



COMPTE-RENDU DERNIER TP JEE

TRINÔME:

FATMA LARIBI

NADINE BOUKADIDA

NADER OUERDIANE

Un petit aperçu sur Spring Boot:

Spring Boot provides a good platform for Java developers to develop a stand-alone and production-grade spring application that you can just run. You can get started with minimum configurations without the need for an entire Spring configuration setup.

Advantages

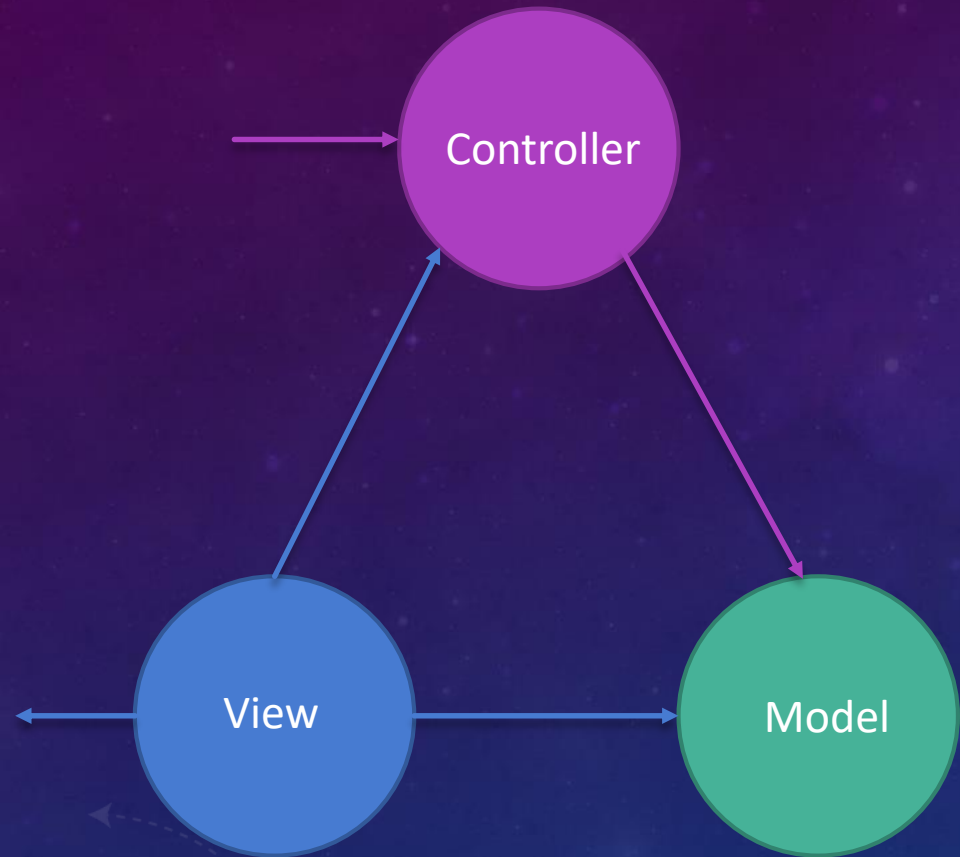
- Easy to understand and develop spring applications
- Increases productivity
- Reduces the development time

Goals

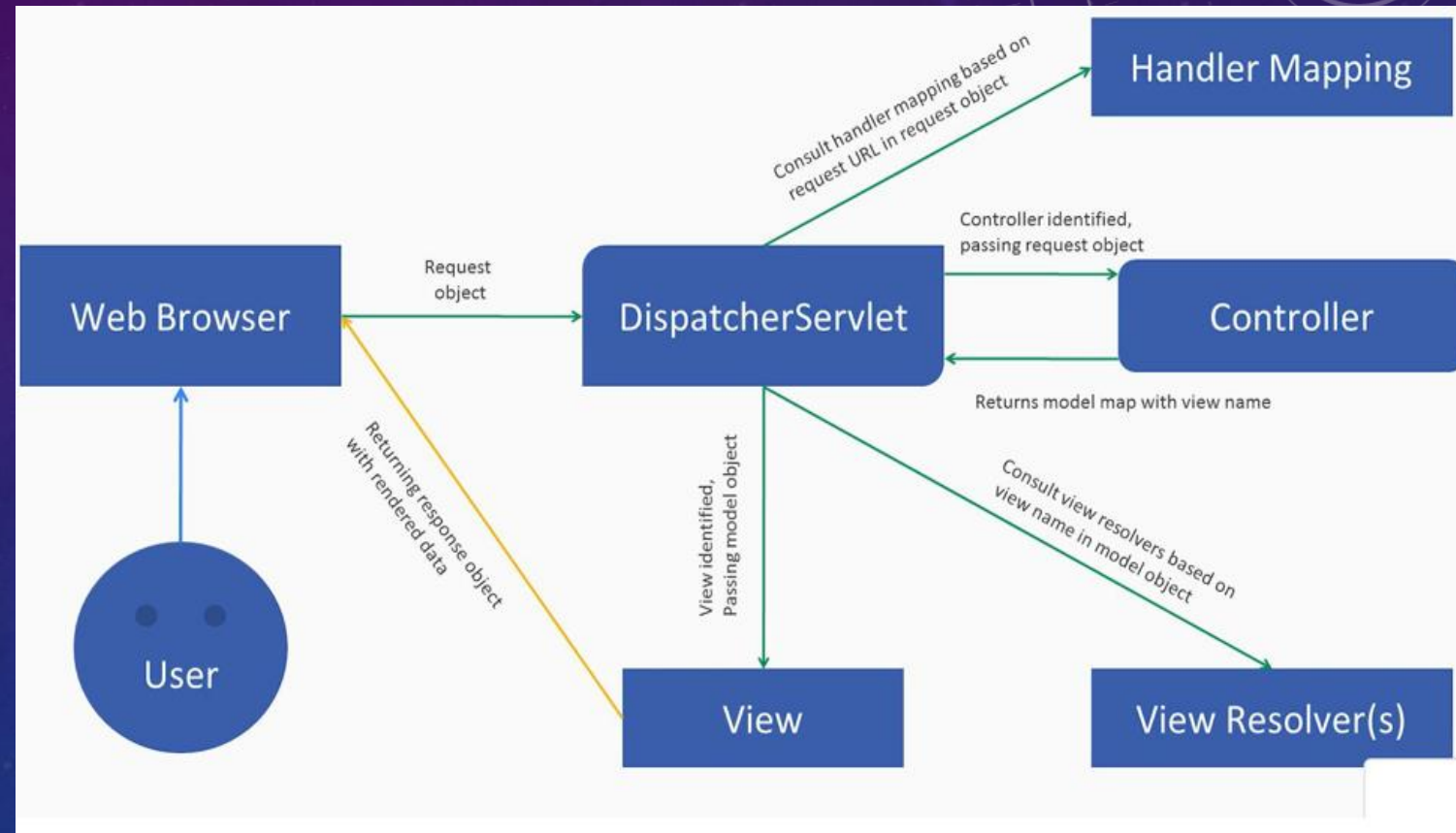
- To avoid complex XML configuration in Spring
- To develop a production ready Spring applications in an easier way
- To reduce the development time and run the application independently
- Offer an easier way of getting started with the application

MVC I & MVC II

L'architecture MVC

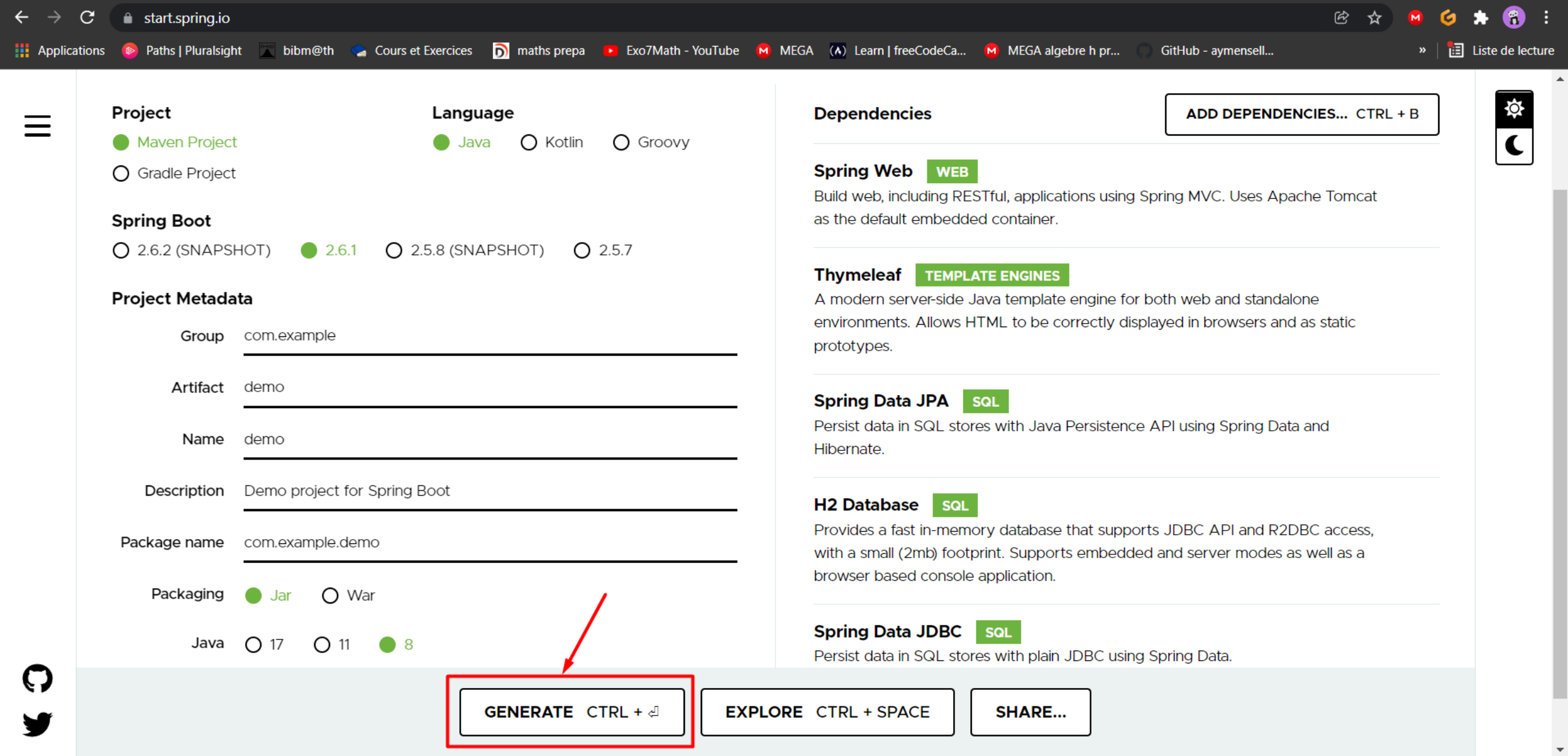


L'architecture MVC II utilisée par Spring MVC et Spring Boot



Il y a maintenant un contrôleur générique DispatcherServlet qui traite toutes les requêtes et fait le routage vers les contrôleurs spécifiques.

On commence par générer le projet Spring avec Spring initializr en précisant les dépendances: Spring Web, Thymeleaf, Spring Data JPA, H2 Database, et Spring Data JDBC



Différence entre le passage du nom de la vue et la redirection:

Grâce au redirect on passe par la méthode associée au path passé.

Si on passe simplement le nom de la vue on y accède sans faire aucun autre traitement

Les dépendances

H2 database:

une base de données in-memory c'est-à-dire qu'elle s'efface si on arrête le serveur

Spring Web:

The single spring-boot-starter-web dependency **transitively pulls in all dependencies related to web development**. It also reduces the build dependency count.

Thymeleaf:

Thymeleaf is a Java-based library used **to create a web application**. It provides a good support for serving a XHTML/HTML5 in web applications. *Il utilise des fragment pour former une page (structure: th:fragment="search-fragment")*

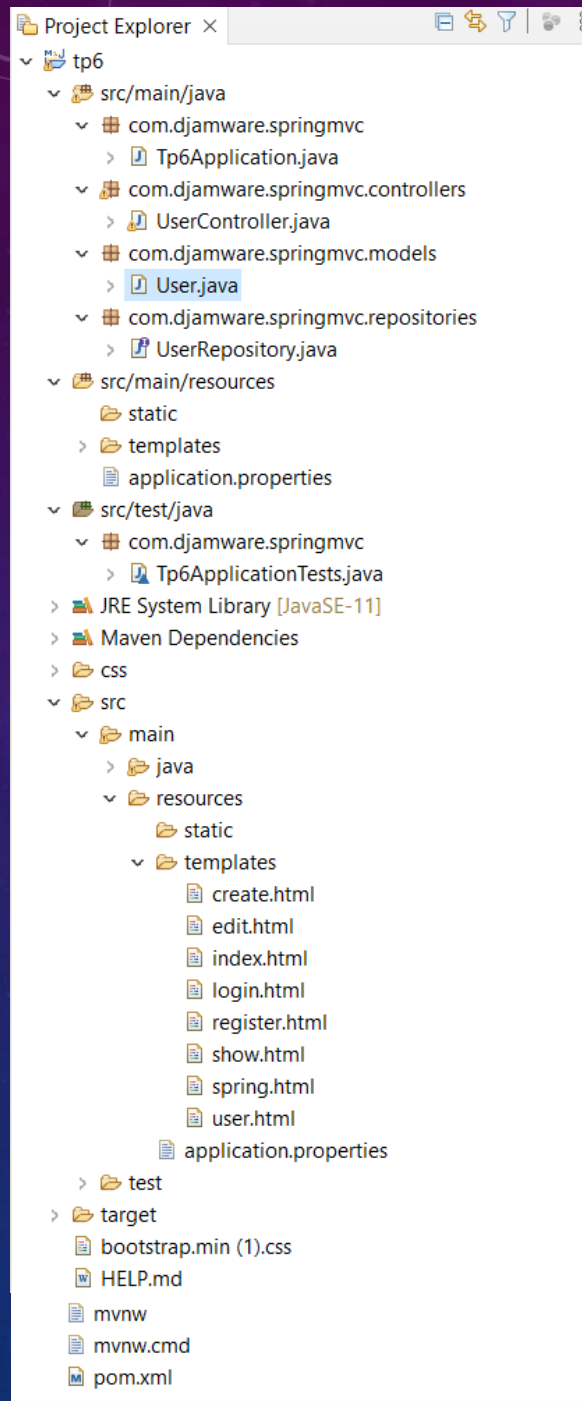
Spring Data JPA:

Spring Data JPA est une bibliothèque de Spring. En accord avec Spring Data JPA nous devons simplement définir une interface étendue de l'interface *Repository<T,ID>*, et déclarer des méthodes pour manipuler des données de cette Entity. Ce Spring Data JPA créera une classe qui implémente (implements) cette interface pour nous, C'est pourquoi dans le TP précédent on a pu utiliser findByProdName directement

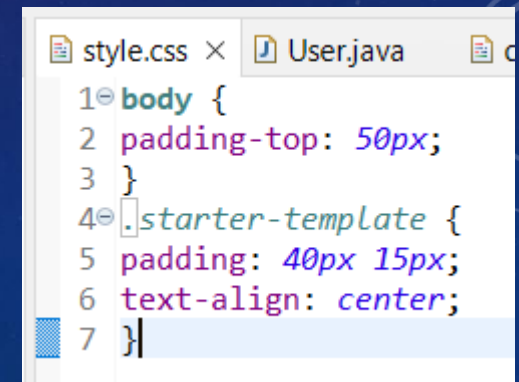
Spring Data JDBC:

To access the Relational Database by using JdbcTemplate in Spring Boot application, we need to add the Spring Boot Starter JDBC dependency in our build configuration file. Then, **if you @Autowired the JdbcTemplate class**, Spring Boot automatically connects the Database and sets the Datasource for the JdbcTemplate object.

Structure du Projet



css/style.css

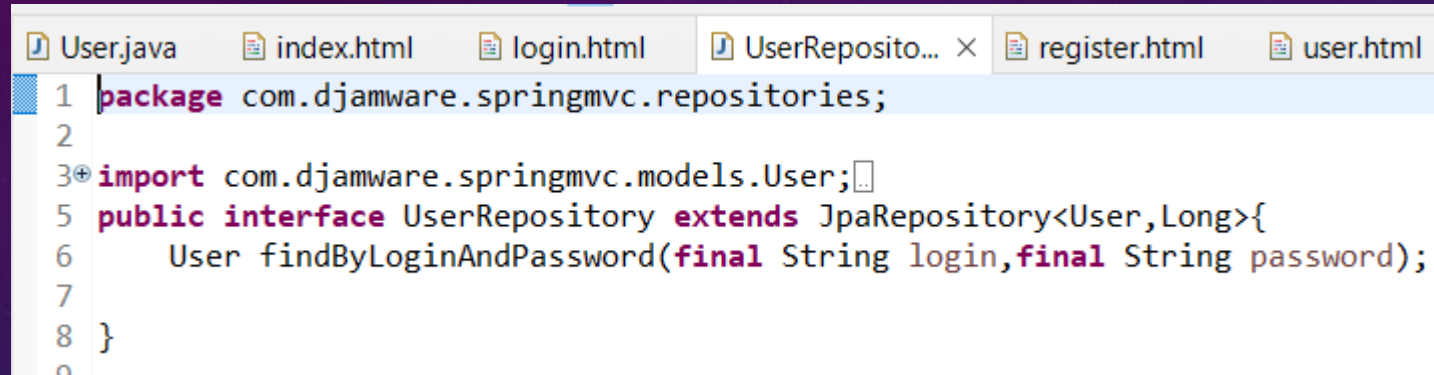


Création du Model: User.java

```
User.java x index.html login.html UserReposito... register.html
1 package com.djamware.springmvc.models;
2
3 import javax.persistence.Entity;
4 import javax.persistence.GeneratedValue;
5 import javax.persistence.GenerationType;
6 import javax.persistence.Id;
7 @Entity
8 public class User {
9     @Id
10    @GeneratedValue(strategy=GenerationType.AUTO)
11    private Long id;
12
13    private String login;
14
15    private String password;
16
17    private String fullName;
18
19    public User() {}
20
21    public User(String login, String password, String fullName) {
22        super();
23        this.login = login;
24        this.password = password;
25        this.fullName = fullName;
26    }
27
28    public Long getId() {
29        return id;
30    }
31
32    public void setId(Long id) {
33        this.id = id;
34    }
35
36    public String getLogin() {
37        return login;
38    }
39
40    public void setLogin(String login) {
41        this.login = login;
42    }
```

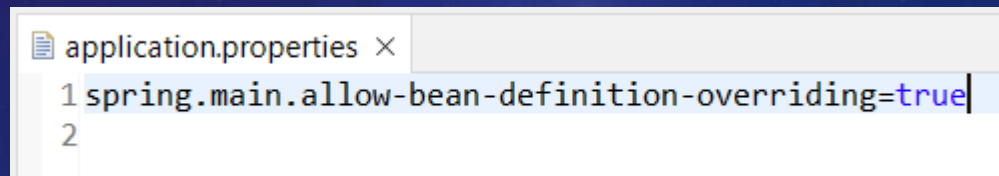
```
44 public String getPassword() {
45     return password;
46 }
47
48 public void setPassword(String password) {
49     this.password = password;
50 }
51
52 public String getFullName() {
53     return fullName;
54 }
55
56 public void setFullName(String fullName) {
57     this.fullName = fullName;
58 }
59
60 }
```

La couche DAO est remplacée par le repository



```
User.java  index.html  login.html  UserRepository... x  register.html  user.html
1 package com.djamware.springmvc.repositories;
2
3 import com.djamware.springmvc.models.User;
4
5 public interface UserRepository extends JpaRepository<User, Long>{
6     User findByLoginAndPassword(final String login, final String password);
7
8 }
9
```

On active la surcharge:



```
application.properties x
1 spring.main.allow-bean-definition-overriding=true
2
```


Création du contrôleur UserController.java

```
User.java  UserControll...  login.html  UserReposito...  register.html  user.html  create.html  edit.html  show.I
1 package com.djamware.springmvc.controllers;
2
3 import com.djamware.springmvc.models.User;
4 import com.djamware.springmvc.repositories.UserRepository;
5
6 import javax.servlet.http.HttpServletRequest;
7 import javax.servlet.http.HttpSession;
8
9 import org.springframework.beans.factory.annotation.Autowired;
10 import org.springframework.stereotype.Controller;
11 import org.springframework.ui.Model;
12 import org.springframework.web.bind.annotation.PathVariable;
13 import org.springframework.web.bind.annotation.RequestMapping;
14 import org.springframework.web.bind.annotation.RequestParam;
15
16 @Controller
17 public class UserController {
18
19     @Autowired
20     UserRepository userRepository;
21     @RequestMapping("/")
22     public String index(Model model) {
23         return "index";
24     }
25     @RequestMapping("/login")
26     public String login(Model model, HttpServletRequest request) {
27         HttpSession session=request.getSession(false);
28         if(session==null) {
29             return "login";
30         } else return "redirect:/user";
31     }
32     @RequestMapping("/register")
33     public String register(Model model, HttpServletRequest request) {
34         HttpSession session=request.getSession(false);
35         if(session==null) {
36             return "register";
37         } else return "redirect:/user";
38     }
39     @RequestMapping("/logging")
40     public String logging(@RequestParam String login,@RequestParam String password,HttpServletRequest request) {
41         User user=userRepository.findByLoginAndPassword(login,password);
42         if(user!= null) {
43             HttpSession session= request.getSession();
44             return "redirect:/user";
45         }
46     }
47 }
```

Avant de diriger vers la page login on s'assure qu'il n'y a pas une session courante (c'est-à-dire que l'utilisateur n'est pas déjà authentifié) sinon on le redirige vers /user De même pour la page /register

Pour logging (à laquelle on accède après envoi du form du log in) on vérifie que l'utilisateur existe → redirection vers /user sinon direction vers login

```

    }
    else {return "login";}
}
@RequestMapping("/registration")
public String registration(Model model,@RequestParam String login,@RequestParam String password,@RequestParam String fullName,HttpServletRequest request) {
    HttpSession session=request.getSession(false);
    if(session==null) {
        User user=new User();
        user.setFullName(fullName);
        user.setLogin(login);
        user.setPassword(password);
        userRepository.save(user);
        session=request.getSession(true);}

    return "redirect:/user";
}
@RequestMapping("/logout")
public String logout(Model model,HttpServletRequest request) {
    HttpSession session=request.getSession(false);
    if(session!=null) {
        session.invalidate();}
    return "index";
}
@RequestMapping("/spring")
public String spring(Model model) {
    return "spring";
}
@RequestMapping("/user")
public String user(Model model,HttpServletRequest request) {
    HttpSession session=request.getSession(false);
    if(session!=null) {
        model.addAttribute("users", userRepository.findAll());
        return "user";}
    else {return "login";}
}
@RequestMapping("/create")
public String create(Model model,HttpServletRequest request) {
    HttpSession session=request.getSession(false);
    if(session!=null) {
        return "create";}
    else {return "login";}
}
@RequestMapping("/save")

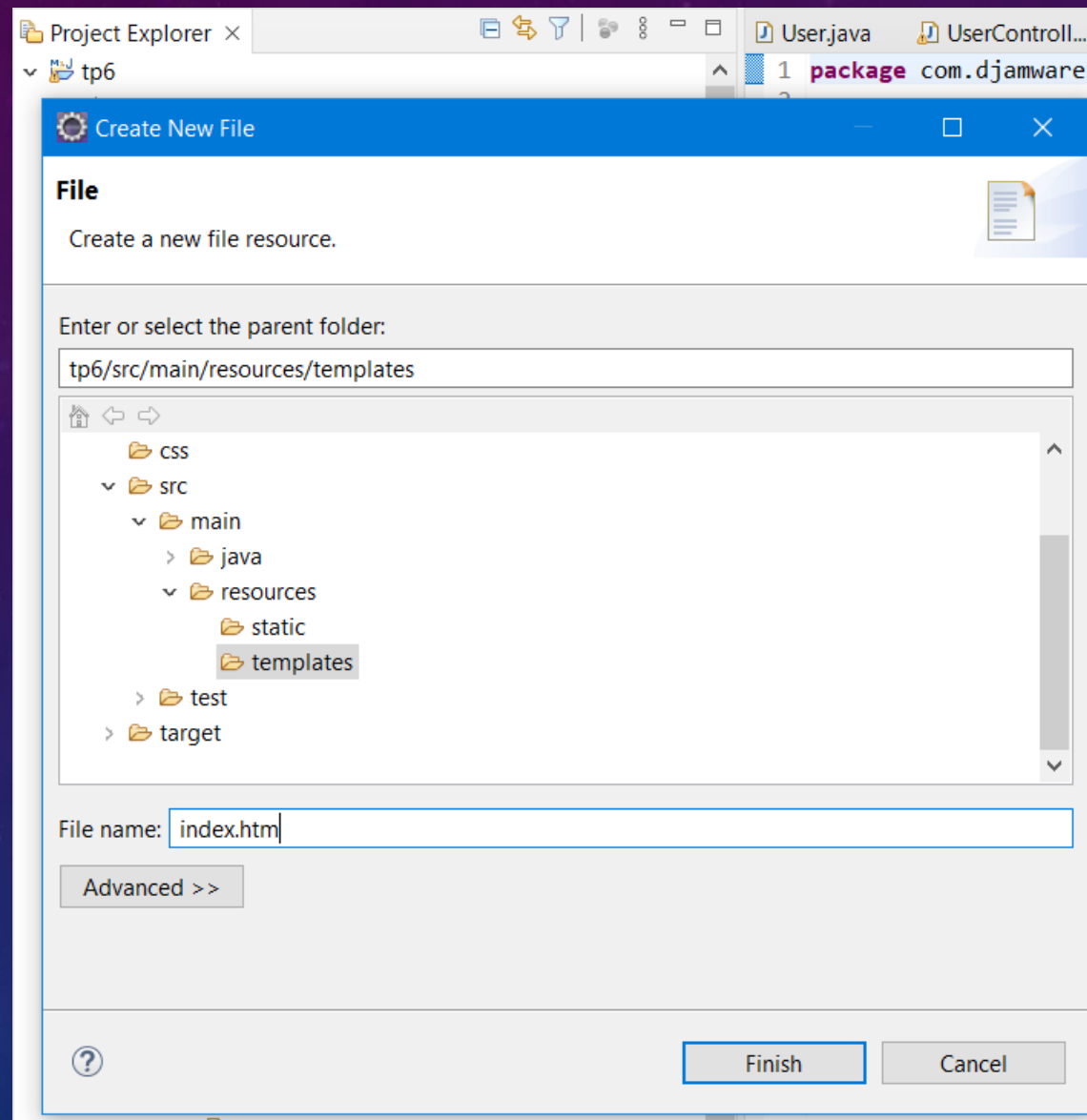
```

Pour registration on vérifie que l'utilisateur n'est pas connecté. Puis on enregistre le nouvel utilisateur + on crée une session pour qu'il soit connecté automatiquement.

Pour tout path, on vérifie que l'utilisateur est connecté (grâce à l'existence de la session).

```
ser.java | UserControll... x | login.html | UserReposito... | register.html | user.html | create.html | edit.html | show.html | spring.html »
3=@RequestMapping("/save")
9 public String save(@RequestParam String login,@RequestParam String password,@RequestParam String fullName,HttpServletRequest request) {
9     HttpSession session=request.getSession(false);
10     if(session!=null) {
11         User user=new User();
12         user.setFullName(fullName);
13         user.setLogin(login);
14         user.setPassword(password);
15         userRepository.save(user);
16
17         return "redirect:/show/"+user.getId();}
18     else {return "login";}
19 }
20
21 @RequestMapping("/show/{id}")
22 public String show(@PathVariable Long id, Model model) {
23     model.addAttribute("user",userRepository.findById(id).orElse(null));
24     return "show";
25 }
26
27 @RequestMapping("/delete")
28 public String delete(@RequestParam Long id,HttpServletRequest request) {
29     HttpSession session=request.getSession(false);
30     if(session!=null) {
31         User user=userRepository.findById(id).orElse(null);
32         userRepository.delete(user);
33
34         return"redirect:/user";}
35     else {return "login";}
36 }
37
38 @RequestMapping("/edit/{id}")
39 public String edit(@PathVariable Long id, Model model,HttpServletRequest request) {
40     HttpSession session=request.getSession(false);
41     if(session!=null) {
42         model.addAttribute("user", userRepository.findById(id).orElse(null));
43         return "edit";}
44     else {return "login";}
45 }
46
47 @RequestMapping("/update")
48 public String update(@RequestParam Long id,@RequestParam String fullName,@RequestParam String login,@RequestParam String password,HttpServletRequest request) {
49     HttpSession session=request.getSession(false);
50     if(session!=null) {
51         User user=userRepository.findById(id).orElse(null);
52         user.setFullName(fullName);
53         user.setLogin(login);
54         user.setPassword(password);
55         userRepository.save(user);
56         return "redirect:/show/"+user.getId();}
57     else {return "login";}
58 }
59 }
```

Création de la partie Vue



index.html

```
index.html x User.java UserControll... Tp6Applicati... UserReposito... user.html create.html edit.html show.html spring.html »3
1 <!DOCTYPE html>
2 <html lang="en"
3 xmlns="http://www.w3.org/1999/xhtml"
4 xmlns:th="http://www.thymeleaf.org"
5 xmlns:layout="http://www.ultraq.net.nz/thymeleaf/layout">
6 <head>
7     <meta charset="UTF-8">
8     <meta http-equiv="X-UA-Compatible" content="IE=edge">
9     <meta name="viewport" content="width=device-width, initial-scale=1.0">
10    <title>Home</title>
11    <link rel="stylesheet" href="/tp6/bootstrap.min (1).css">
12    <link rel="stylesheet" href="/tp6/css/style.css">
13    <script src="https://code.jquery.com/jquery-3.3.1.slim.min.js" integrity="sha384-q8i/X+965Dz00rT7abK41JStQIAqVgRVzpbzo5smXKp4YfRvH+8abttTE1Pi6jizo
14    <script src="https://cdnjs.cloudflare.com/ajax/libs/popper.js/1.14.7/umd/popper.min.js" integrity="sha384-U02eT0CpHqdSJQ6hJty5KVphtPhzWj9WO1cLHTM
15    <link rel="stylesheet" href="https://stackpath.bootstrapcdn.com/bootstrap/4.3.1/css/bootstrap.min.css" integrity="sha384-ggOyR0iXCbMQv3Xipma34MD+
16    <script src="https://stackpath.bootstrapcdn.com/bootstrap/4.3.1/js/bootstrap.min.js" integrity="sha384-JjSmVgyd0p3pXB1rRibZUAYoIIy60rQ6VrjIEaFf/n
17 </head>
18 <body>
19 <nav class="navbar navbar-inverse navbar-fixed-top">
20 <div class="container">
21 <div class="navbar-header">
22 <button type="button" class="navbar-toggle collapsed" data-toggle="collapse" data-target="#navbar" aria-expanded="false" aria-controls="navbar">
23 <span class="sr-only">Toggle navigation</span>
24 <span class="icon-bar"></span>
25 <span class="icon-bar"></span>
26 <span class="icon-bar"></span>
27 </button>
28 <a class="navbar-brand" href="/spring">Spring MVC</a>
29 </div>
30 <div id="navbar" class="collapse navbar-collapse">
31 <ul class="nav navbar-nav">
32 <li class="active"><a href="/user">Home</a></li>
33 <li class="active"><a href="/login">Login</a></li>
34 <li class="active"><a href="/register">Register</a></li>
35 </ul>
36 </div><!-- /.nav-collapse -->
37 </div>
38 </nav>
39 <div class="container">
40 <div class="starter-template" layout:fragment="content"></div>
41 </div><!-- /.container -->
42
43 </body>
44 </html>
```

spring.html

```
spring.html x register.html user.html create.html edit.html show.html
1 <!DOCTYPE html>
2 <html lang="en">
3 <head>
4   <meta charset="UTF-8">
5   <meta http-equiv="X-UA-Compatible" content="IE=edge">
6   <meta name="viewport" content="width=device-width, initial-scale=1.0">
7   <link rel="stylesheet" href="/tp6/bootstrap.min (1).css">
8   <link rel="stylesheet" href="/tp6/css/style.css">
9   <script src="https://code.jquery.com/jquery-3.3.1.slim.min.js" integrity="sha384-q8i/X+965Dz00rT7abK41JStQIAqVgRVzpbzo5smXKp4YfRvH+8abtTE1Pi6jizo"
10  <script src="https://cdnjs.cloudflare.com/ajax/libs/popper.js/1.14.7/umd/popper.min.js" integrity="sha384-U02eT0CpHqSJQ6hJty5KVphtPhzWj9W01cLHTMG
11  <link rel="stylesheet" href="https://stackpath.bootstrapcdn.com/bootstrap/4.3.1/css/bootstrap.min.css" integrity="sha384-ggOyR0iXCbMQv3Xipma34MD+d
12  <script src="https://stackpath.bootstrapcdn.com/bootstrap/4.3.1/js/bootstrap.min.js" integrity="sha384-JjSmVgyd0p3pXB1rRibZUAYoIIy60rQ6VrjIEaFf/nJ
13   <title>More on Spring MVC</title>
14 </head>
15 <body>
16 <div>
17 <iframe src="https://mohehn-my.sharepoint.com/personal/fatmalaribi_insat_u-carthage_tn/_layouts/15/Doc.aspx?sourcedoc={403e87d5-8213-4790-86a7-fe5a72
18 <div><a href="/" class="btn btn-lg btn-primary">Retour</a></div>
19 </div>
20 </body>
21 </html>
```

login.html

```
login.html x Tp6Application.java UserRepository.java register.html user.html create.html edit.html show.html spring.html
1 <!DOCTYPE html>
2 <html lang="en">
3 <head>
4   <meta charset="UTF-8">
5   <meta http-equiv="X-UA-Compatible" content="IE=edge">
6   <meta name="viewport" content="width=device-width, initial-scale=1.0">
7   <link rel="stylesheet" href="/tp6/bootstrap.min (1).css">
8   <link rel="stylesheet" href="/tp6/css/style.css">
9   <script src="https://code.jquery.com/jquery-3.3.1.slim.min.js" integrity="sha384-q8i/X+965Dz00rT7abK41JStQIAqVgRVzpbzo5smXKp4YfRvH+8abtTE1Pi6jizo" >
10  <script src="https://cdnjs.cloudflare.com/ajax/libs/popper.js/1.14.7/umd/popper.min.js" integrity="sha384-U02eT0CpHqdSJQ6hJty5KVphtPhzWj9WO1cLHTM" >
11  <link rel="stylesheet" href="https://stackpath.bootstrapcdn.com/bootstrap/4.3.1/css/bootstrap.min.css" integrity="sha384-ggOyR0iXCbMQv3Xipma34MD+
12  <script src="https://stackpath.bootstrapcdn.com/bootstrap/4.3.1/js/bootstrap.min.js" integrity="sha384-JjSmVgyd0p3pXB1rRibZUAYoIIy60RQ6VrjIEaFf/n" >
13
14   <title>Login</title>
15 </head>
16 <body>
17   <a href="#" class="btn btn-lg btn-primary"><span class="glyphicon glyphicon-list"></span> Login</a>
18   <form action="/logging">
19     <div class="form-group">
20       <label for="email">User Login</label>
21       <input type="text" class="form-control" name="login"></input>
22     </div>
23     <div class="form-group">
24       <label for="email">User password</label>
25       <input type="password" class="form-control" name="password" />
26     </div>
27     <button type="submit" class="btn btn-success">Log in</button>
28   </form>
29 </body>
30 </html>
```

register.html

```
register.html x user.html create.html edit.html show.html
1 <!DOCTYPE html>
2 <html lang="en">
3 <head>
4   <meta charset="UTF-8">
5   <meta http-equiv="X-UA-Compatible" content="IE=edge">
6   <meta name="viewport" content="width=device-width, initial-scale=1.0">
7   <link rel="stylesheet" href="/tp6/bootstrap.min (1).css">
8   <link rel="stylesheet" href="/tp6/css/style.css">
9   <script src="https://code.jquery.com/jquery-3.3.1.slim.min.js" integrity="sha384-q8i/X+965Dz00rT7abK41JStQIAqVgRVzpbzo5smXKp4YfRvH+8abtTE1Pi6jizo" >
10  <script src="https://cdn.jsdelivr.net/npm/popper.js@1.14.7/dist/umd/popper.min.js" integrity="sha384-U02eT0CpHqdSJQ6hJty5KVphtPhzWj9WO1cLHTM" >
11  <link rel="stylesheet" href="https://stackpath.bootstrapcdn.com/bootstrap/4.3.1/css/bootstrap.min.css" integrity="sha384-ggOyR0iXCbMQv3Xipma34MD+" >
12  <script src="https://stackpath.bootstrapcdn.com/bootstrap/4.3.1/js/bootstrap.min.js" integrity="sha384-JjSmVgyd0p3pXB1rRibZUAYoIIy60rQ6VrjIEaFf/n" >
13
14   <title>Register</title>
15 </head>
16 <body>
17   <a href="#" class="btn btn-lg btn-primary"><span class="glyphicon glyphicon-list"></span> Registration</a>
18   <form action="/registration">
19     <div class="form-group">
20       <label for="fullName">User FullName:</label>
21       <input type="text" class="form-control" name="fullName" />
22     </div>
23     <div class="form-group">
24       <label for="email">User Login</label>
25       <input type="text" class="form-control" name="login"></input>
26     </div>
27     <div class="form-group">
28       <label for="password">User password</label>
29       <input type="password" class="form-control" name="password" />
30     </div>
31     <button type="submit" class="btn btn-success">Register</button>
32   </form>
33 </body>
34 </html>
```


user.html

```
user.html x create.html edit.html show.html
2<html lang="en"
3  xmlns="http://www.w3.org/1999/xhtml"
4  xmlns:th="http://www.thymeleaf.org"
5  xmlns:layout="http://www.ultraq.net.nz/thymeleaf/layout"
6  layout:decorate="default">
7<head>
8  <link rel="stylesheet" href="/tp6/bootstrap.min (1).css">
9  <link rel="stylesheet" href="/tp6/css/style.css">
10 <script src="https://code.jquery.com/jquery-3.3.1.slim.min.js" integrity="sha384-q8i/X+965Dz00rT7abK41JStQIAqVgRVzpbzo5smXKp4YfRvH+8abtTE1Pi6jizo"
11 <script src="https://cdnjs.cloudflare.com/ajax/libs/popper.js/1.14.7/umd/popper.min.js" integrity="sha384-U02eT0CpHqdSJQ6hJty5KVphtPhzWj9WO1cLHTMG
12 <link rel="stylesheet" href="https://stackpath.bootstrapcdn.com/bootstrap/4.3.1/css/bootstrap.min.css" integrity="sha384-ggOyR0iXCbMQv3Xipma34MD+d
13 <script src="https://stackpath.bootstrapcdn.com/bootstrap/4.3.1/js/bootstrap.min.js" integrity="sha384-JjSmVgyd0p3pXB1rRibZUAYoIIy6OrQ6VrjIEaFf/nJ
14 <title>User List</title>
15 </head>
16<body>
17<div layout:fragment="content" class="row">
18<div class="col-xs-8 col-md-8">
19<h3>
20 <a href="/create" class="btn btn-primary"><span class="glyphicon glyphicon-plus-sign"></span> User</a>
21 </h3>
22 <h2>User List</h2>
23<div class="table-responsive">
24<table class="table" id="user-table">
25<thead>
26<tr>
27 <th>User FullName</th>
28 <th>User login</th>
29 <th>User password</th>
30 </tr>
31 </thead>
32<tbody>
33<tr th:each="user : ${users}">
34 <td><a th:text="${user.fullName}" th:href="@{'/show/' + ${user.id}}"></a></td>
35 <td th:text="${user.login}"></td>
36 <td th:text="${user.password}"></td>
37 </tr>
38 </tbody>
39 </table>
40 </div>
41 </div>
42 </div>
43 <a href="/logout" class="btn btn-warning">Logout</a>
44 </body>
45 </html>
```

create.html

create.html x edit.html show.html

```
1 <!DOCTYPE HTML>
2 <html lang="en"
3 xmlns="http://www.w3.org/1999/xhtml"
4 xmlns:th="http://www.thymeleaf.org"
5 xmlns:layout="http://www.ultraq.net.nz/thymeleaf/layout"
6 layout:decorate="default">
7 <head>
8   <link rel="stylesheet" href="/tp6/bootstrap.min (1).css">
9   <link rel="stylesheet" href="/tp6/css/style.css">
10  <script src="https://code.jquery.com/jquery-3.3.1.slim.min.js" integrity="sha384-q8i/X+965Dz00rT7abK41JStQIAqVgRVzpbzo5smXKp4YfRvH+8abtTE1Pi6jizo"
11  <script src="https://cdnjs.cloudflare.com/ajax/libs/popper.js/1.14.7/umd/popper.min.js" integrity="sha384-U02eT0CpHqdSJQ6hJty5KVphtPhzWj9WO1cLHTM0
12  <link rel="stylesheet" href="https://stackpath.bootstrapcdn.com/bootstrap/4.3.1/css/bootstrap.min.css" integrity="sha384-ggOyR0iXCbMQv3Xipma34MD+d
13  <script src="https://stackpath.bootstrapcdn.com/bootstrap/4.3.1/js/bootstrap.min.js" integrity="sha384-JjSmVgyd0p3pXB1rRibZUAYoIIy60rQ6VrjIEaFf/n
14  <title>Create User</title>
15 </head>
16 <body>
17 <div layout:fragment="content" class="row">
18 <div class="col-xs-8 col-md-8">
19 <h3>
20 <a href="/user" class="btn btn-lg btn-primary"><span class="glyphicon glyphicon-list"></span> User</a>
21 </h3>
22 <h2>Create User</h2>
23 <form action="/save">
24 <div class="form-group">
25 <label for="email">User FullName:</label>
26 <input type="text" class="form-control" name="fullName" />
27 </div>
28 <div class="form-group">
29 <label for="email">User Login</label>
30 <input type="text" class="form-control" name="login"></input>
31 </div>
32 <div class="form-group">
33 <label for="email">User password</label>
34 <input type="password" class="form-control" name="password" />
35 </div>
36
37 <button type="submit" class="btn btn-success">Save</button>
38 </form>
39 </div>
40 </div>
41 </body>
42 </html>
```

```
edit.html x show.html
1 <!DOCTYPE HTML>
2 <html lang="en"
3 xmlns="http://www.w3.org/1999/xhtml"
4 xmlns:th="http://www.thymeleaf.org"
5 xmlns:layout="http://www.ultraq.net.nz/thymeleaf/layout"
6 layout:decorate="default">
7 <head>
8 <link rel="stylesheet" href="/tp6/bootstrap.min (1).css">
9 <link rel="stylesheet" href="/tp6/css/style.css">
10 <script src="https://code.jquery.com/jquery-3.3.1.slim.min.js" integrity="sha384-q8i/X+965Dz00rT7abK41JStQIAqVgRVzpbzo5smXKp4YfRvH+8abtTE1Pi6jizo"
11 <script src="https://cdnjs.cloudflare.com/ajax/libs/popper.js/1.14.7/umd/popper.min.js" integrity="sha384-U02eT0CpHqdSJQ6hJty5KVphtPhzWj9W01cLHTMG
12 <link rel="stylesheet" href="https://stackpath.bootstrapcdn.com/bootstrap/4.3.1/css/bootstrap.min.css" integrity="sha384-ggOyR0iXCbMQv3Xipma34MD+d
13 <script src="https://stackpath.bootstrapcdn.com/bootstrap/4.3.1/js/bootstrap.min.js" integrity="sha384-JjSmVgyd0p3pXB1rRibZUAYoIIy60rQ6VrjIEaFf/nJ
14 <title>Edit User</title>
15 </head>
16 <body>
17 <div layout:fragment="content" class="row">
18 <div class="col-xs-8 col-md-8">
19 <h3>
20 <a href="/user" class="btn btn-lg btn-primary"><span class="glyphicon glyphicon-list"></span> User</a>
21 </h3>
22 <h2>Edit User</h2>
23 <form action="/update">
24 <div class="form-group">
25 <label for="email">User FullName:</label>
26 <input type="text" class="form-control" name="fullName" th:value="${user.fullName}" />
27 </div>
28 <div class="form-group">
29 <label for="email">User Login</label>
30 <input type="text" class="form-control" name="login" th:value="${user.login}"></input>
31 </div>
32 <div class="form-group">
33 <label for="email">User password</label>
34 <input type="password" class="form-control" name="password" th:value="${user.password}" />
35 </div>
36
37 <input type="hidden" name="id" th:value="${user.id}" />
38 <button type="submit" class="btn btn-success">Save</button>
39 </form>
40 </div>
41 </div>
42 </body>
43 </html>
```

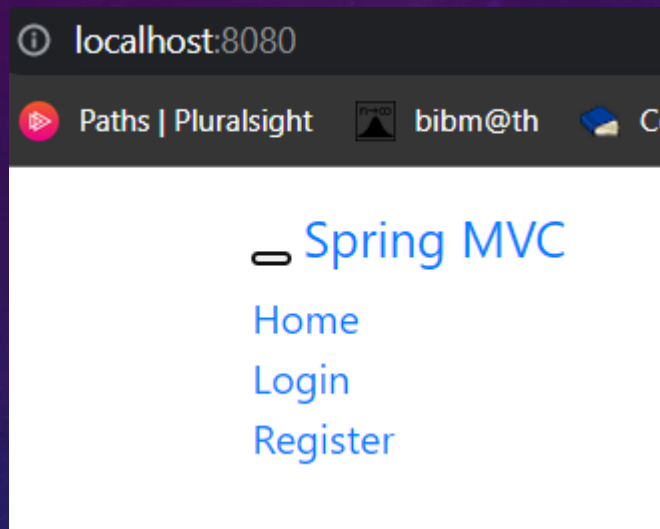
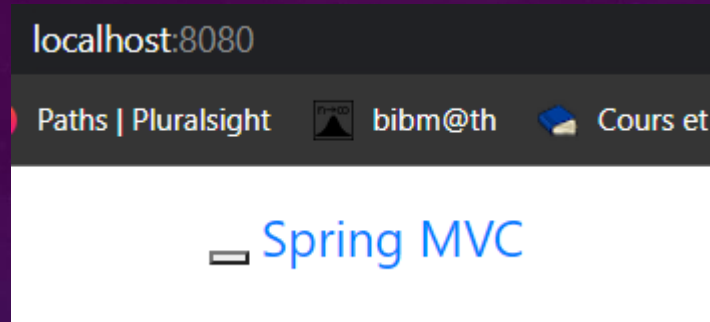
show.html

```
show.html x
1 <!DOCTYPE HTML>
2 <html lang="en"
3 xmlns="http://www.w3.org/1999/xhtml"
4 xmlns:th="http://www.thymeleaf.org"
5 xmlns:layout="http://www.ultraq.net.nz/thymeleaf/layout"
6 layout:decorate="default">
7 <head>
8 <link rel="stylesheet" href="/tp6/bootstrap.min (1).css">
9 <link rel="stylesheet" href="/tp6/css/style.css">
10 <script src="https://code.jquery.com/jquery-3.3.1.slim.min.js" integrity="sha384-q8i/X+965Dz00rT7abK41JStQIAqVgRVzpbzo5smXKp4YfRvH+8abtTE1Pi6jizo"
11 <script src="https://cdn.jsdelivr.net/npm/popper.js@1.14.7/umd/popper.min.js" integrity="sha384-U02eT0CpHqdSJQ6hJty5KVphtPhzWj9WO1cLHTMG
12 <link rel="stylesheet" href="https://stackpath.bootstrapcdn.com/bootstrap/4.3.1/css/bootstrap.min.css" integrity="sha384-ggOyR0iXCbMQv3Xipma34MD+d
13 <script src="https://stackpath.bootstrapcdn.com/bootstrap/4.3.1/js/bootstrap.min.js" integrity="sha384-JjSmVgyd0p3pXB1rRibZUAYoIIy60rQ6VrjIEaFf/nJ
14 <title>Show User</title>
15 </head>
16 <body>
17 <div layout:fragment="content" class="row">
18 <div class="col-xs-8 col-md-8">
19 <h3>
20 <a href="/user" class="btn btn-primary"><span class="glyphicon glyphicon-list"></span> User</a>
21 </h3>
22 <h2 th:text="${user.fullName}"></h2>
23 <dl class="list">
24 <dt>User login</dt>
25 <dd th:text="${user.login}"></dd>
26 <dt>User password</dt>
27 <dd th:text="${user.password}"></dd>
28 </dl>
29 <form action="/delete">
30 <input type="hidden" name="id" th:value="${user.id}" />
31 <h2><input type="submit" class="btn btn-danger" value="Delete" onclick="return confirm('Are you sure?');" />
32 <a th:href="@{/edit/} + ${user.id}" class="btn btn-warning">Edit</a></h2>
33 </form>
34 </div>
35 </div>
36 </body>
37 </html>
```


Exécution



1. Accueil



2. On clique sur Register

A screenshot of the registration page at 'localhost:8080/register'. The browser's address bar shows the URL. The page has a blue header with the word 'Registration' in white. Below the header, there are three input fields: 'User FullName:' with the value 'fatma laribi', 'User Login' with the value 'fatma', and 'User password' with masked characters '....'. A green 'Register' button is located at the bottom of the form.

3. On accède à la Liste des utilisateurs

← → ↻ localhost:8080/user

Applications Paths | Pluralsight bibm@th Cours et Exercices maths prepa Exo7Math - YouTube MEGA Learn | freeCodeCa... MEGA

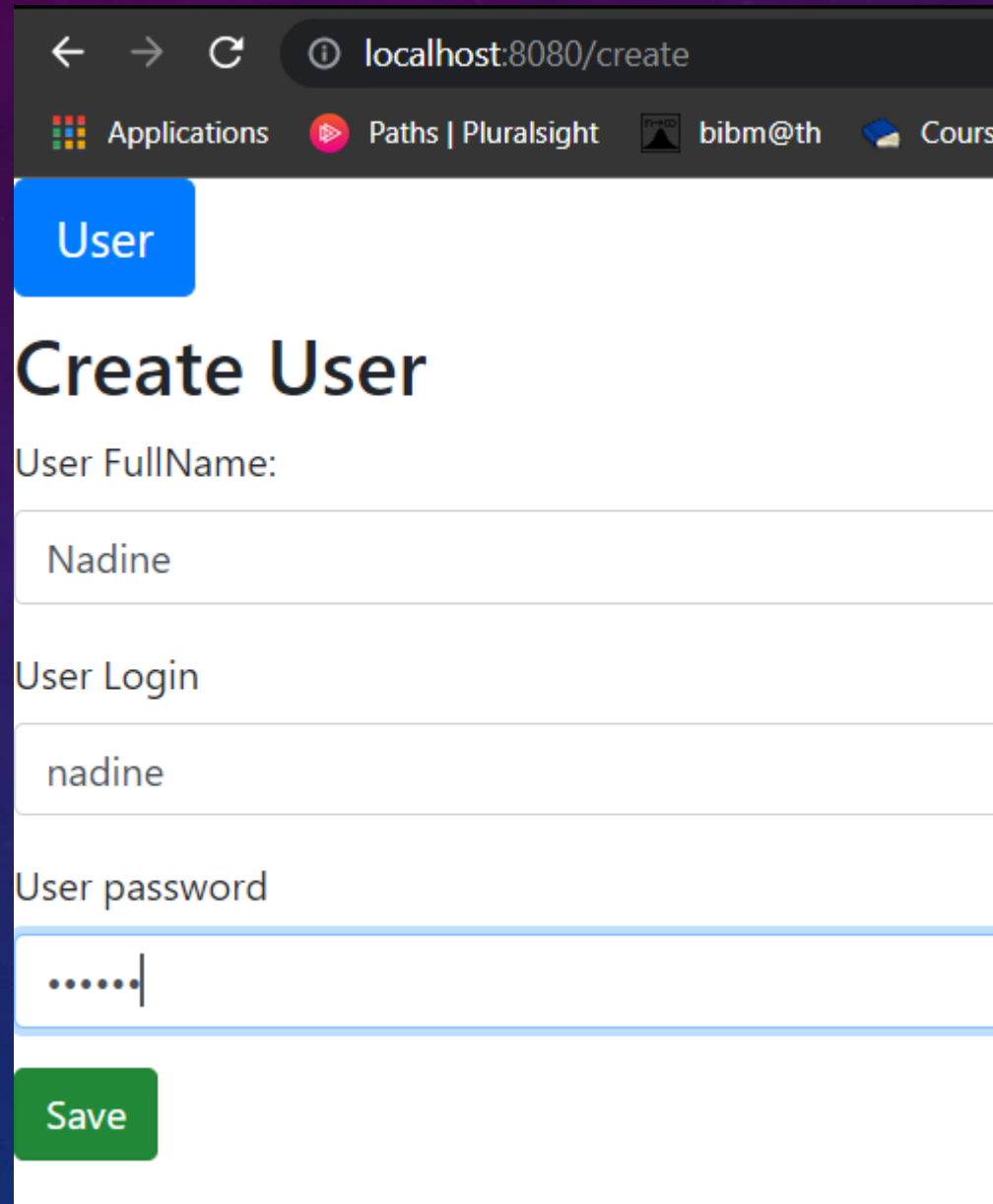
User

User List

| User FullName | User login | User password |
|---------------|------------|---------------|
| fatma laribi | fatma | fatma |

Logout

4. On clique sur User dans la page précédente pour accéder à create User



← → ↻ ⓘ localhost:8080/create

Applications Paths | Pluralsight bibm@th Cours

User

Create User

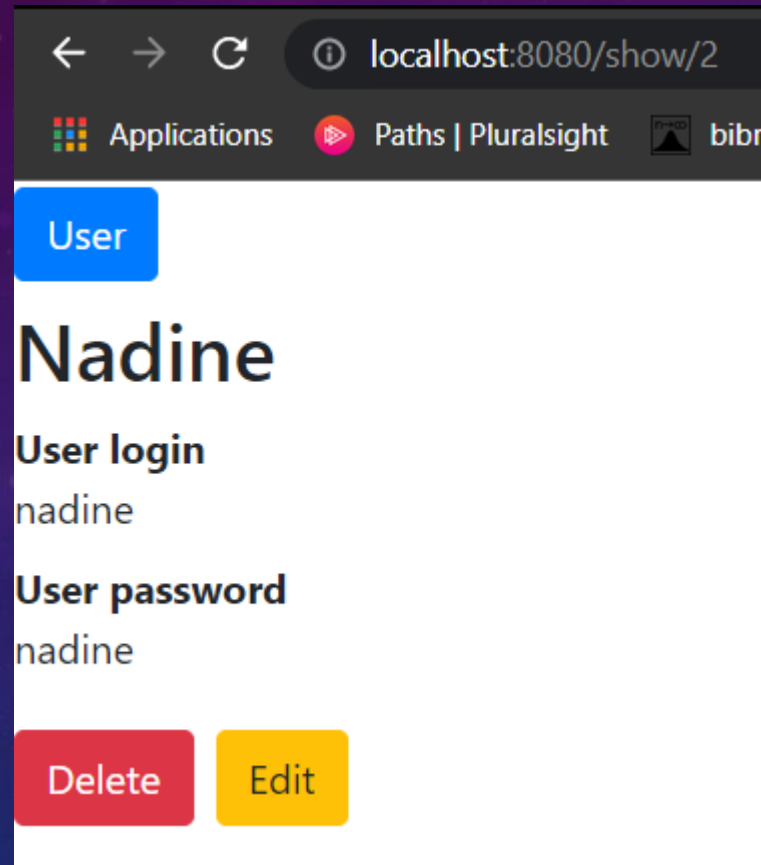
User FullName:

User Login

User password

Save

5. Le user Nadine est maintenant crée et on est redirigé vers show/2 qui peut nous permettre de le modifier ou supprimer



6. Modification du user Nadine

localhost:8080/edit/2

User

Edit User

User FullName:

Nadine

User Login

nadine2

User password

.....

Save

localhost:8080/show/2

User

Nadine

User login

nadine2

User password

nadine

Delete Edit

localhost:8080/user

User

User List

| User FullName | User login | User password |
|---------------|------------|---------------|
| fatma laribi | fatma | fatma |
| Nadine | nadine2 | nadine |

Logout

7. Suppression du user fatma

1)

User

User List

| User FullName | User login | User password |
|------------------------------|------------|---------------|
| fatma laribi | fatma | fatma |
| Nadine | nadine2 | nadine |

Logout

2)

localhost:8080/show/1

User

fatma laribi

User login
fatma

User password
fatma

Delete Edit

3)

localhost:8080/show/1

User

fatma laribi

User login
fatma

User password
fatma

Delete Edit

localhost:8080 indique
Are you sure?
OK Annuler

4)

localhost:8080/user

User

User List

| User FullName | User login | User password |
|------------------------|------------|---------------|
| Nadine | nadine2 | nadine |

Logout

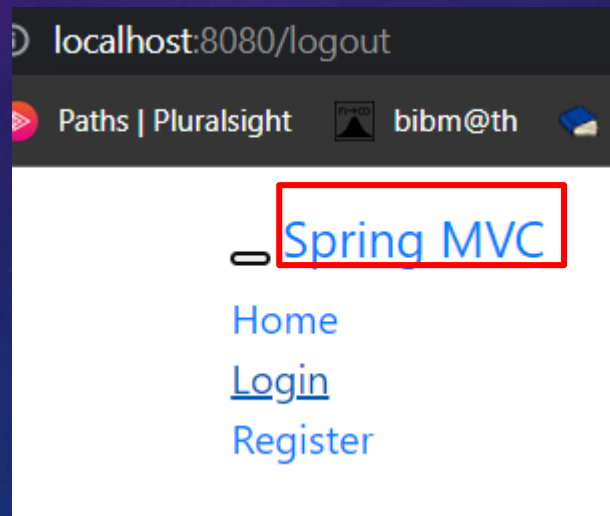
8. Essai du logout et visite de la page Spring MVC qui contient une présentation sur Spring (embedded)

User

User List

| User FullName | User login | User password |
|------------------------|------------|---------------|
| Nadine | nadine2 | nadine |

[Logout](#)



Un petit aperçu sur Spring Boot:

Spring Boot provides a good platform for Java developers to develop a stand-alone and production-grade spring application that you can just run. You can get started with minimum configurations without the need for an entire Spring configuration setup.

Advantages

- Easy to understand and develop spring applications
- Increases productivity
- Reduces the development time

Goals

- To avoid complex XML configuration in Spring
- To develop a production ready Spring applications in an easier way
- To reduce the development time and run the application independently
- Offer an easier way of getting started with the application



9. Essai de la page login

← → ↻ ⓘ localhost:8080/login

Applications Paths | Pluralsight bibm@

Login

User Login

nadine2

User password

.....

Log in

← → ↻ ⓘ localhost:8080/user

Applications Paths | Pluralsight bibm@th Cours et Exercices maths prepa Exo7Math - YouTube MEGA Learn | free

User

User List

| User FullName | User login | User password |
|---------------|------------|---------------|
| Nadine | nadine2 | nadine |

Logout

10. Si on essaye d'accéder à /user par exemple alors qu'on n'est pas connecté on est redirigé vers la page login

← → ↻ 🔍 localhost:8080/user

Applications Paths | Pluralsight bibm@th

Login

User Login

User password

Log in

The background is a gradient of dark blue and purple, speckled with small white dots. It features several abstract geometric elements: a large, thick black spiral on the left; a smaller black spiral on the right; a black four-pointed star in the upper left; a black circle with a dot in the center in the upper right; and a black circle with a dot in the center in the lower left. In the bottom right corner, there is a complex diagram consisting of concentric circles, a spiral, and a dashed line with an arrow pointing upwards. The text "Merci pour votre attention!" is centered in the middle of the image in a white, sans-serif font.

Merci pour votre attention!