

Rapport Controle J2EE

Créer un projet Java avec Spring Initializr de type Maven:

The screenshot shows the Spring Initializr web form. At the top, 'Spring Boot' is set to '3.0.6'. A checkbox 'Download pre-built shared indexes for JDK and Maven libraries' is checked. Under 'Dependencies:', a list of categories is shown on the left, including Developer Tools, Web, Template Engines, Security, SQL, NoSQL, Messaging, I/O, Ops, Observability, Testing, Spring Cloud, Spring Cloud Tools, Spring Cloud Config, and Spring Cloud Discovery. On the right, it says 'Nothing selected'. Below that, the 'Added dependencies:' section lists 'H2 Database', 'Spring Web', 'Lombok', and 'Spring Data JPA'. At the bottom, there are buttons for '?', 'Previous', 'Create' (highlighted in blue), and 'Cancel'.

Créer l'entité JPA Patient

```
@Entity
@Data @NoArgsConstructor @AllArgsConstructor
public class Patient {
    @Id @GeneratedValue(strategy = GenerationType.IDENTITY)
    private Long id;
    @NotEmpty
    @Size(min = 2, max = 40)
    private String nom;
    @Temporal(TemporalType.DATE)
    private Date dateNaissance;
    private boolean malade;
}
```

```
@DecimalMax("100")
private int score;
}
```

Configurer le fichier `application.properties`

- avec *mariadb*

```
spring.datasource.url=jdbc:h2:mem:patients_db
spring.h2.console.enabled=true
server.port=8081
```

- avec *h2-database*

```
spring.datasource.url=jdbc:mariadb://localhost:8082/patients_db
spring.datasource.username=root
spring.datasource.password=root
spring.jpa.hibernate.ddl-auto=create
spring.jpa.properties.hibernate.dialect=org.hibernate.dialect.MariaDBDialect
server.port=8081
```

Tester la méthode Chercher dans `PatientRepository`

```
public interface PatientRepository extends JpaRepository<Patient, Long> {
    Page<Patient> findByNomContainsIgnoreCase(String kw, Pageable pageable);
}
```

Créer le contrôleur de chaque chemin des patients `PatientController`

```
public String patients(Model model, ...) {
    Page<Patient> pagePatients = patientRepository
        .findByNomContainsIgnoreCase(keyword, PageRequest.of(page, size));
    ...
    return "patients";
}
```

Ajouter la vue `patients.html` pour afficher la liste des patients

- Bar de recherche

```
<form method="get" th:action="@{/user/index}">
    <input type="text" name="keyword" th:value="${keyword}">
    <button type="submit"> Chercher </button>
</form>
```

- Liste des patients

```
<table class="table mt-3">
    ...
```

```

<tr th:each="p:${listPatients}">
  <td th:text="${p.id}"></td>
  <td th:text="${p.nom}"></td>
  <td th:text="${p.datenaissance}"></td>
  ...
  <td th:if="${#authorization.expression('hasRole(''ADMIN'')')}">
    Visible pour l'admin seul
    <a th:href="@{/admin/editPatient(id=${p.id})}"> Modifier </a>
    <a th:href="@{/admin/delete(id=${p.id})}"> Supprimer </a>
  </td>
</tr>
...
</table>

```

- Liste des pages

```

<div role="group">
  <button th:each="page,status:${pages}">
    <a th:text="${status.index}"
      th:href="@{/user/index(page=${status.index}, keyword=${keyword})}"></a>
  </button>
</div>

```

Navbar Home Patient About ADMIN

Liste des patients

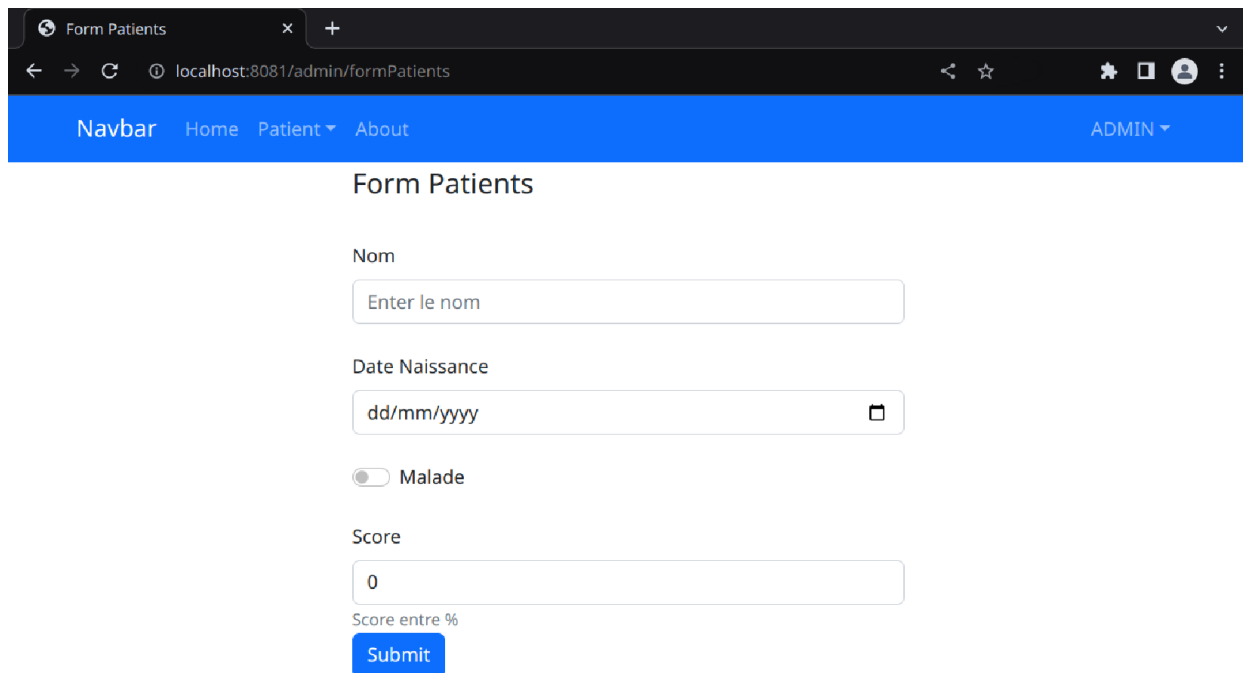
Chercher

Id	Nom	Date	Malade	Score	Action
1	Abdelaziz	2023-04-27	<input type="radio"/>	54%	Modifier Supprimer
2	Adil	2023-04-27	<input checked="" type="radio"/>	21%	Modifier Supprimer
3	Hind	2023-04-27	<input type="radio"/>	84%	Modifier Supprimer
4	Jamila	2023-04-27	<input type="radio"/>	93%	Modifier Supprimer
5	Khalid	2023-04-27	<input type="radio"/>	64%	Modifier Supprimer
6	Larbi	2023-04-27	<input checked="" type="radio"/>	51%	Modifier Supprimer
7	Latifa	2023-04-27	<input type="radio"/>	84%	Modifier Supprimer
8	Majida	2023-04-27	<input type="radio"/>	63%	Modifier Supprimer
9	Mobarek	2023-04-27	<input type="radio"/>	98%	Modifier Supprimer
10	Naima	2023-04-27	<input type="radio"/>	49%	Modifier Supprimer

Vue `formPatients.html` & Contrôleur pour remplir le formulaire des patients

```
<form method="post" th:action="@{/admin/save}">
  Nom <input type="text" th:value="${pat.nom}" name="nom">
  Date Naissa.. <input type="date" th:value="${pat.datenaissance}"
name="datenaissance">
  ...
  <button type="submit">Submit</button>
</form>
```

```
@PostMapping(path = "/admin/save")
@PreAuthorize("hasRole('ROLE_ADMIN')")
public String save(Model model, @Valid Patient patient, BindingResult bindingResult)
{
    if (bindingResult.hasErrors()) return "formPatients";
    patientRepository.save(patient);
    return "redirect:/user/index";
}
```



The screenshot shows a web browser window with the address bar displaying 'localhost:8081/admin/formPatients'. The page has a blue header with a 'Navbar' and links to 'Home', 'Patient', and 'About'. The main content area is titled 'Form Patients' and contains several input fields: 'Nom' (text input), 'Date Naissance' (date input), 'Malade' (checkbox), and 'Score' (text input). A 'Submit' button is at the bottom.

Page d'authentification

Authentication

Nom d'utilisateur

Mot de passe

☐ Se souvenir de moi

Se connecter