

Packager une application client riche avec l'API

Nous allons apprendre comment packager une application frontend pour la compiler puis la servir avec notre API sur la route `/`. Nous allons prendre l'exemple d'une application Angular.

Tout d'abord, créez un dossier `frontend` dans `src/main/resources` et créez ou déplacez votre projet angular dedans. Ensuite dans votre fichier `angular.json` (situé à la racine de votre projet angular), cherchez `outputPath` et affectez y la valeur `../public`.

Cela va permettre de dire à votre projet Angular de stocker son résultat de compilation dans le dossier des fichiers statiquement servis par Spring Boot.

Ensuite, afin de compiler le projet Angular avec Maven, rajouter dans votre `pom.xml` le plugin suivant :

```
<plugin>
    <groupId>com.github.eirslett</groupId>
    <artifactId>frontend-maven-plugin</artifactId>
    <version>1.6</version>
    <configuration>
        <workingDirectory>src/main/resources/frontend</workingDirectory>
        <!-- where to install npm -->
        <installDirectory>${project.build.directory}/install</installDirectory>
    </configuration>
    <executions>
        <execution>
            <id>install-node-and-npm</id>
            <goals>
                <goal>install-node-and-npm</goal>
            </goals>
            <configuration>
                <nodeVersion>latest-v14.x</nodeVersion>
                <npmVersion>6.14.10</npmVersion>
            </configuration>
        </execution>
        <execution>
            <id>npm-install</id>
```

```
        <goals>
          <goal>npm</goal>
        </goals>
        <configuration>
          <arguments>install</arguments>
        </configuration>
      </execution>
    <execution>
      <id>build</id>
      <goals>
        <goal>npm</goal>
      </goals>
      <configuration>
        <arguments>run build</arguments>
      </configuration>
    </execution>
  </executions>
</plugin>
```

Voilà ! Maintenant, en tapant :

```
mvn clean package
```

Le projet Angular sera généré et packagé avec le projet Spring, qui servira statiquement le front sur la route `/`. Cela prend un peu de temps la première fois à cause de l'installer de NodeJS, de NPM et l'exécution de `npm install` qui téléchargem toutes les dépendances.

Revision #4

Created 2021-01-22 09:51:22 UTC by Arsène Lapostolet

Updated 2021-02-03 17:30:01 UTC by Arsène Lapostolet