

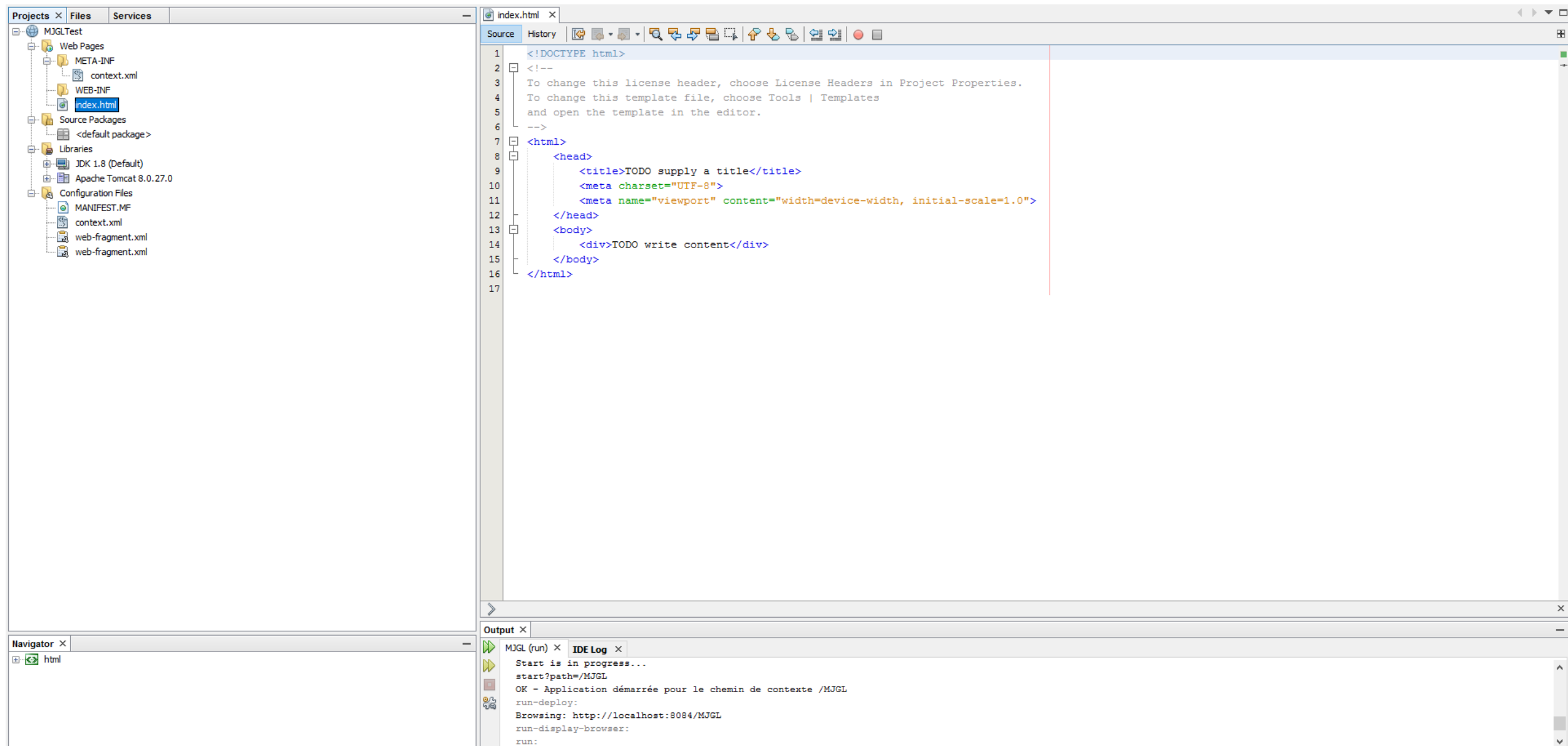
Apprehender MJGL

(Molecular Java GuideLine)

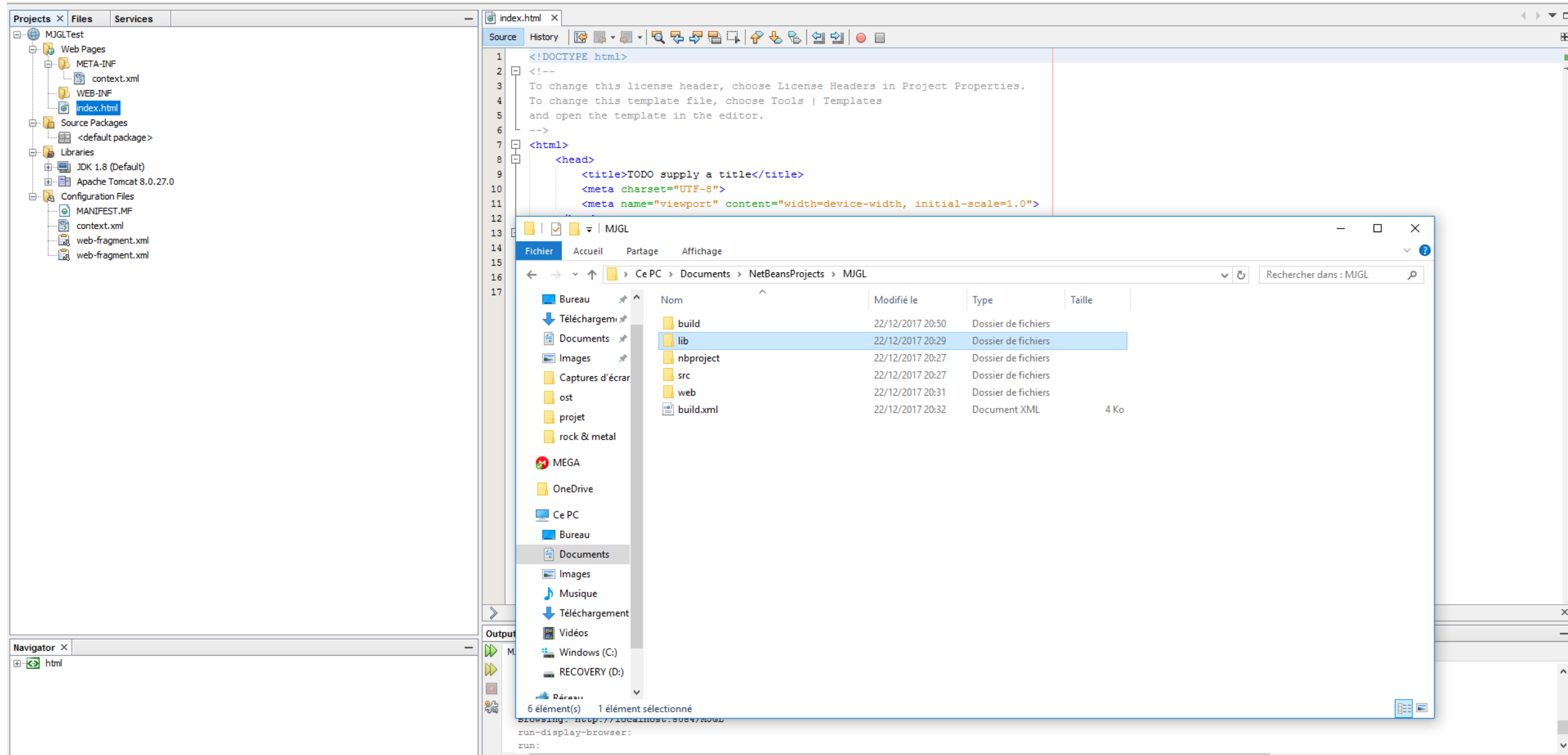
Créer un projet

- File > New Project > Java Web > Web Application
- Choisir le nom du projet (ici on va prendre “MJGLTest”)
- Tomcat 8.0.27 et Java EE 7 Web

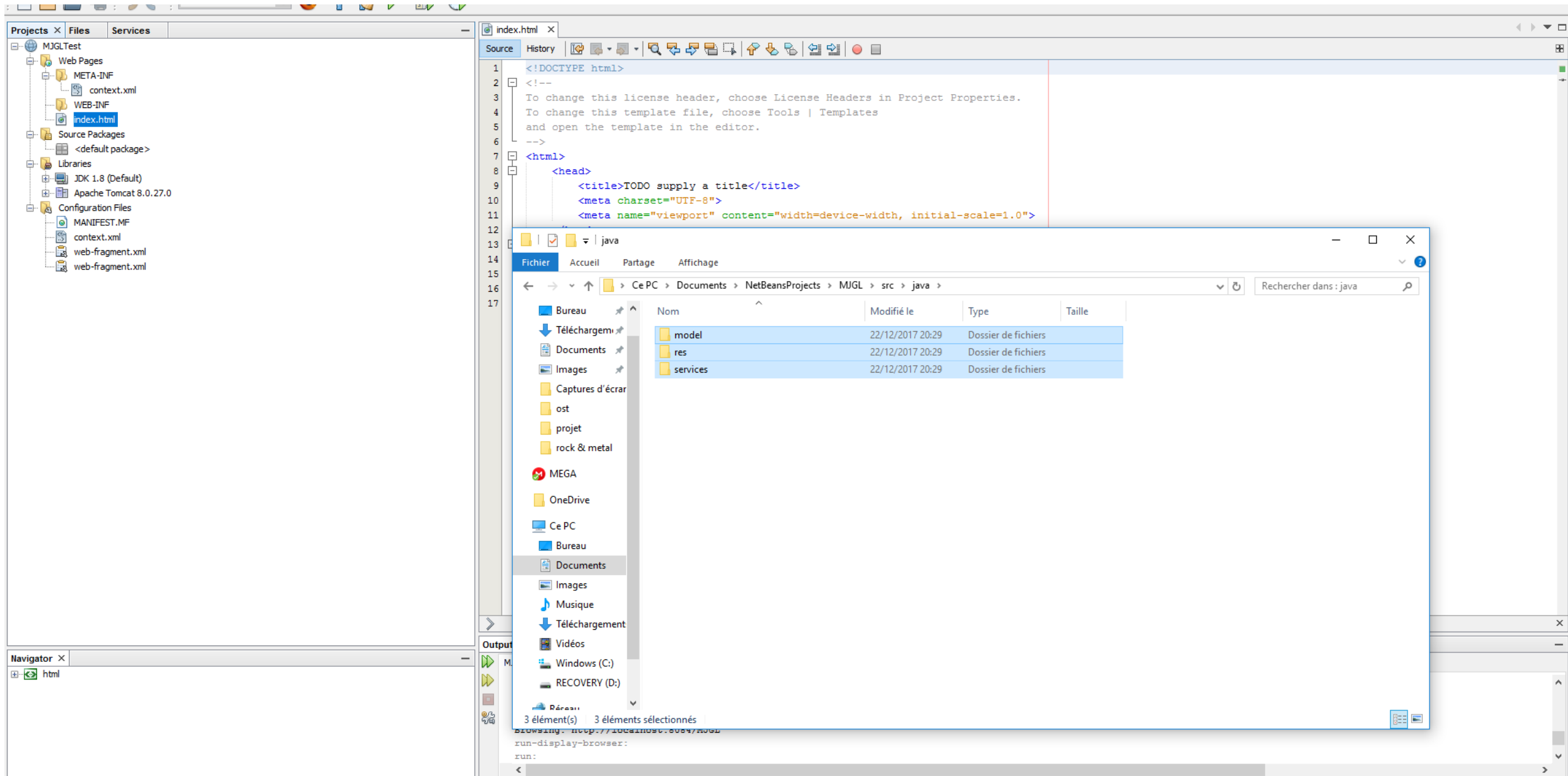
Vous devrez tomber sur quelque chose de ce genre



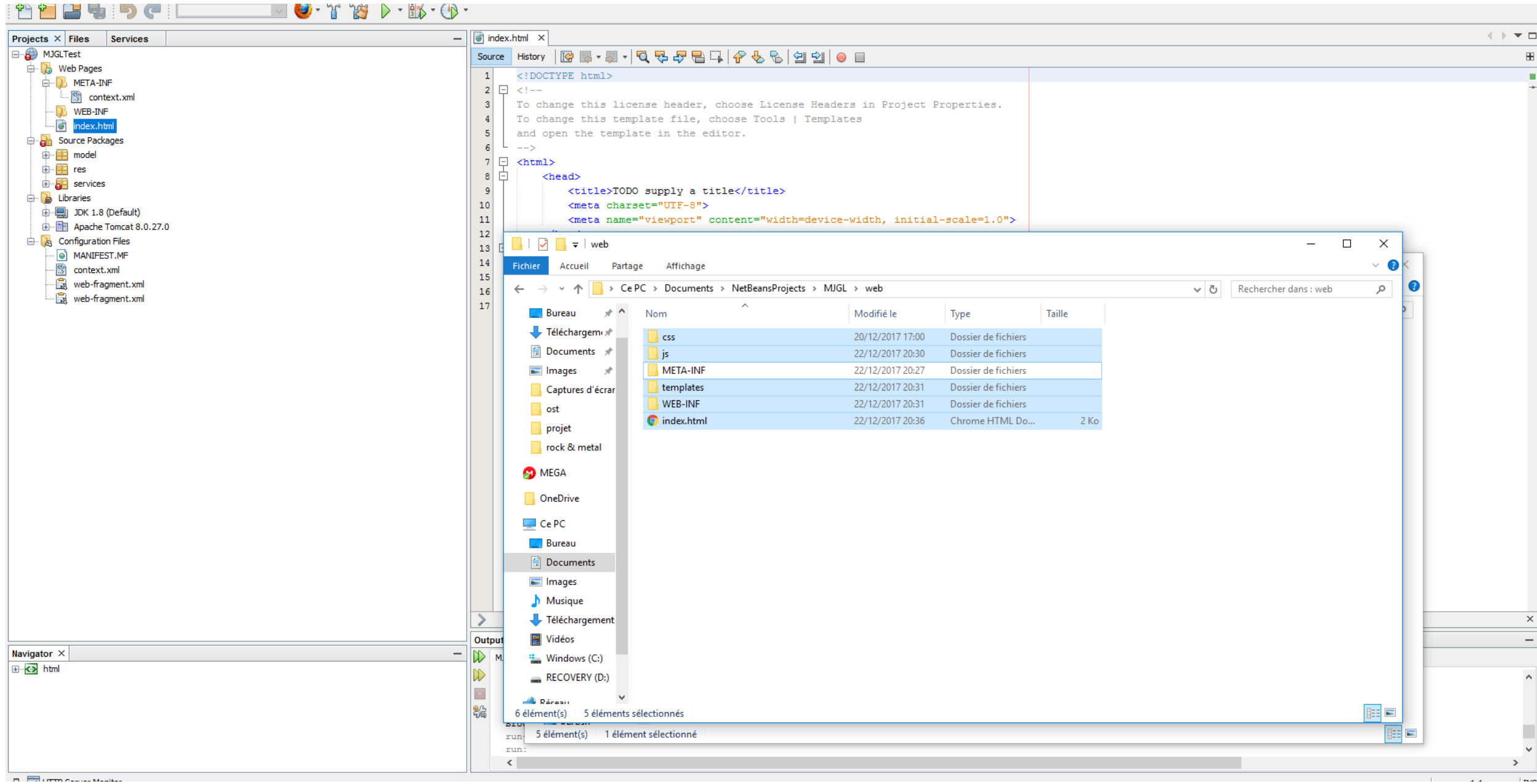
Alors , vous allez copier le dossier “lib” de notre framework dans votre projet



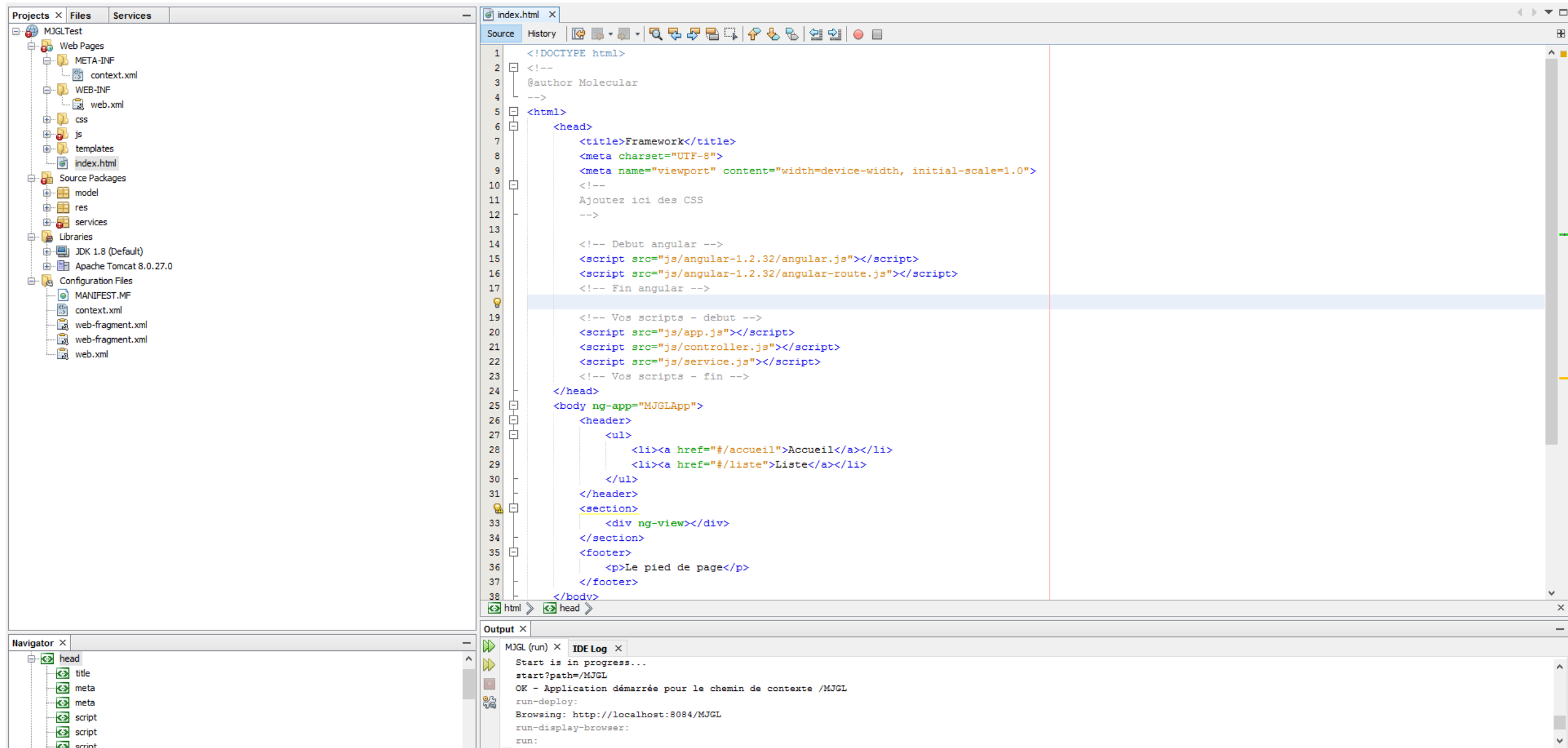
Copiez ensuite le contenu du repertoire “src/java” vers le repertoire du meme nom de notre projet



Copiez aussi le contenu du repertoire “web” de notre framework (a part “META-INF”) dans le meme repertoire de notre projet



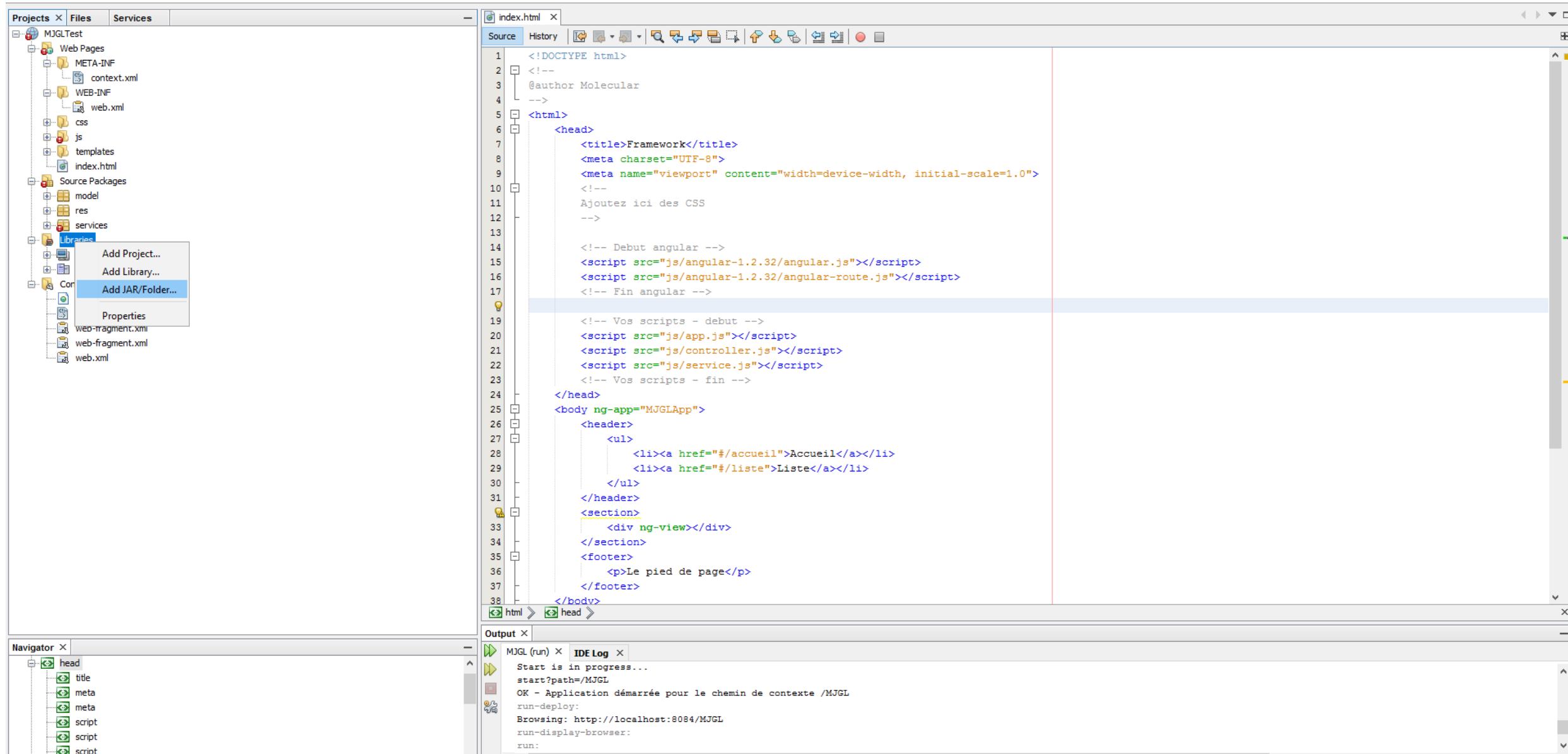
Et donc , en gros on devrait avoir quelque chose comme ca .



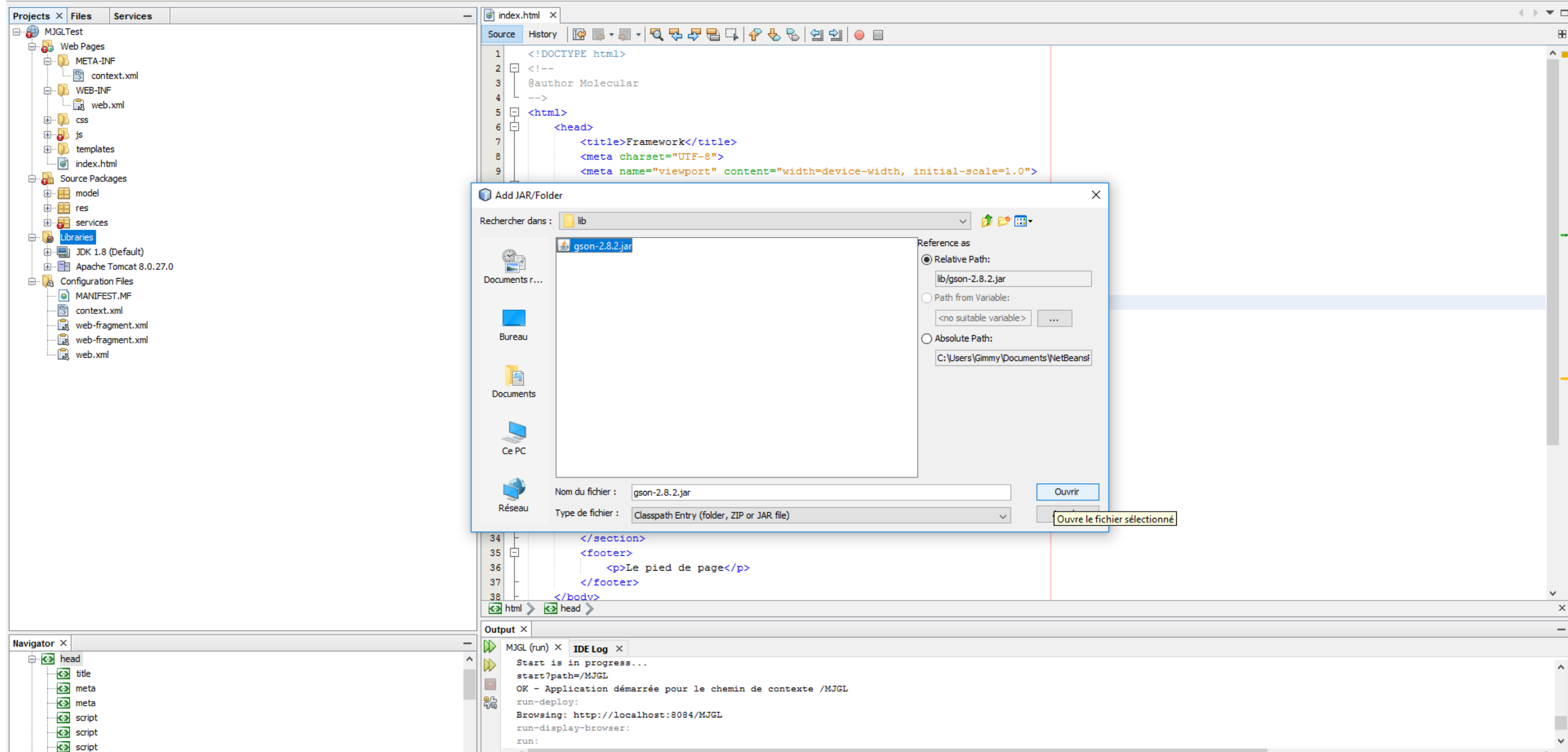
Petite explication de ce qu'on vient de faire

- Dans le dossier “lib” se trouve une librairie .JAR qui permet de traduire un objet Java en JSON
- Dans le dossier “model” se trouve (et doit se trouver) les classes model cote serveur de votre projet
- Dans le dossier “res” se trouve un petit objet qui va nous servir comme structure de reponse par le serveur aux clients
- Dans le dossier “services” se trouve (et doit se trouver) les classes controller cote serveur de votre projet (les servlet)
- Pour les autres , c’est pour les fichiers de view (sauf pour le web.xml)

Poursuivons ... Faites un clic-droit sur le dossier “Librairies” de notre projet dans notre IDE et choisissez “Add JAR/Folder”



Situez-vous dans le dossier “lib” que vous avez copie et selectionnez le fichier “gson-2.8.2.jar”



Et c'est pas fini ! Si vous voulez changer la valeur de l'attribut "ng-app", il faudra changer certaines choses dans d'autres fichiers aussi .

The screenshot displays an IDE interface with the following components:

- Project Explorer (Left):** Shows a project named "MJGLTest". It contains several folders: "Web Pages" (with "META-INF" and "WEB-INF"), "CSS", "JS", "templates", "index.html", "Source Packages" (with "model", "res", "services"), "Libraries" (with "gson-2.8.2.jar", "JDK 1.8 (Default)", "Apache Tomcat 8.0.27.0"), and "Configuration Files" (with "MANIFEST.MF", "context.xml", "web-fragment.xml", "web.xml").
- Source Editor (Center):** Displays the content of "index.html". The code is as follows:

```
1 <!DOCTYPE html>
2 <!--
3 @author Molecular
4 -->
5 <html>
6   <head>
7     <title>Framework</title>
8     <meta charset="UTF-8">
9     <meta name="viewport" content="width=device-width, initial-scale=1.0">
10    <!--
11    Ajoutez ici des CSS
12    -->
13
14    <!-- Debut angular -->
15    <script src="js/angular-1.2.32/angular.js"></script>
16    <script src="js/angular-1.2.32/angular-route.js"></script>
17    <!-- Fin angular -->
18
19    <!-- Vos scripts - debut -->
20    <script src="js/app.js"></script>
21    <script src="js/controller.js"></script>
22    <script src="js/service.js"></script>
23    <!-- Vos scripts - fin -->
24  </head>
25  <body ng-app="MJGLApp">
26    <header>
27      <ul>
28        <li><a href="#/accueil">Accueil</a></li>
29        <li><a href="#/liste">Liste</a></li>
30      </ul>
31    </header>
32    <section>
33      <div ng-view></div>
34    </section>
35    <footer>
36      <p>Le pied de page</p>
37    </footer>
38  </body>
```
- Output Window (Bottom):** Shows the execution log. It includes the following messages:
 - "Start is in progress..."
 - "start?path=/MJGL"
 - "OK - Application démarrée pour le chemin de contexte /MJGL"
 - "run-deploy:"
 - "Browsing: http://localhost:8084/MJGL"
 - "run-display-browser:"
 - "run:"

Comme ici (app.js)

The screenshot displays an IDE interface with the following components:

- Projects Panel (Left):** Shows a project named 'MJGLTest' with a file tree including 'Web Pages', 'META-INF', 'WEB-INF', 'css', 'js' (containing 'angular-1.2.32', 'app.js', 'controller.js', 'service.js'), 'templates', 'index.html', 'Remote Files', 'Source Packages', 'model', 'res', 'services', 'Libraries' (including 'gson-2.8.2.jar', 'JDK 1.8 (Default)', 'Apache Tomcat 8.0.27.0'), and 'Configuration Files' (including 'MANIFEST.MF', 'context.xml', 'web-fragment.xml', 'web.xml').
- Source Editor (Center):** Displays the 'app.js' file with the following code:

```
1 var MJGLApp = angular.module('MJGLApp', ['routeMJGL']);
2
3 var routeMJGL = angular.module('routeMJGL', ['ngRoute']);
4
5 routeMJGL.config(['$routeProvider', function($routeProvider){
6
7     $routeProvider
8       .when('/accueil', {
9         templateUrl: 'templates/accueil.html',
10        controller: 'ctrl-accueil'
11      })
12
13     .when('/liste', {
14       templateUrl: 'templates/liste.html',
15       controller: 'ctrl-liste'
16     })
17
18     .otherwise({
19       redirectTo: '/accueil'
20     });
21
22 }]);
```
- Find/Replace Bar (Bottom Center):** Shows a search for 'MJGL' with 6 matches. The 'Replace With' field contains 'MJGLTest'. Options include 'Previous', 'Next', 'Select', 'Replace', 'Replace All', 'Replace Backwards', and 'Preserve Case'.
- Navigator Panel (Bottom Left):** Lists the following instances:
 - MJGLApp : angular.module.angular-1_3_6_11749.moduleInstance
 - routeMJGL : angular.module.angular-1_3_6_11749.moduleInstance
- Output Panel (Bottom Right):** Shows the 'MJGL (run)' log with the following messages:
 - Start is in progress...
 - start?path=/MJGL
 - OK - Application démarrée pour le chemin de contexte /MJGL
 - run-deploy:
 - Browsing: http://localhost:8084/MJGL
 - run-display-browser:
 - run:

Et ici aussi (controller.js)

The screenshot displays an IDE interface for a project named MJGLTest. The left sidebar shows the project structure, including Web Pages, META-INF, WEB-INF, css, js, templates, and Remote Files. The main editor area shows the controller.js file, which contains two AngularJS controllers: 'ctrl-accueil' and 'ctrl-liste'. The 'ctrl-accueil' controller uses the \$http service to fetch data from 'http://localhost:8084/MJGL/index'. The 'ctrl-liste' controller uses the \$http service to fetch data from 'http://localhost:8084/MJGL/liste'. The IDE Log at the bottom shows the output of the run command, indicating a successful build and deployment.

```
1 MJGLApp.controller('ctrl-accueil', ['$scope', '$http', '$timeout', 'MJGLService', function($scope, $http, $timeout, MJGLService){
2
3   $http.get("http://localhost:8084/MJGL/index").success(function(result){
4
5     if(!result.error){
6
7       $scope.welcome = result.data;
8
9     }
10
11   });
12
13 })
14
15 .controller('ctrl-liste', ['$scope', '$http', '$timeout', 'MJGLService', function($scope, $http, $timeout, MJGLService){
16
17   $http.get("http://localhost:8084/MJGL/liste").success(function(result){
18
19     if(!result.error){
20
21       $scope.liste = result.data;
22
23     }
24
25   });
26
27 })];
```

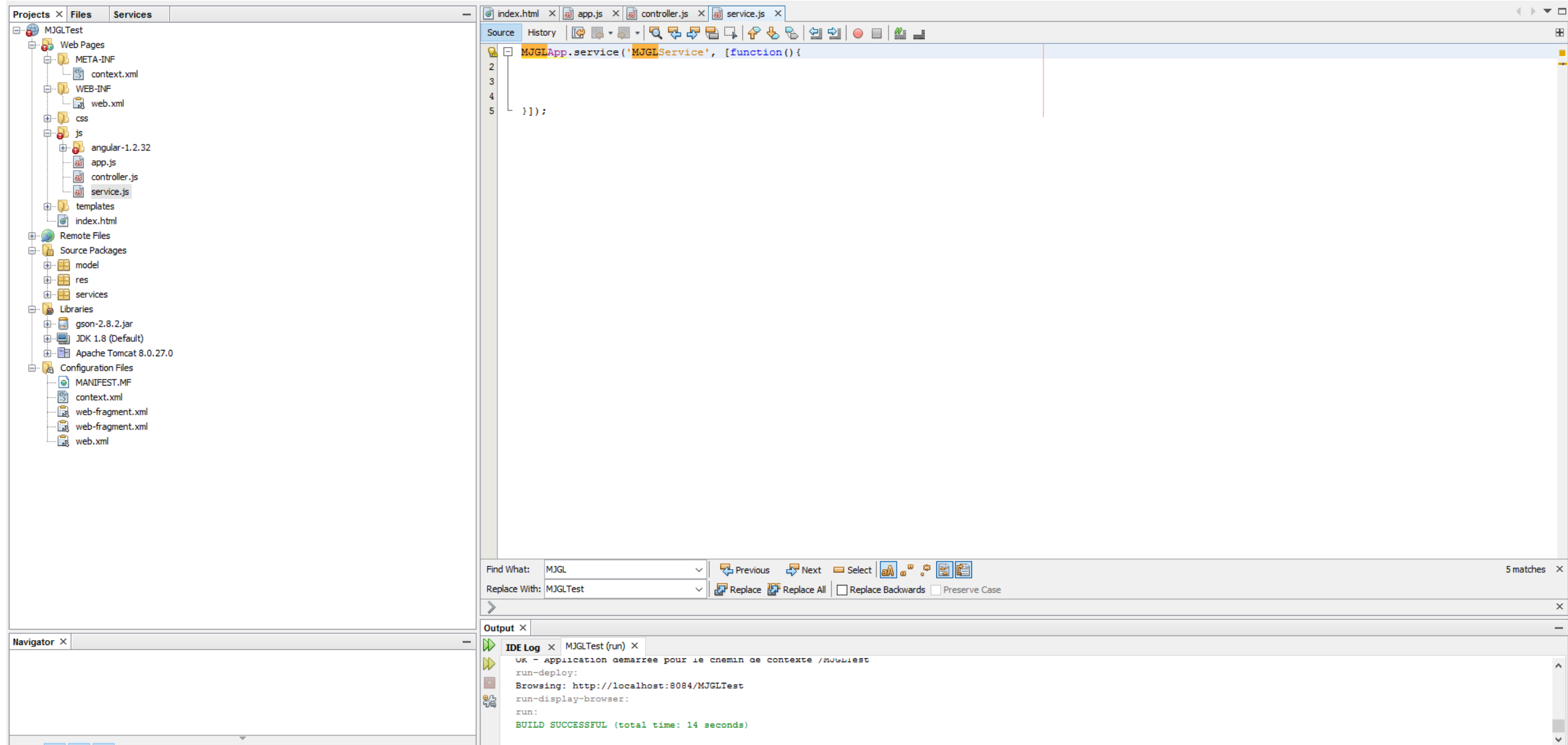
Find What: MJGL
Replace With: MJGLTest
2 of 7 matches

Output
IDE Log x MJGLTest (run) x
Completed incremental distribution of http://localhost:8084/MJGLTest
run-deploy:
Browsing: http://localhost:8084/MJGLTest
run-display-browser:
run:
BUILD SUCCESSFUL (total time: 2 seconds)

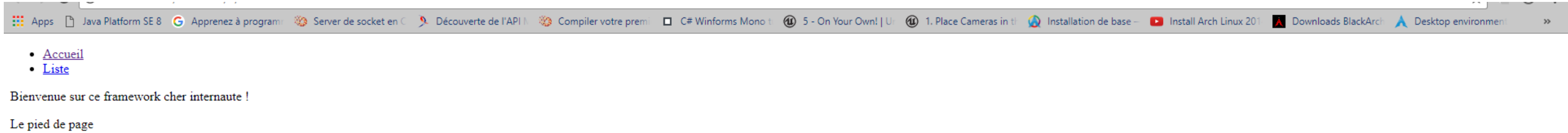
N.B pour controller.js :

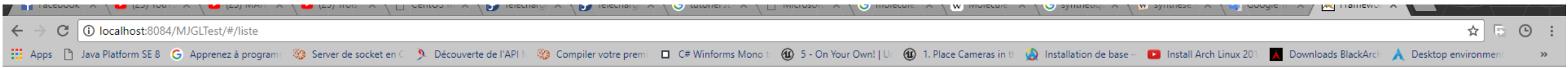
- A noter que meme si vous ne changer pas la valeur de l'attribut precedent , il faudra quand meme modifier la valeur de l'url de la method "get" du service "\$http" pour qu'il vise le web service de votre projet
- Pour utiliser la method "post" du service "\$http" , voyez donc ici comment on fait : [https://docs.angularjs.org/api/ng/service/\\$http](https://docs.angularjs.org/api/ng/service/$http)

Ici aussi il y a du changement a faire si on change la valeur de l'attribut precedent



Après , vous n'aurez plus qu'à exécuter votre projet pour obtenir quelque chose comme ça





- [Accueil](#)
- [Liste](#)

PR001 ~ Casquette BTS pour 12 Euros

PR002 ~ Cache bouche tokyo ghoul pour 9 Euros

PR003 ~ Bracelet DotA pour 5 Euros

Le pied de page



Derniere remarque

- Si vous avez de l'experience dans le Servlet , alors vous devez savoir que c'est dans le fichier "web.xml" du repertoire "web/WEB-INF" que vous allez rattacher une URL a un Servlet