

# **Formation Angular 11**

---

**TP N°7: Configuration de la navigation entre les pages avec le module Routing.**

---

## SOMMAIRE

|    |  |    |
|----|--|----|
| 1. | Pré-requis : .....   | 3  |
| 2. | Objectifs .....  | 3  |
| 3. | Développement .....  | 3  |
| a. | Création de l'application.....   | 3  |
| b. | Installation de bootstrap .....  | 3  |
| c. | Installer jquery .....   | 3  |
| a. | Configurer les scripts et les feuilles CSS de bootstrap et jquery dans angular.json..... | 4  |
| b. | Ajouter le style du template utilisé dans styles.css.....                                | 4  |
| c. | Création du composant navbar .....   | 4  |
| d. | Création du composant login .....  | 5  |
| e. | Création des composants introduction, content et footer .....                            | 12 |
| f. | Création du composant home .....   | 15 |
| g. | Configuration des Routes .....   | 15 |
| h. | Tester votre application.....  | 16 |

### 1. Pré-requis :

- Réaliser les TPs 1-6 en premier.

### 2. Objectifs

- ✓ Intégrer bootstrap et jquery dans votre application.
- ✓ Utiliser un Template fourni par bootstrap.
- ✓ Utiliser un Template pour l'authentification fourni par bootstrap.
- ✓ Configurer les routes avec le module @angular/routing.

### 3. Développement

#### a. Création de l'application

Créer l'application tprouting :

```
PS C:\workspace_Angular> ng n tprouting
? Do you want to enforce stricter type checking and stricter bundle budgets in the workspace?
  This setting helps improve maintainability and catch bugs ahead of time.
  For more information, see https://angular.io/strict: Yes
? Would you like to add Angular routing? Yes
? Which stylesheet format would you like to use? CSS
CREATE tprouting/angular.json (3647 bytes)
CREATE tprouting/package.json (1199 bytes)
CREATE tprouting/README.md (1018 bytes)
```

#### b. Installation de bootstrap

```
PS C:\workspace_Angular> cd .\tprouting\
PS C:\workspace_Angular\tprouting> npm install bootstrap
npm WARN deprecated popper.js@1.16.1: You can find the new Popper v
v1
added 3 packages, and audited 1632 packages in 7s
```

#### c. Installer jquery

```
PS C:\workspace_Angular\tprouting> npm install jquery
up to date, audited 1632 packages in 3s
96 packages are looking for funding
run `npm fund` for details
```

a. Configurer les scripts et les feuilles CSS de bootstrap et jquery dans angular.json

```
"styles": [  
  "src/styles.css",  
  "./node_modules/bootstrap/dist/css/bootstrap.min.css"  
],  
"scripts": [  
  "./node_modules/bootstrap/dist/js/bootstrap.bundle.min.js",  
  "./node_modules/jquery/dist/jquery.min.js"  
]
```

b. Ajouter le style du template utilisé dans styles.css

Ajouter les lignes suivantes au niveau du fichier styles.css :

```
# styles.css M X  
tprouting > src > # styles.css > body  
1 | body {  
2 |   padding-top: 56px;  
3 | }
```

c. Création du composant navbar

- Créer le composant navbar comme suit :

```
PS C:\workspace_Angular\tprouting> ng g c template/navbar  
CREATE src/app/template/navbar/navbar.component.html (21 bytes)  
CREATE src/app/template/navbar/navbar.component.spec.ts (626 b  
CREATE src/app/template/navbar/navbar.component.ts (275 bytes)  
CREATE src/app/template/navbar/navbar.component.css (0 bytes)  
UPDATE src/app/app.module.ts (484 bytes)  
PS C:\workspace_Angular\tprouting>
```

- Modifier le contenu du fichier navbar.component.html comme suit :

```
<!-- Navigation -->  
<nav class="navbar navbar-expand-lg navbar-dark bg-dark fixed-top">  
  <div class="container">  
    <a class="navbar-brand" href="#">Formation Génie Logiciel</a>  
    <button class="navbar-toggler" type="button" data-toggle="collapse" data-  
target="#navbarResponsive" aria-controls="navbarResponsive" aria-  
expanded="false" aria-label="Toggle navigation">
```

```

    <span class="navbar-toggler-icon"></span>
  </button>
  <div class="collapse navbar-collapse" id="navbarResponsive">
    <ul class="navbar-nav ml-auto">
      <li class="nav-item">
        <a class="nav-link" href="#">Home</a>
      </li>

      <li class="nav-item">
        <a class="nav-link" href="#">About</a>
      </li>
      <li class="nav-item">
        <a class="nav-link" href="#">Services</a>
      </li>
      <li class="nav-item">
        <a class="nav-link" href="#">Contact</a>
      </li>
      <li class="nav-item active">
        <a class="nav-link" href="#">Login
          <span class="sr-only">(current)</span>
        </a>
      </li>
    </ul>
  </div>
</div>
</nav>

```

Lancer l'application avec la commande **ng serve** et observer le résultat :



#### d. Création du composant login

- Créer le composant login :

```
PS C:\workspace_Angular\tprouting> ng g c authentication/login
CREATE src/app/authentication/login/login.component.html (20 bytes)
CREATE src/app/authentication/login/login.component.spec.ts (619 bytes)
CREATE src/app/authentication/login/login.component.ts (271 bytes)
CREATE src/app/authentication/login/login.component.css (0 bytes)
UPDATE src/app/app.module.ts (577 bytes)
PS C:\workspace_Angular\tprouting>
```

- Pour le Template, nous allons utiliser le suivant :

<https://bootsnipp.com/snippets/dldxB>

- Aller au site ci-dessous et copier le contenu du Template au niveau de la page **login.component.html** :

```
<div class="wrapper fadeInDown">
  <div id="formContent">
    <!-- Tabs Titles -->

    <!-- Icon -->
    <div class="fadeIn first">
      
    </div>

    <!-- Login Form -->
    <form>
      <input type="text" id="login" class="fadeIn second" name="login" placeholder="login">
      <input type="text" id="password" class="fadeIn third" name="login" placeholder="password">
      <input type="submit" class="fadeIn fourth" value="Log In">
    </form>

    <!-- Remind Passowrd -->
    <div id="formFooter">
      <a class="underlineHover" href="#">Forgot Password?</a>
    </div>

  </div>
</div>
```

Ajouter une image (ici l'image logo.jpg à mettre dans le dossier *assets/logo*).

Copier le contenu de la feuille de style du Template dans le fichier **login.component.css** :

```

html {
  background-color: #56baed;
}

body {
  font-family: "Poppins", sans-serif;
  height: 100vh;
}

a {
  color: #92badd;
  display:inline-block;
  text-decoration: none;
  font-weight: 400;
}

h2 {
  text-align: center;
  font-size: 16px;
  font-weight: 600;
  text-transform: uppercase;
  display:inline-block;
  margin: 40px 8px 10px 8px;
  color: #cccccc;
}

/* STRUCTURE */

.wrapper {
  display: flex;
  align-items: center;
  flex-direction: column;
  justify-content: center;
  width: 100%;
  min-height: 100%;
  padding: 20px;
}

#formContent {
  -webkit-border-radius: 10px 10px 10px 10px;
  border-radius: 10px 10px 10px 10px;
  background: #fff;
  padding: 30px;
  width: 90%;
  max-width: 450px;
}

```

```

    position: relative;
    padding: 0px;
    -webkit-box-shadow: 0 30px 60px 0 rgba(0,0,0,0.3);
    box-shadow: 0 30px 60px 0 rgba(0,0,0,0.3);
    text-align: center;
}

#formFooter {
    background-color: #f6f6f6;
    border-top: 1px solid #dce8f1;
    padding: 25px;
    text-align: center;
    -webkit-border-radius: 0 0 10px 10px;
    border-radius: 0 0 10px 10px;
}

/* TABS */

h2.inactive {
    color: #cccccc;
}

h2.active {
    color: #0d0d0d;
    border-bottom: 2px solid #5fbae9;
}

/* FORM TYPOGRAPHY*/

input[type=button], input[type=submit], input[type=reset] {
    background-color: #56baed;
    border: none;
    color: white;
    padding: 15px 80px;
    text-align: center;
    text-decoration: none;
    display: inline-block;
    text-transform: uppercase;
    font-size: 13px;
    -webkit-box-shadow: 0 10px 30px 0 rgba(95,186,233,0.4);
    box-shadow: 0 10px 30px 0 rgba(95,186,233,0.4);
    -webkit-border-radius: 5px 5px 5px 5px;
    border-radius: 5px 5px 5px 5px;
    margin: 5px 20px 40px 20px;
    -webkit-transition: all 0.3s ease-in-out;

```



```

    -moz-transition: all 0.3s ease-in-out;
    -ms-transition: all 0.3s ease-in-out;
    -o-transition: all 0.3s ease-in-out;
    transition: all 0.3s ease-in-out;
}

input[type=button]:hover, input[type=submit]:hover, input[type=reset]:hover {
    background-color: #39ace7;
}

input[type=button]:active, input[type=submit]:active, input[type=reset]:active {
    -moz-transform: scale(0.95);
    -webkit-transform: scale(0.95);
    -o-transform: scale(0.95);
    -ms-transform: scale(0.95);
    transform: scale(0.95);
}

input[type=text] {
    background-color: #f6f6f6;
    border: none;
    color: #0d0d0d;
    padding: 15px 32px;
    text-align: center;
    text-decoration: none;
    display: inline-block;
    font-size: 16px;
    margin: 5px;
    width: 85%;
    border: 2px solid #f6f6f6;
    -webkit-transition: all 0.5s ease-in-out;
    -moz-transition: all 0.5s ease-in-out;
    -ms-transition: all 0.5s ease-in-out;
    -o-transition: all 0.5s ease-in-out;
    transition: all 0.5s ease-in-out;
    -webkit-border-radius: 5px 5px 5px 5px;
    border-radius: 5px 5px 5px 5px;
}

input[type=text]:focus {
    background-color: #fff;
    border-bottom: 2px solid #5fbae9;
}

input[type=text]:placeholder {
    color: #cccccc;
}

```

```

/* ANIMATIONS */

/* Simple CSS3 Fade-in-down Animation */
.fadeInDown {
  -webkit-animation-name: fadeInDown;
  animation-name: fadeInDown;
  -webkit-animation-duration: 1s;
  animation-duration: 1s;
  -webkit-animation-fill-mode: both;
  animation-fill-mode: both;
}

@-webkit-keyframes fadeInDown {
  0% {
    opacity: 0;
    -webkit-transform: translate3d(0, -100%, 0);
    transform: translate3d(0, -100%, 0);
  }
  100% {
    opacity: 1;
    -webkit-transform: none;
    transform: none;
  }
}

@keyframes fadeInDown {
  0% {
    opacity: 0;
    -webkit-transform: translate3d(0, -100%, 0);
    transform: translate3d(0, -100%, 0);
  }
  100% {
    opacity: 1;
    -webkit-transform: none;
    transform: none;
  }
}

/* Simple CSS3 Fade-in Animation */
@-webkit-keyframes fadeIn { from { opacity:0; } to { opacity:1; } }
@-moz-keyframes fadeIn { from { opacity:0; } to { opacity:1; } }
@keyframes fadeIn { from { opacity:0; } to { opacity:1; } }

.fadeIn {
  opacity:0;
  -webkit-animation:fadeIn ease-in 1;
  -moz-animation:fadeIn ease-in 1;
  animation:fadeIn ease-in 1;
}

```

```

    -webkit-animation-fill-mode: forwards;
    -moz-animation-fill-mode: forwards;
    animation-fill-mode: forwards;

    -webkit-animation-duration: 1s;
    -moz-animation-duration: 1s;
    animation-duration: 1s;
}

.fadeIn.first {
    -webkit-animation-delay: 0.4s;
    -moz-animation-delay: 0.4s;
    animation-delay: 0.4s;
}

.fadeIn.second {
    -webkit-animation-delay: 0.6s;
    -moz-animation-delay: 0.6s;
    animation-delay: 0.6s;
}

.fadeIn.third {
    -webkit-animation-delay: 0.8s;
    -moz-animation-delay: 0.8s;
    animation-delay: 0.8s;
}

.fadeIn.fourth {
    -webkit-animation-delay: 1s;
    -moz-animation-delay: 1s;
    animation-delay: 1s;
}

/* Simple CSS3 Fade-in Animation */
.underlineHover:after {
    display: block;
    left: 0;
    bottom: -10px;
    width: 0;
    height: 2px;
    background-color: #56baed;
    content: "";
    transition: width 0.2s;
}

.underlineHover:hover {
    color: #0d0d0d;
}

```

```

.underlineHover:hover:after{
  width: 100%;
}

/* OTHERS */

*:focus {
  outline: none;
}

#icon {
  width:60%;
}

```

#### e. Création des composants introduction, content et footer

- Créer les trois composants introduction, content et footer comme suit :

```

found 0 vulnerabilities
PS C:\workspace_Angular\tprouting> ng g c template/introduction
CREATE src/app/template/introduction/introduction.component.html (27 bytes)
CREATE src/app/template/introduction/introduction.component.spec.ts (668 bytes)
CREATE src/app/template/introduction/introduction.component.ts (299 bytes)
CREATE src/app/template/introduction/introduction.component.css (0 bytes)
UPDATE src/app/app.module.ts (775 bytes)
PS C:\workspace_Angular\tprouting> ng g c template/content
CREATE src/app/template/content/content.component.html (22 bytes)
CREATE src/app/template/content/content.component.spec.ts (633 bytes)
CREATE src/app/template/content/content.component.ts (279 bytes)
CREATE src/app/template/content/content.component.css (0 bytes)
UPDATE src/app/app.module.ts (870 bytes)
PS C:\workspace_Angular\tprouting> ng g c template/footer
CREATE src/app/template/footer/footer.component.html (21 bytes)
CREATE src/app/template/footer/footer.component.spec.ts (626 bytes)
CREATE src/app/template/footer/footer.component.ts (275 bytes)
CREATE src/app/template/footer/footer.component.css (0 bytes)
UPDATE src/app/app.module.ts (961 bytes)
PS C:\workspace_Angular\tprouting>

```

- Modifier les Template comme suit :

La page introduction.component.html :

```
<!-- Page Content -->
```

```

<!-- Heading Row -->
<div class="row align-items-center my-5">
  <div class="col-lg-7">
    
  </div>
  <!-- /.col-lg-8 -->
  <div class="col-lg-5">
    <h1 class="font-weight-light">Formation Génie Logiciel</h1>
    <p>Ce site contient un ensemble de formation pour la filiale Génie Logiciel. V
ous trouverez plusieurs modules :
      Java de base, Java avancé, Java EE, Spring BOOT, Angular, ...
    </p>
    <a class="btn btn-primary" href="#">Ajouter une formation</a>
  </div>
  <!-- /.col-md-4 -->
</div>
<!-- /.row -->

<!-- Call to Action Well -->
<div class="card text-white bg-secondary my-5 py-4 text-center">
  <div class="card-body">
    <p class="text-white m-0" style="font-size: xx-
large;">Apprendre les technologies Java EE & Spring BOOT & Angular</p>
  </div>
</div>

```

La page content.component.html :

```

<div class="row">
  <div class="col-md-4 mb-5">
    <div class="card h-100">
      <div class="card-body">
        <h2 class="card-title">Java de Base</h2>
        <p class="card-text">Apprendre la base de Java SE.</p>
      </div>
      <div class="card-footer">
        <a href="#" class="btn btn-primary btn-sm">Plus d'information</a>
      </div>
    </div>
  </div>
  <!-- /.col-md-4 -->
  <div class="col-md-4 mb-5">
    <div class="card h-100">
      <div class="card-body">

```

```

        <h2 class="card-title">Java EE</h2>
        <p class="card-text">JSF, JPA, Java WS, Java RS, JNDI</p>
    </div>
    <div class="card-footer">
        <a href="#" class="btn btn-primary btn-sm">Plus d'information</a>
    </div>
</div>
</div>
<!-- /.col-md-4 -->
<div class="col-md-4 mb-5">
    <div class="card h-100">
        <div class="card-body">
            <h2 class="card-title">Spring</h2>
            <p class="card-
text">Spring IOC, Spring AOP, Spring Data, Spring BOOT, ...</p>
        </div>
        <div class="card-footer">
            <a href="#" class="btn btn-primary btn-sm">Plus d'information</a>
        </div>
    </div>
</div>
<!-- /.col-md-4 -->

<!-- /.col-md-5 -->
<div class="col-md-4 mb-5">
    <div class="card h-100">
        <div class="card-body">
            <h2 class="card-title">Angular</h2>
            <p class="card-text">Angular Module, Angular Component, Angular Routing,
                Angular Forms, Angular Service, Angular Dependency Injection, ... </p>
        </div>
        <div class="card-footer">
            <a href="#" class="btn btn-primary btn-sm">Plus d'information</a>
        </div>
    </div>
</div>
<!-- /.col-md-5 -->

</div>
<!-- /.row -->

<!-- /.container -->

```

La page footer.component.html :

```

<footer class="py-5 bg-dark">
    <div class="container">
        <p class="m-0 text-center text-white">Formation CIGMA 2021</p>
    </div>
</footer>

```

```

</div>
<!-- /.container -->
</footer>

```

#### f. Création du composant home

- Créer le composant home :

```

UPDATE src/app/app.module.ts (577 bytes)
PS C:\workspace_Angular\tprouting> ng g c template/home
CREATE src/app/template/home/home.component.html (19 bytes)
CREATE src/app/template/home/home.component.spec.ts (612 bytes)
CREATE src/app/template/home/home.component.ts (267 bytes)
CREATE src/app/template/home/home.component.css (0 bytes)
UPDATE src/app/app.module.ts (660 bytes)
PS C:\workspace_Angular\tprouting>

```

Modifier la page *home.component.html* comme suite :

```

<div class="container">
  <app-introduction></app-introduction>
  <app-content></app-content>
</div>
<app-footer></app-footer>

```

#### g. Configuration des Routes

- Modifier le fichier *app-routing.module.ts* comme suit :

```

import { NgModule } from '@angular/core';
import { RouterModule, Routes } from '@angular/router';
import { LoginComponent } from '../authentication/login/login.component';
import { HomeComponent } from '../template/home/home.component';

const routes: Routes = [
  { path: 'login', component: LoginComponent },
  { path: 'home', component: HomeComponent }
];

@NgModule({
  imports: [RouterModule.forRoot(routes)],
  exports: [RouterModule]
})
export class AppRoutingModule { }

```

- Modifier ensuite la page **app.component.html** comme suit :

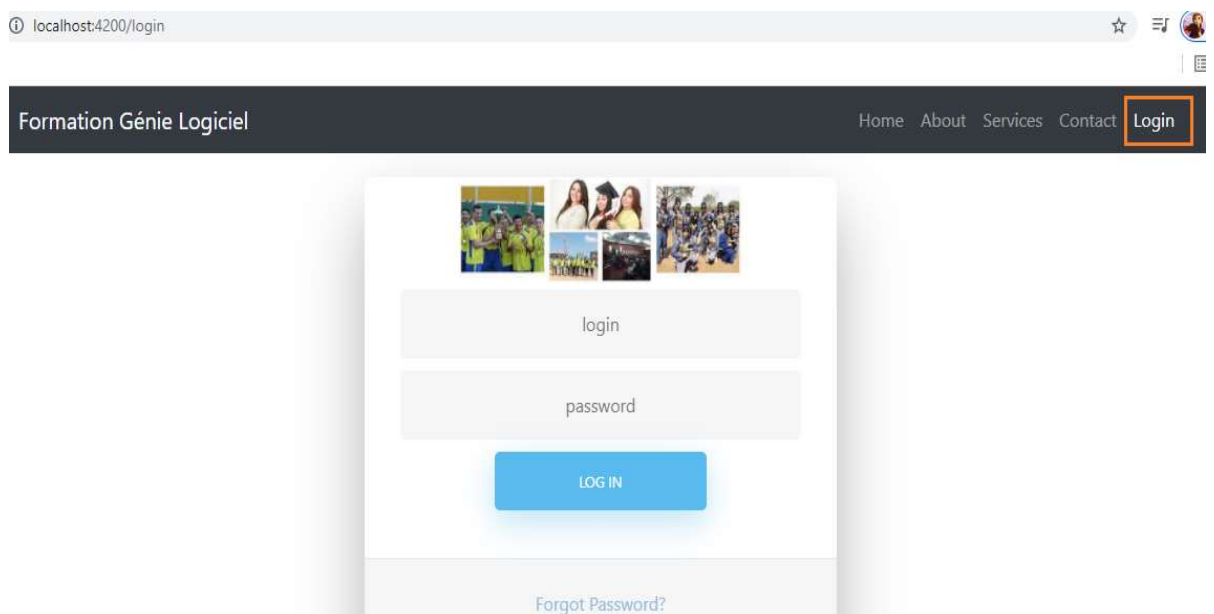
```
<app-navbar></app-navbar>  
<router-outlet></router-outlet>
```

#### h. Tester votre application

- Lorsque vous lancez l'URL <http://localhost:4200>, le composant navbar est affiché comme suit :



- Lorsque vous cliquez sur le menu « Login », le composant Login est affiché :



- Lorsque vous cliquez sur le menu Home, le composant Home est affiché :



## Formation Génie Logiciel

Ce site contient un ensemble de formation pour la filiale Génie Logiciel. Vous trouverez plusieurs modules : Java de base, Java avancé, Java EE, Spring BOOT, Angular, ...

[Ajouter une formation](#)

### Apprendre les technologies Java EE & Spring BOOT & Angular

#### Java de Base

Apprendre la base de Java SE.

[Plus d'information](#)

#### Java EE

JSF, JPA, Java WS, Java RS, JNDI

[Plus d'information](#)

#### Spring

Spring IOC, Spring AOP, Spring Data, Spring BOOT, ...

[Plus d'information](#)

Vous venez de configurer la navigation entre vos composants !

Vous pouvez passer au TP n° 8.

**Fin du TP 7.**