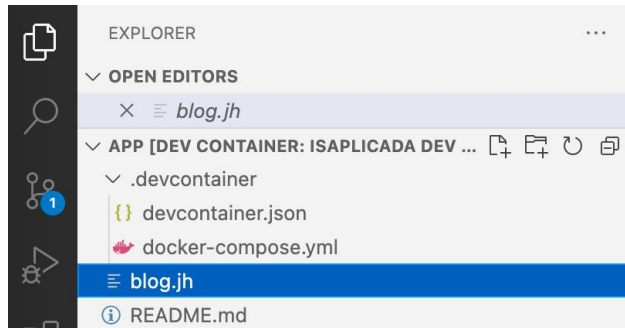


INGENIERÍA DE SOFTWARE APLICADA 2021

jeremias.castillo@um.edu.ar

EJEMPLO JHIPSTER APP

Blog App - GENERACIÓN DE APP



PROBLEMS OUTPUT TERMINAL PORTS DEBUG CONSOLE

To run a command as administrator (user "root"), use "sudo <command>".
See "man sudo_root" for details.

```
jhipster@39e815c1f20f:~/app$ jhipster jdl blog.jh
```

```
create src/main/docker/jib/entrypoint.sh
conflict README.md
? Overwrite README.md? (ynarxdeH) y
>> overwrite
```

- Definición de la App

blog.jh

- Generamos la aplicación mediante el generador de JHipster

```
~/app$ jhipster jdl blog.jh
```

...

Reemplazar README.md

Blog App - CREACIÓN DE ARCHIVOS

PROBLEMS OUTPUT TERMINAL PORTS DEBUG CONSOLE

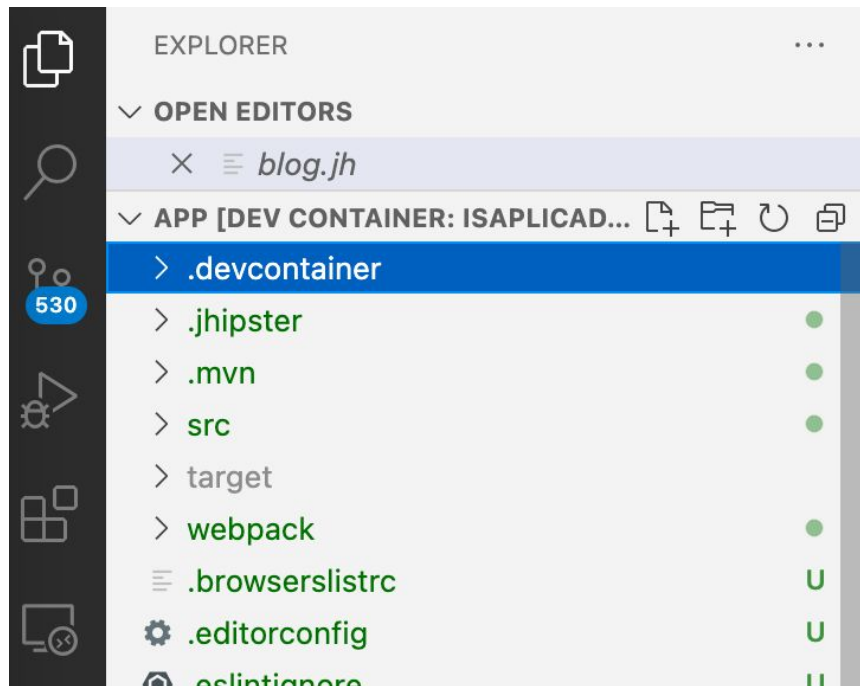
node + v

```
create src/main/webapp/app/entities/blog/list/blog.component.ts
create src/main/webapp/app/entities/blog/detail/blog-detail.component.ts
create src/main/webapp/app/entities/blog/service/blog.service.ts
create src/main/webapp/app/entities/blog/update/blog-update.component.html
create src/main/webapp/app/entities/blog/delete/blog-delete-dialog.component.html
create src/main/webapp/app/entities/blog/update/blog-update.component.ts
create src/main/webapp/app/entities/blog/delete/blog-delete-dialog.component.ts
create src/main/webapp/app/entities/blog/detail/blog-detail.component.spec.ts
create src/main/webapp/app/entities/blog/list/blog.component.spec.ts
create src/main/webapp/app/entities/blog/route/blog-routing-resolve.service.spec.ts
create src/main/webapp/app/entities/blog/service/blog.service.spec.ts
create src/main/webapp/app/entities/blog/update/blog-update.component.spec.ts
create src/main/webapp/app/entities/blog/delete/blog-delete-dialog.component.spec.ts
create src/main/webapp/118/en/blog.json
create src/main/java/ar/edu/um/isa/blog/domain/Post.java
create src/main/java/ar/edu/um/isa/blog/web/rest/PostResource.java
create src/main/java/ar/edu/um/isa/blog/repository/PostRepository.java
create src/test/java/ar/edu/um/isa/blog/web/rest/PostResourceIT.java
create src/test/java/ar/edu/um/isa/blog/domain/PostTest.java
create src/test/javascript/cypress/integration/entity/blog.spec.ts
create src/main/webapp/app/entities/post/post.model.ts
create src/main/webapp/app/entities/post/list/post.component.html
create src/main/webapp/app/entities/post/post.module.ts
create src/main/webapp/app/entities/post/detail/post-detail.component.html
create src/main/webapp/app/entities/post/delete/post-delete-dialog.component.html
create src/main/webapp/app/entities/post/route/post-routing.module.ts
create src/main/webapp/app/entities/post/service/post.service.ts
create src/main/webapp/app/entities/post/list/post.component.ts
create src/main/webapp/app/entities/post/update/post-update.component.html
create src/main/webapp/app/entities/post/update/post-update.component.ts
create src/main/webapp/app/entities/post/delete/post-delete-dialog.component.ts
create src/main/webapp/app/entities/post/detail/post-detail.component.ts
create src/main/webapp/app/entities/post/route/post-routing-resolve.service.ts
create src/main/webapp/app/entities/post/detail/post-detail.component.spec.ts
create src/main/webapp/app/entities/post/list/post.component.spec.ts
create src/main/webapp/app/entities/post/route/post-routing-resolve.service.spec.ts
create src/main/webapp/app/entities/post/service/post.service.spec.ts
create src/main/webapp/app/entities/post/update/post-update.component.spec.ts
create src/main/webapp/app/entities/post/delete/post-delete-dialog.component.spec.ts
create src/main/webapp/app/entities/post/delete/post-delete-dialog.component.spec.ts
```

- Crea archivos de configuración
- Crea archivos de proyecto backend (java)
- Crea archivos de proyecto frontend (angular)
- Instala dependencias.

Entity Blog generated successfully.
Entity Post generated successfully.
Entity Tag generated successfully.
INFO! Generator app succeed
Congratulations, JHipster execution is complete!

Blog App - APP GENERADA



- Podemos comprobar que con un modelos de solo 3 entidades se ha generado una infraestructura que implica:

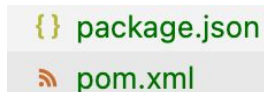
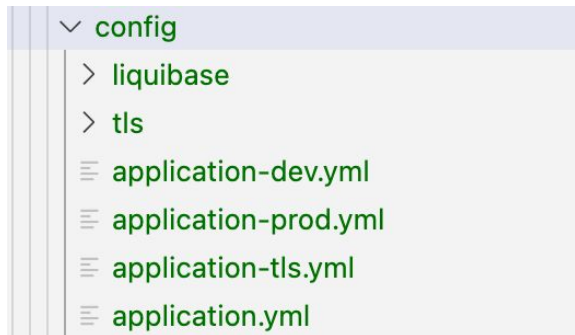
+ de 500 Archivos

- También podemos ver cuales son todas las opciones predeterminadas que utilizó el generador ejecutando

```
~/app$ jhipster export-jdl blog-generated.jh
```

... o en el archivo **/.yo-rc.json**

Blog App - CONFIGURACIONES PRINCIPALES



- Definición de **Entornos**:

/src/main/resources/config

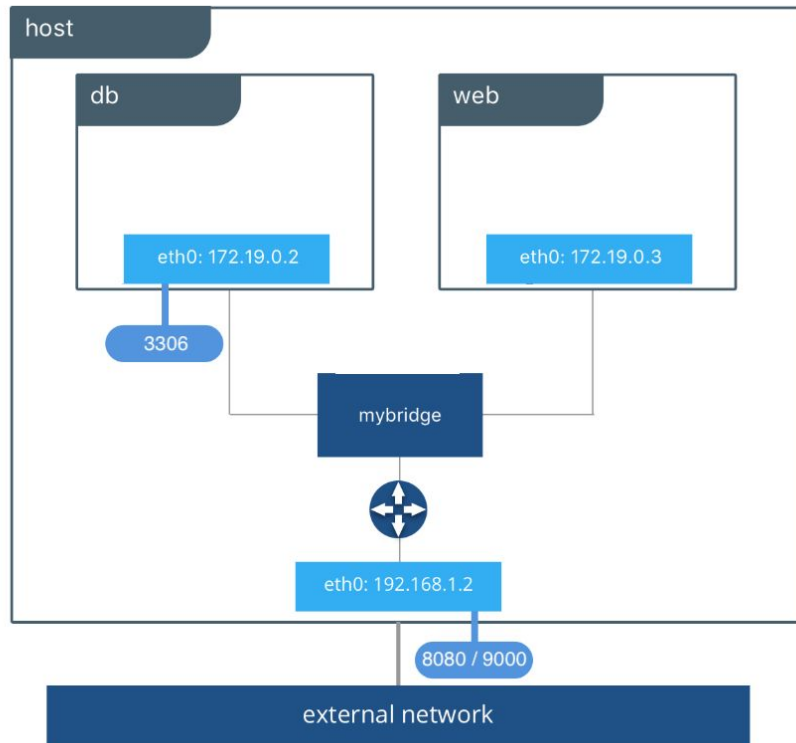
- *application.yml*
- *application-dev.yml*
-
-

- Definición de Dependencias / Build / Run / Test

/

- *pom.xml* ← (MAVEN)
-
- *package.json* ← (NPM)

Blog App - NETWORK && DB CONFIG



- Tenemos una red compartida por ambos contenedores:

```
networks:  
  isa-blog-backend-dev:  
    driver: 'bridge'
```

- Docker me permite referenciar al contenedor mediante el nombre del servicio:
 - isa-blog-db
- Apuntamos en pom.xml y application-dev.yml al nombre del servicio donde está la BD.

```
<url>jdbc:mariadb://isa-blog-db:3306/blog</url>
```

Blog App - RUN & DEBUG

```
[INFO] --- spring-boot-maven-plugin:2.4.7:run (default-cli) @ blog ---
[INFO] Attaching agents: []

    ??? ???    ??? ?????????? ?????????? ?????????? ?????????? ??????????
    ??? ???    ??? ?????????? ?????????? ?????????? ?????????? ??????????
    ??? ?????????? ??? ?????????? ?????????? ??? ?????????? ??????????
    ??? ??? ?????????? ??? ?????????? ?????????? ??? ?????????? ??????????
    ?????????? ??? ??? ?????????? ??? ?????????? ??? ?????????? ??? ???
    ?????????? ??? ??? ?????????? ??? ?????????? ??? ?????????? ??? ???

:: JHipster ? :: Running Spring Boot 2.4.7 ::
:: https://www.jhipster.tech ::
```

```
✓ Compiled successfully.
[Browsersync] Proxying: http://localhost:4200
[Browsersync] Access URLs:
-----
    Local: http://localhost:9000
    External: http://172.30.0.3:9000
-----
    UI: http://localhost:3001
    UI External: http://localhost:3001
-----
```

- En una terminal corremos la App utilizando el wrapper de Maven

`~/app$./mvnw`

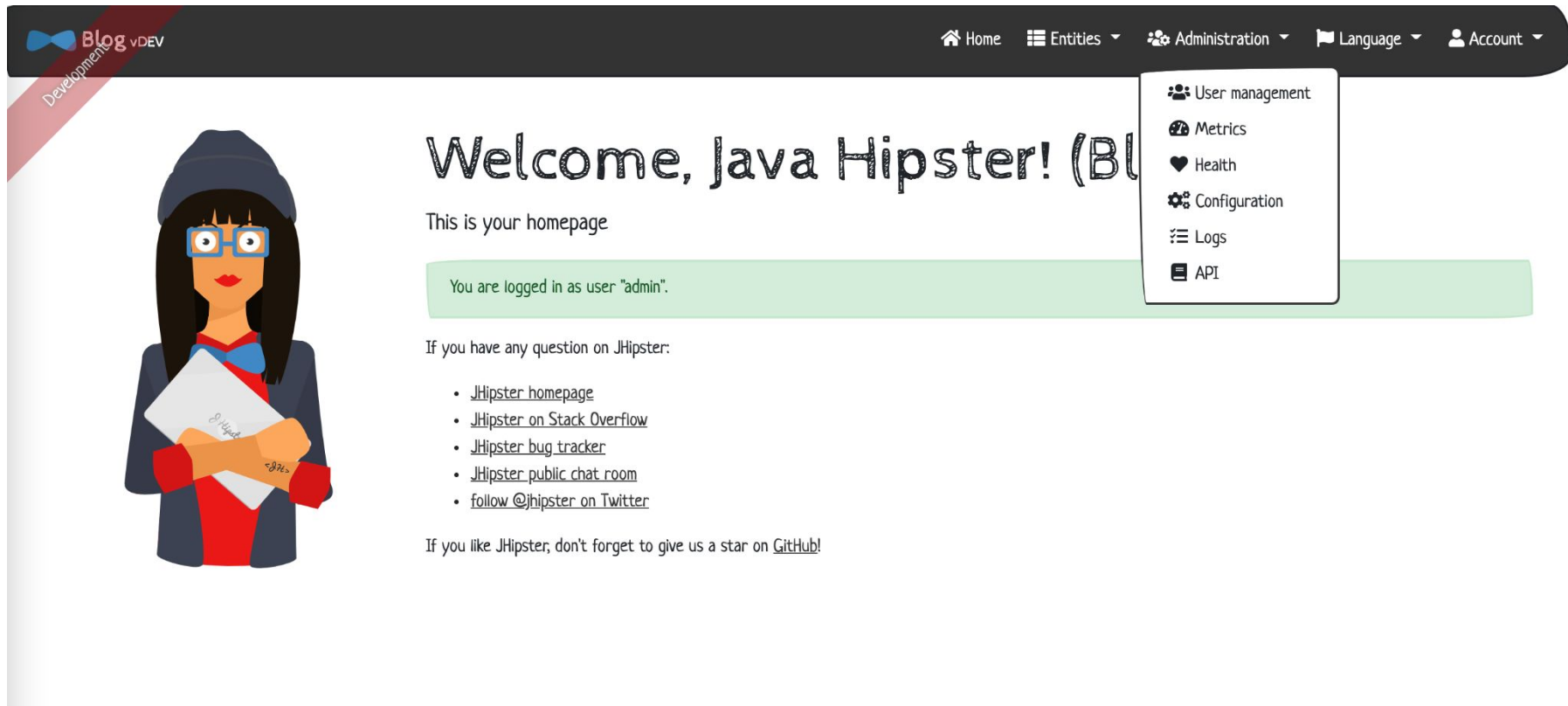
- En otra terminal corremos el frontend mediante el wrapper de npm

`~/app$./npmw start`

- Para realizar el debug del backend podemos ejecutar el proyecto desde el Spring Boot Dashboard en VSCode



Blog App - APPLICATION



Blog App - BLOQUES ARQUITECTURA

