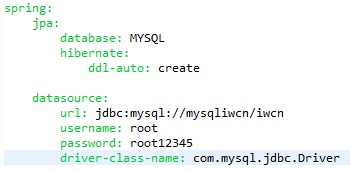
**Ingeniería Web y Computación en la Nube**

Tarea Docker Advanced

Nombre: Jonathan Ordoñez

**1. Creación de una propia imagen**

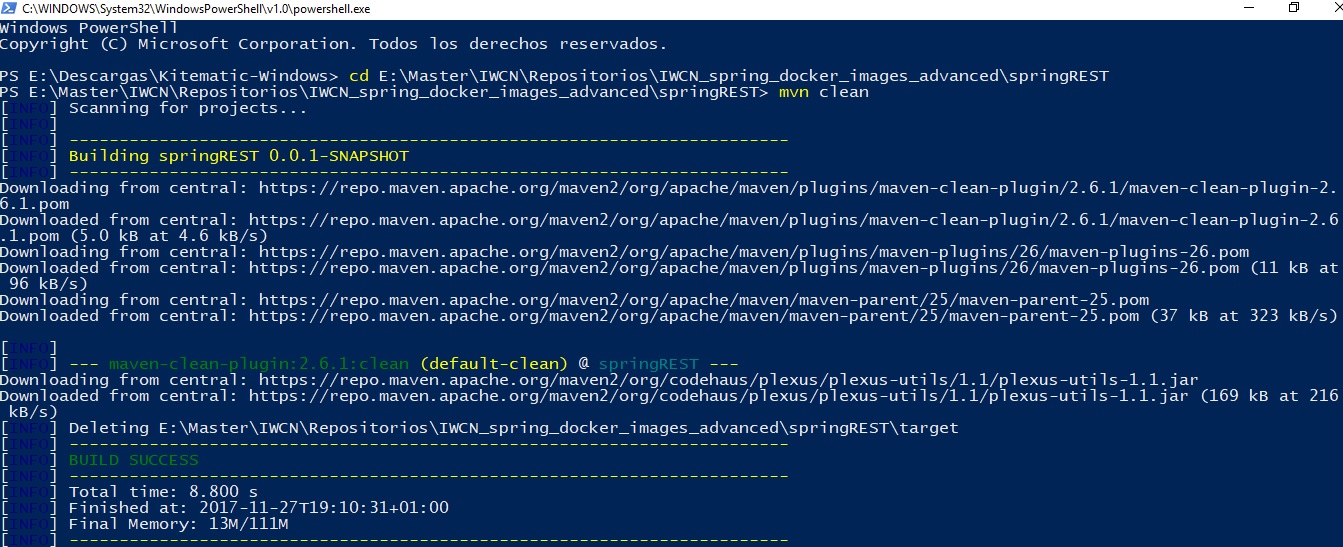
Partiendo del ejercicio anterior se cambia en la aplicación (servidor rest) la url del datasource en el fichero .yml el localhost:3306 por el nombre del contenedor que contiene la base de datos mysqliwcn.



Se ingresa a la ruta de la carpeta del proyecto y construimos el .jar de nuestra aplicación con Maven.

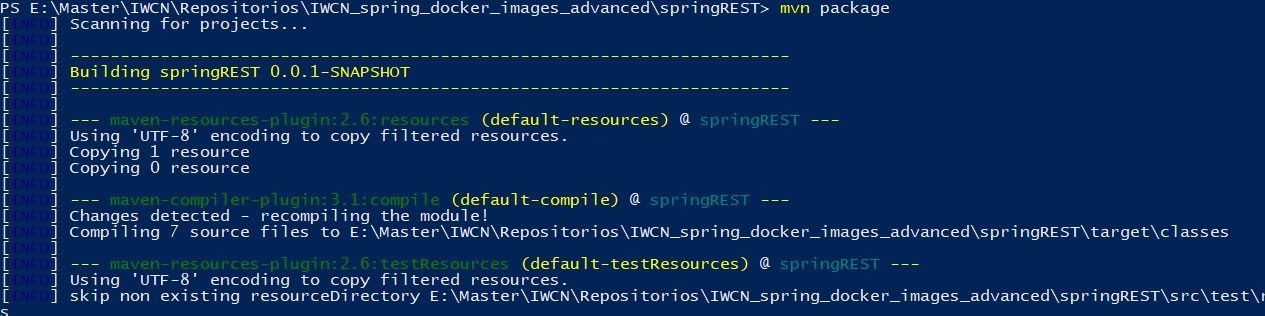
**mvn clean**

Este comando intenta limpiar los archivos y directorios generados por Maven durante su compilación. Si bien hay complementos que generan archivos adicionales

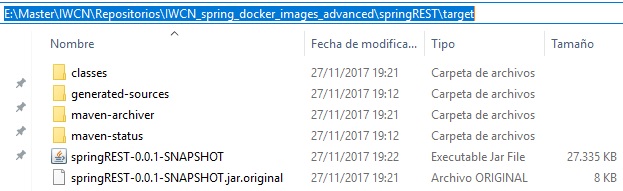


**mvn package**

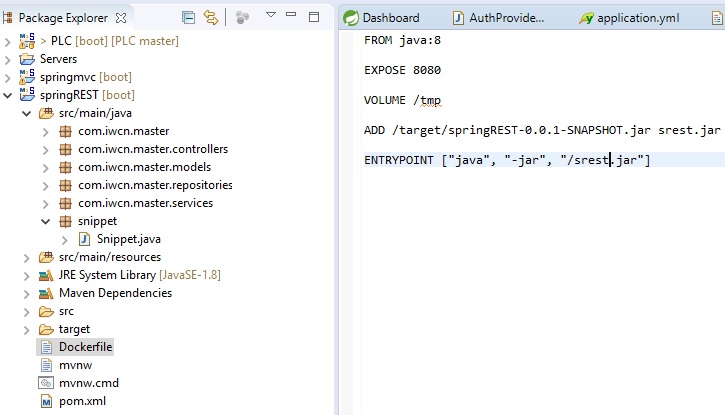
Este comando compila y empaqueta. En este caso, crea un .jar



Una vez generado el .jar podemos visualizarlo en la carpeta target, luego procedemos a crear fichero Dockerfile.



Para la creación del Dockerfile agregamos un fichero al proyecto con el siguiente con las siguientes líneas de código:



Configura la imagen base para usar.

**FROM java:8**

Expone un puerto al exterior.

**EXPOSE 8080**

Monta un directorio desde el host al contenedor.

**VOLUME /tmp**

Copia un archivo desde el host en el contenedor.

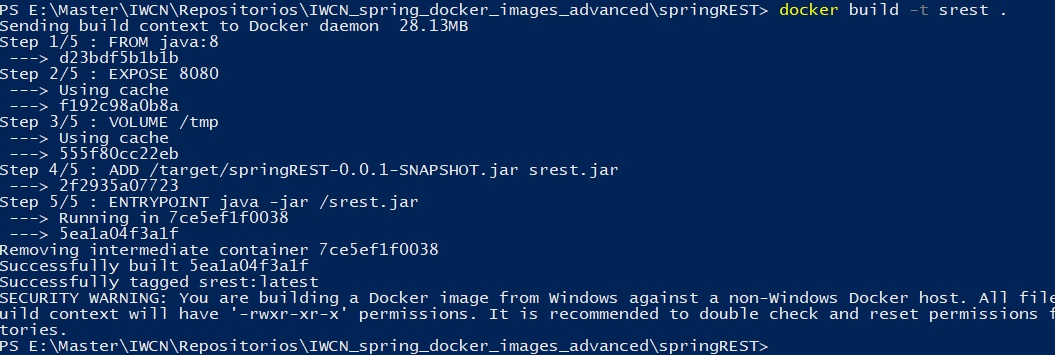
**ADD /target/springREST-0.0.1-SNAPSHOT.jar srest.jar**

Ajusta el punto de entrada por defecto de la aplicación desde el contenedor.

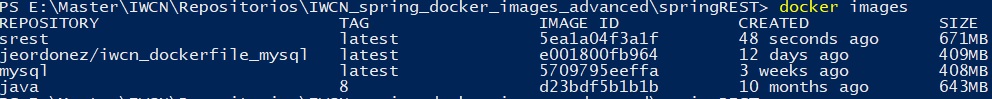
**ENTRYPOINT ["java", "-jar", "/ srest.jar"]**

Para construir la imagen a partir del fichero Dockerfile creado se ejecuta el comando:

**docker build –t srest .**

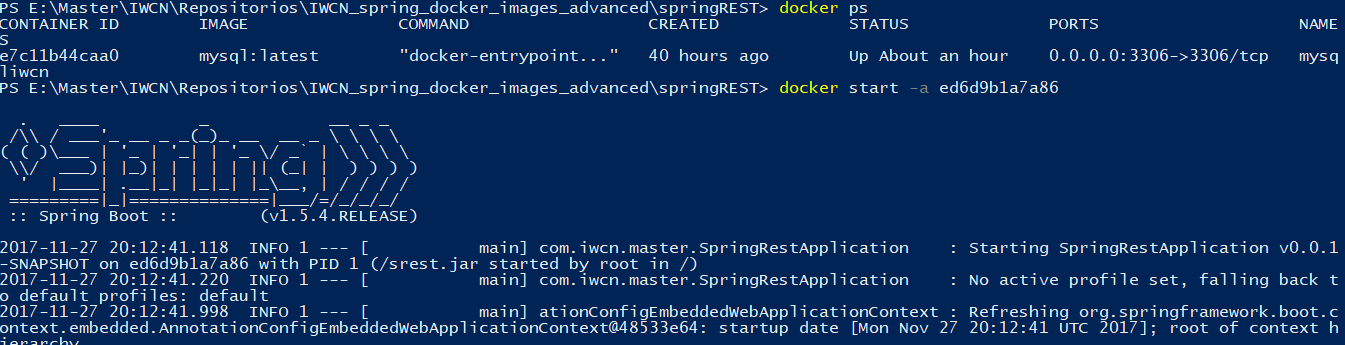


Hacemos un **docker images** para revisar que nuestra imagen se ha creado correctamente.

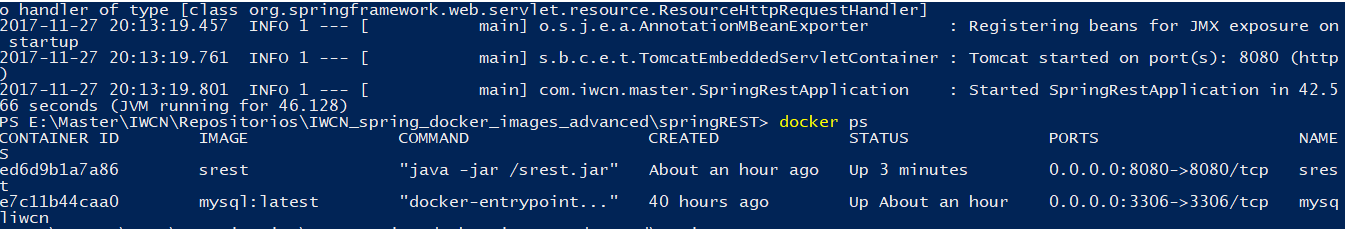


Para comprobar que el funcionamiento se inicia el contenedor a partir de la imagen

**docker start –a <id\_del\_contenedor>**

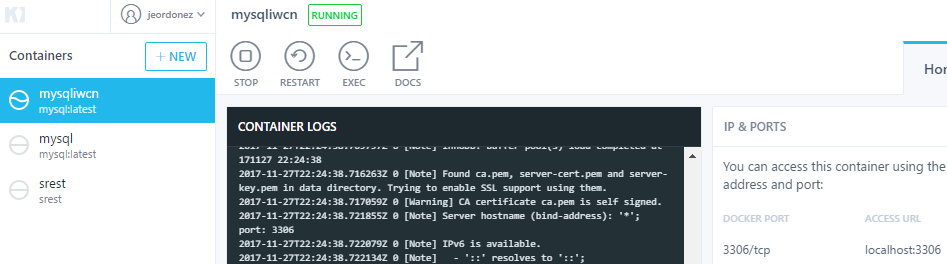


El contenedor se ha iniciado correctamente.



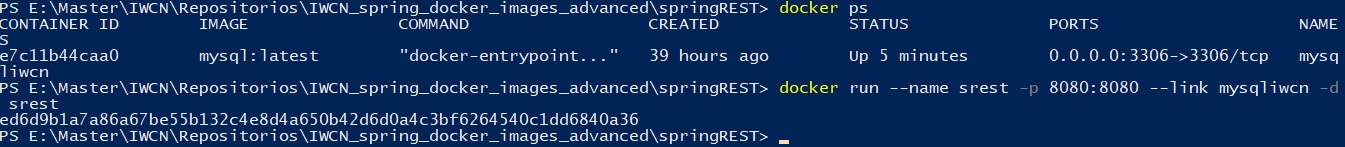
**2. Conectar dos contenedores**

Para conectar dos contenedores iniciamos el contenedor con la base de datos.

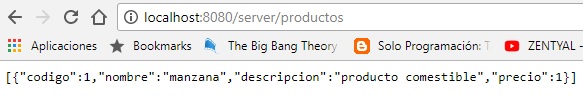


Luego iniciamos el contenedor con la aplicación mediante un link con el contenedor de la base de datos.

**docker run --name srest -p 8080:8080 --link mysqliwcn -d srest**

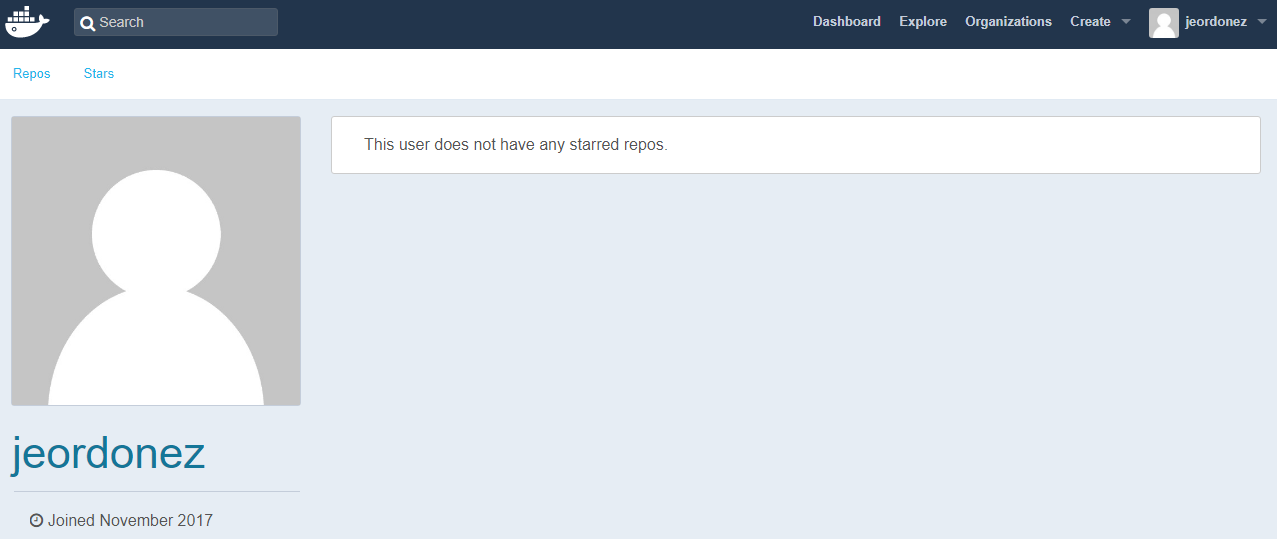


Ingresamos al navegador y comprobamos que existe comunicación entre los contenedores.



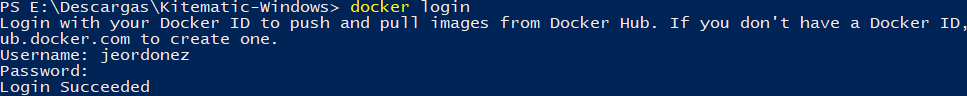
**3. Subir imágenes a DockerHub**

Creamos una cuenta en <https://hub.docker.com>



Iniciar las credenciales de Docker Hub con el comando:

**docker login**

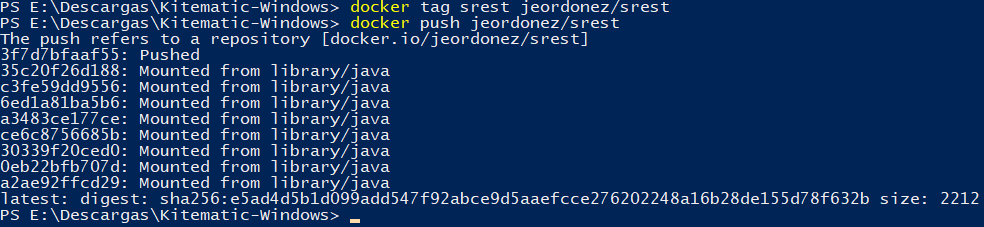


Agregamos una etiqueta para la imagen a subir con el comando:

**docker tag my\_image DOCKER\_ID\_USER/my\_image**

Finalmente se realiza un push hacia el repositorio para subir la imagen.

**docker push DOCKER\_ID\_USER/my\_image**



Comprobamos en el repositorio Docker Hub que se ha subido correctamente la imagen.

