

BTS SIO	SLAM4
	AngularJS

3.4 LE ROUTAGE ANGULAR.JS

Je rappelle qu'AngularJS est un framework de type **SPA** (Single Page Application) : cela signifie que pour l'utilisateur, la navigation paraît rapide, car il n'y a pas de téléchargement vers le serveur pour récupérer des pages.

Mais alors, comment une application JavaScript peut utiliser des liens et en même temps, ne rien charger à distance ?

Sans rentrer dans les détails techniques, il existe un moyen de dire au navigateur qu'il ne doit pas changer de page mais se déplacer dans la page : il s'agit d'une balise HTML appelée "ancree" et dont le symbole est #. L'usage des ancrs (# dans une URL) permet à Angular de traiter le lien URL, sans envoyer de requête au serveur.

3.4.1 Explications

Le **routing** est une des options les plus intéressantes dans tous les frameworks. Cependant, il faut charger d'autres modules et sources.

L'avantage de mettre en place des routes, est d'afficher une URL explicite, plutôt qu'une URL complexe.

- URL complexe : <http://monsite/index.php?id=17&profil=aR3ft&mode=vt>
- URL explicite : <http://monsite/user/adm/edit>

Peu importe si l'affichage de l'URL correspond à l'arborescence des dossiers, c'est la logique qui compte.

Pour cela, il faut établir une table de routage⁶. Par exemple :

URL	Page réelle
/consultation	/web/patient/consultation.html
/operation	/web/chu/service/operer.html
/accueil	/index.html

Le module **ngRoute** permet de créer les routes grâce au composant **\$routeProvider** : c'est le fournisseur (provider).

Ce module permet également de lire les URL demandées par le navigateur, grâce au composant **\$route** : c'est un service (nous l'appellerons rarement).

Il reste l'utilisation de la directive **ng-view** pour insérer le **template** de la page dans le **"layout"** général. C'est une directive, qui permet d'utiliser le même format pour l'ensemble des pages qui seront à l'intérieur du modèle (nous verrons cela plus loin dans l'exemple).

3 Finalement, il n'y a pas que les SISR qui font des tables de routage

BTS SIO	SLAM4
	AngularJS

3.4.2 Installation ngRoute

Comme pour AngularJS, nous allons profiter de l'installateur npm, **dans le répertoire principal du projet**, exécutez cette commande :

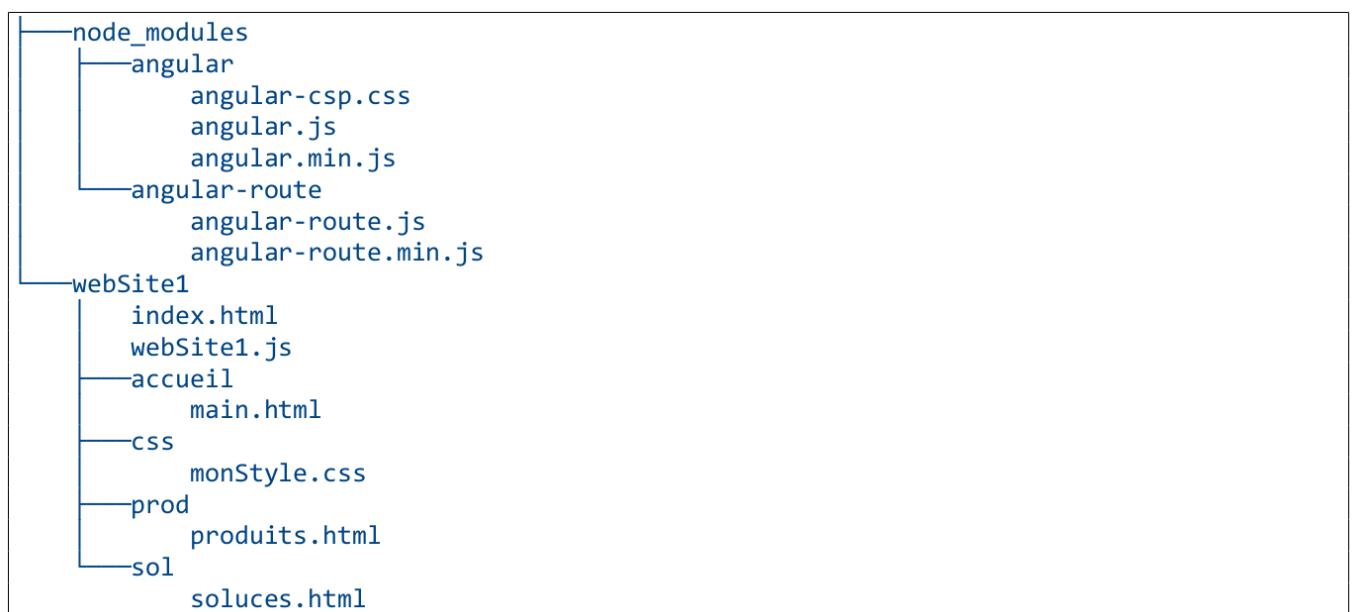
```
npm install --save angular-route
```

Dans le code HTML, il faudra alors ajouter le script angular-route.js qui sera dans le répertoire angular- route :

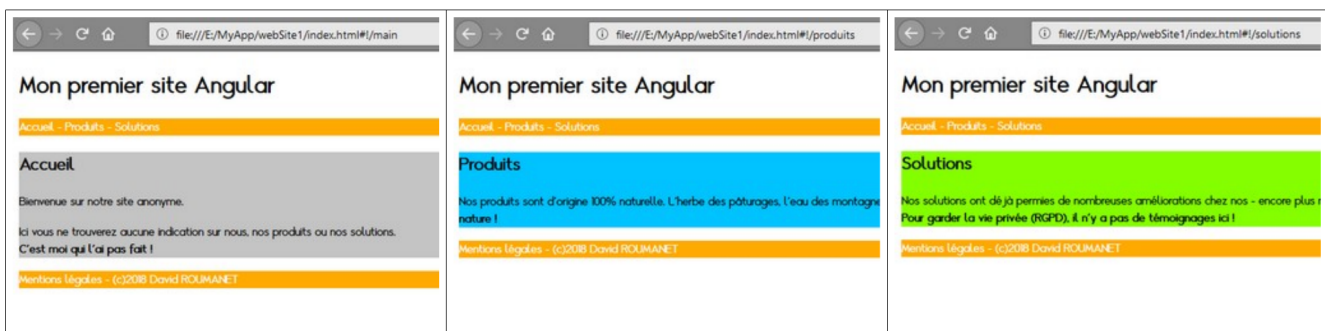
```
<script src="node_modules/angular-route/angular-route.js"></script>
```

3.4.3 Exemple de route

Nous allons créer une véritable application (dossier monApp), avec plusieurs pages, chacune étant dans un dossier différent :

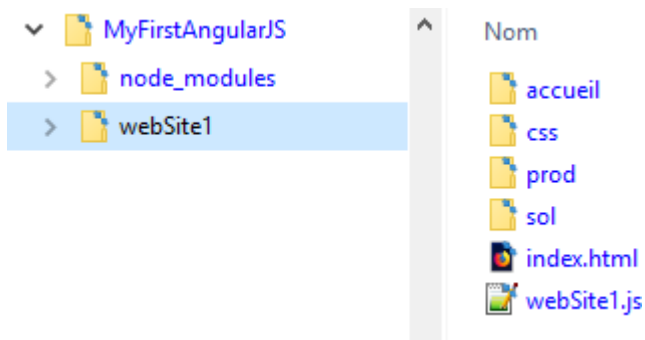


La page ressemblera à ceci, trois sous pages :



BTS SIO	SLAM4
	AngularJS

Pour créer cette application, nous allons utiliser l'arborescence suivante :



Le répertoire css ne contient qu'un fichier CSS pour l'ensemble des vues.

Accueil, prod et sol sont respectivement les vues pour l'accueil, les produits et les solutions (nous ajouterons un menu tarif qui ne sera pas développé pour montrer la route par défaut).

L'application doit être placée dans un **serveur WEB**. Voici le contenu des différents fichiers.



Index.html

```

website1 > <> index.html > ...
1  <!DOCTYPE html>
2  <html lang="en">
3  <head>
4      <meta charset="UTF-8">
5      <meta name="viewport" content="width=device-width, initial-scale=1.0">
6      <title>Accueil du site</title>
7      <script src="./node_modules/angular/angular.js"></script>
8      <script src="./node_modules/angular-route/angular-route.js"></script>
9      <script src="website1.js"></script>
10     <link rel="stylesheet" href="css/monStyle.css">
11 </head>
12 <body ng-app="website1">
13     <header>
14         <h1>Mon premier site Angularjs</h1>
15     </header>
16     <div class="menu">
17         <a href="#!main">Accueil</a> -
18         <a href="#!produits">Produits</a> -
19         <a href="#!solutions">Solutions</a> -
20         <a href="#!tarifs">Tarifs</a>
21     </div>
22     <ng-view>ici...</ng-view>
23     <footer class="menu">
24         Mentions légales - (c)2018 <a
25         href="mailto:chb.baudet@gmail.com">CBB</a>
26     </footer>
27
28 </body>
29 </html>

```

Le menu pointe vers des liens de type ancre : cette astuce permet de ne pas bliger le navigateur à apr et de rester sur la page... ce qui laisse Angular.JS faire le travail.

website1.js (le routeur)

website1 > JS website1.js > ...

```
1  var app = angular.module("website1",["ngRoute"]);
2  app.config(['$routeProvider', function($routeProvider) {
3      $routeProvider
4      .when("/main", {
5          templateUrl : "accueil/main.html"
6      })
7      .when("/produits", {
8          templateUrl : "prod/produits.html"
9      })
10     .when("/solutions", {
11         templateUrl : "sol/soluces.html"
12     })
13     .otherwise({
14         redirectTo : "/main"
15     });
16 }]);
```


Le routeur permet de décorréliser les répertoires et fichiers de l'arborescence dans l'URL, par exemple :

localhost/main affichera la vue dans le répertoire accueil/main.html

L'instruction otherwise permet de faire une route par défaut (pour les vues qui n'existent pas).

Placez le fichier suivant dans le répertoire CSS :

monStyle.css

```
website1 > css > # monStyle.css >  .menu
1  .menu {
2      color:  White;
3      background:  orange;
4  }
5      .menu a:link {
6      color:  white;
7      text-decoration: none;
8  }
9      .menu a:visited {
10     color:  white;
11     text-decoration: none;
12 }
13     .menu a:hover {
14     color:  black;
15     background:  yellow;
16 }
17     .produits {
18     background:  DeepSkyBlue;
19 }
20     .solutions {
21     background:  Chartreuse;
22 }
23     .main {
24     background:  Silver;
25 }
```

Les fichiers des vues en HTML :

accueil/main.html

website1 > accueil > <> main.html > section.main

```
1 <section class="main">
2   <h1>Accueil</h1>
3   <p>Bienvenue sur notre site anonyme.</p>
4   <p>Ici vous ne trouverez aucune indication sur nous, nos produits ou nos solutions.
5   <br />
6   <b>C'est moi qui l'ai pas fait !</b>
7   </p>
8 </section>
```

prod/produits.html

website1 > prod > <> produits.html > section.produits

```
1 <section class="produits">
2   <h1>Produits</h1>
3   <p>Nos produits sont d'origine 100% naturelle. L'herbe des
4   pâturages, l'eau des montagnes filtrée pendant des milliers
5   d'années. <b>Bref, le top de la nature !</b></p>
6 </section>
```

sol/soluces.html

website1 > sol > <> soluces.html > section.solutions

```
1 <section class="solutions">
2   <h1>Solutions</h1>
3   <p>Nos solutions ont déjà permis de nombreuses améliorations
4   chez nos - encore plus nombreux - clients. <br />
5   <b>Pour garder la vie privée (RGPD), il n'y a pas de témoignages ici !</b></p>
6 </section>
```

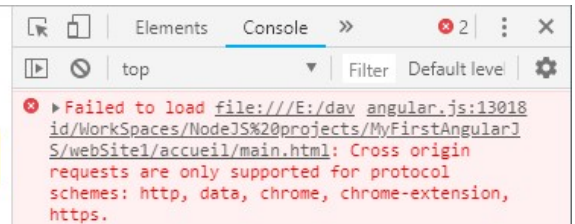
BTS SIO	SLAM4
	AngularJS

Si vous avez respecté l'arborescence, vous pouvez lancer le fichier index.html et tester le résultat.

Désormais, **les navigateurs luttent contre les insertions de codes ou pages provenant d'autres sites**. Malheureusement, si vous travaillez directement en Angular.JS sur votre disque-dur, vous risquez d'avoir le message d'erreur suivant :

Mon premier site Angular

Accueil - Produits - Solutions - Tarifs
Mentions légales - (c)2018 David ROUMANET



La solution est de lancer un serveur web (WAMP/XAMPP/UwAMP...) et de placer votre application Angular.JS à l'intérieur du répertoire public (généralement /www mais cela dépend de votre serveur).

BTS SIO	SLAM4
	AngularJS

4 ANNEXES

4.1 SOURCES

<https://weblogs.asp.net/dwahlin/video-tutorial-angularjs-fundamentals-in-60-ish-minutes> <https://docs.angularjs.org/api> <https://www.youtube.com/playlist?list=PL6n9fhu94yhWKHkcL7RJmmXyxkuFB3KSI>

Routage avec Angular.JS

<http://www.tutoriel-angularjs.fr/tutoriel/2-utilisation-complete-d-angularjs/1-le-routage#la-directive-ngView>

Information sur le CROSS Scripting (CORS) :

<https://medium.com/@tusharrungta18/the-above-image-is-almost-a-horror-movie-43a9df5afd9>