

## DEPARTEMENT MATHÉMATIQUES ET INFORMATIQUE

### Filière : II-BDCC

*Compte Rendu : Technologie web JEE :  
http ,Servlet, JSP, Spring MVC, Thymleaf.*

Réalisé par : Abdenasser Aroukhsiss

Encadré par : M. Youssfi Mohamed

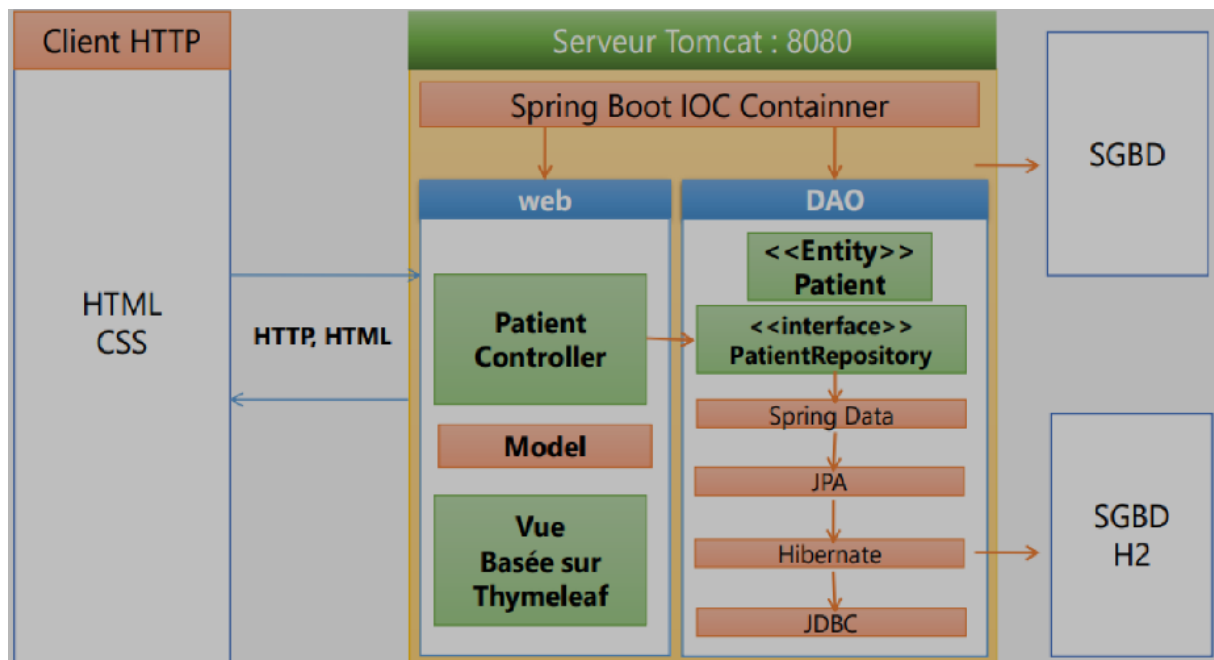
**Année Universitaire : 2020-2021**

## L'énoncé de l'activité :

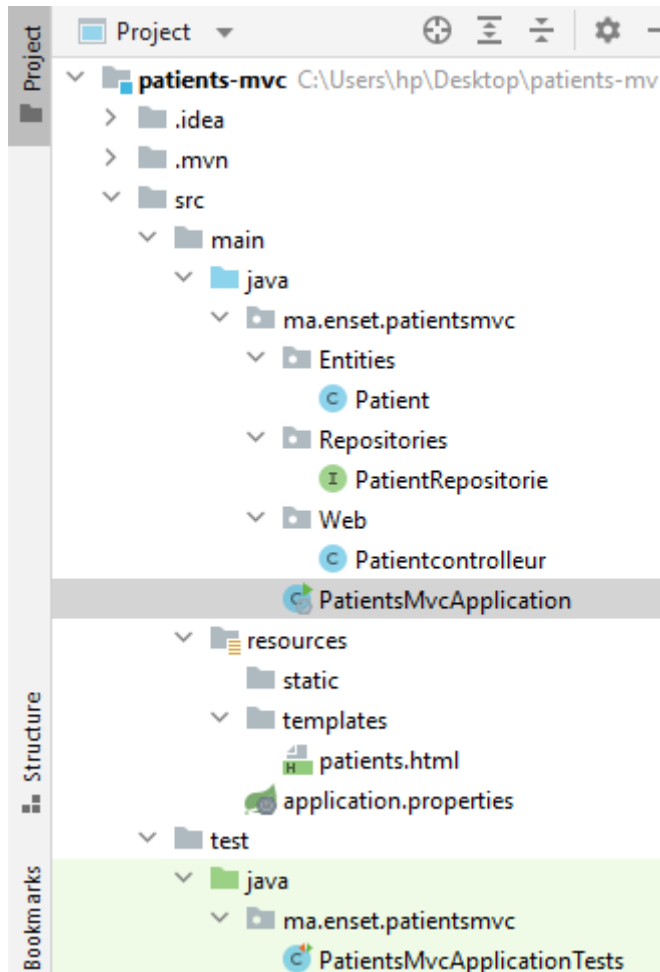
Créer une application Web JEE basée sur Spring MVC, Thymeleaf et Spring Data JPA qui permet de gérer les patients. L'application doit permettre les fonctionnalités suivantes :

- Afficher les patients
- Faire la pagination
- Chercher les patients
- Supprimer un patient
- Faire des améliorations supplémentaires

## Architecture du Projet :



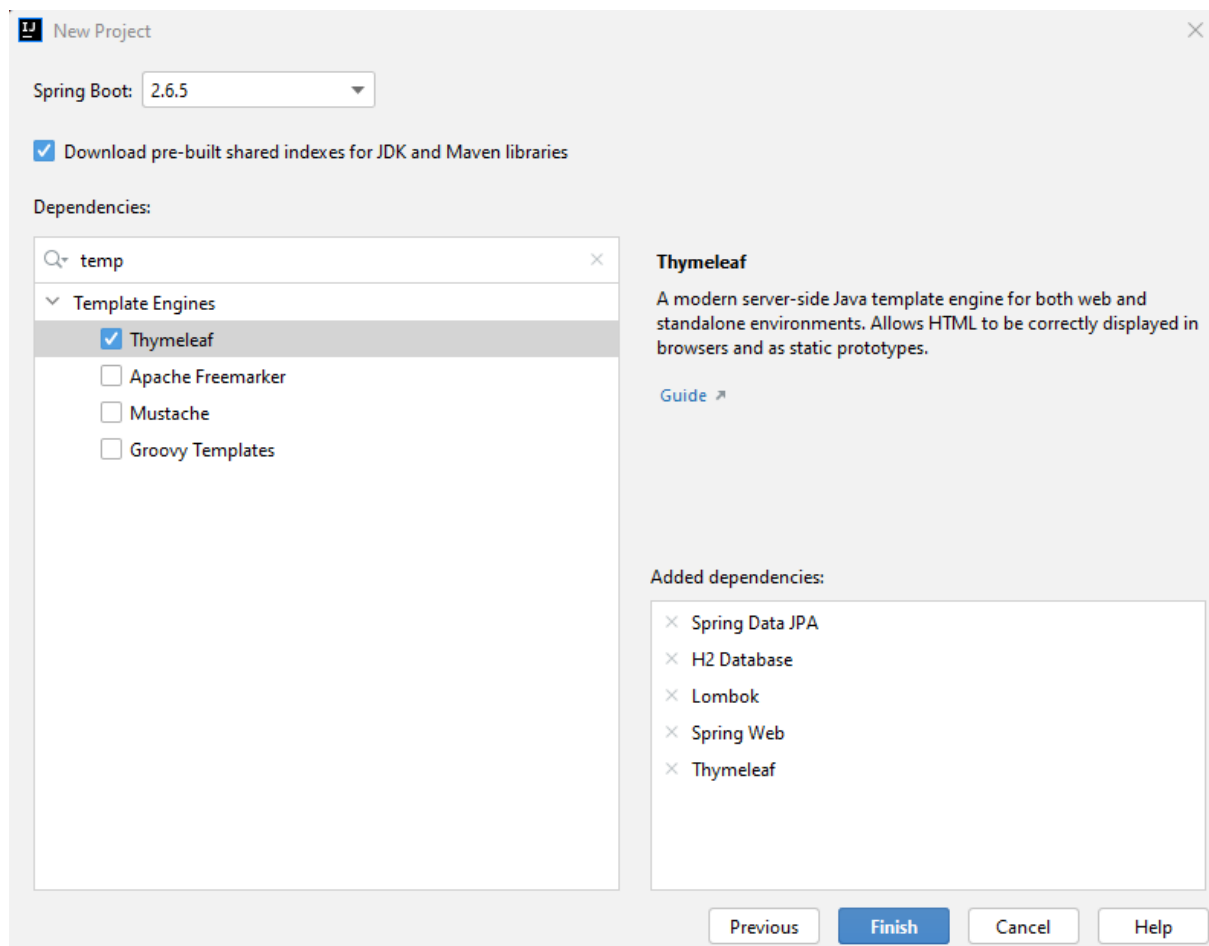
## Structure du Projet :



## Application.properties:

```
1 #spring.datasource.url=jdbc:h2:mem:hospital_db
2 #spring.h2.console.enabled=true
3 server.port=8080
4 spring.datasource.url=jdbc:mysql://localhost:3306/Patients_DB?createDatabaseIfNotExist=true
5 spring.datasource.username=root
6 spring.datasource.password=
7 spring.jpa.hibernate.ddl-auto=update
8 spring.jpa.properties.hibernate.dialect=org.hibernate.dialect.MariaDBDialect
9 spring.jpa.show-sql=true
10
11
```

## Dépendances :



New Project

Spring Boot: 2.6.5

☒ Download pre-built shared indexes for JDK and Maven libraries

Dependencies:

temp

Template Engines

- ☒ Thymeleaf
- ☐ Apache Freemarker
- ☐ Mustache
- ☐ Groovy Templates

**Thymeleaf**

A modern server-side Java template engine for both web and standalone environments. Allows HTML to be correctly displayed in browsers and as static prototypes.

[Guide](#)

Added dependencies:

- × Spring Data JPA
- × H2 Database
- × Lombok
- × Spring Web
- × Thymeleaf

Previous Finish Cancel Help

## Les entities :

```
package ma.enset.patientsmvc.Entities;

import lombok.AllArgsConstructor;
import lombok.Data;
import lombok.NoArgsConstructor;

import javax.persistence.*;
import java.util.Date;

@Entity @Data @AllArgsConstructor @NoArgsConstructor
public class Patient {
    @Id
    @GeneratedValue(strategy = GenerationType.IDENTITY)
    private Long id;
    private String nom;
    @Temporal(TemporalType.DATE)
    private Date date_de_naissance;
    private boolean malade;
    private int score;
}
```

## Repositories :

```
package ma.enset.patientsmvc.Repositories;

import ma.enset.patientsmvc.Entities.Patient;
import org.springframework.data.jpa.repository.JpaRepository;

public interface PatientRepository extends JpaRepository<Patient, Long> {
}
```

## Controller :

```
package ma.enset.patientsmvc.Web;

import lombok.AllArgsConstructor;
import ma.enset.patientsmvc.Repositories.PatientRepository;
import org.springframework.beans.factory.annotation.Autowired;
import org.springframework.stereotype.Controller;
import org.springframework.web.bind.annotation.GetMapping;

@Controller @AllArgsConstructor
public class PatientControleur {
    //on déclare une patient repository pour que notre controll
    PatientRepository patientRepository;

    @GetMapping(path="/index")
    public String patients(){
        return "patients";//cette methode va nous retourner une
    }
}
```

## Application :

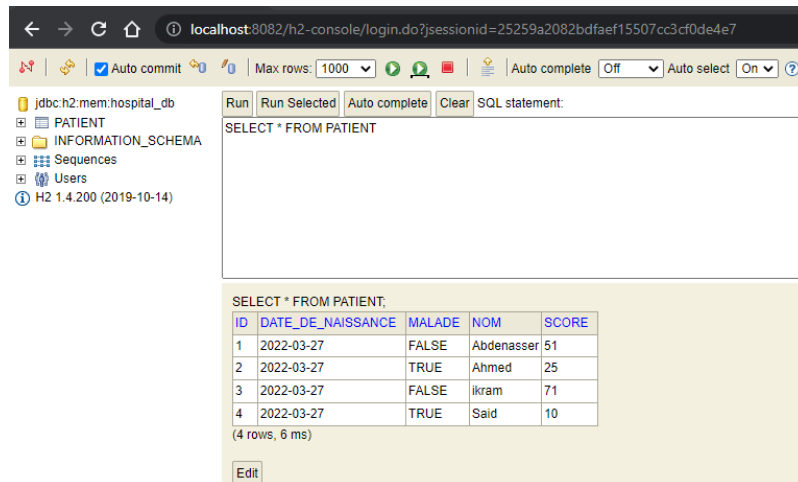
```
package ma.enset.patientsmvc;

import ...

@SpringBootApplication
public class PatientsMvcApplication {

    public static void main(String[] args) { SpringApplication.run(PatientsMvcApplication.class, args); }
    //@bean : pour que cette methode s'exécute au démarrage.
    @Bean
    CommandLineRunner commandLineRunner(PatientRepository patientRepository){
        return args->{
            patientRepository.save(new Patient( id: null, nom: "Abdenasser", new Date(), malade: false, score: 51));
            patientRepository.save(new Patient( id: null, nom: "Ahmed", new Date(), malade: true, score: 25));
            patientRepository.save(new Patient( id: null, nom: "ikram", new Date(), malade: false, score: 71));
            patientRepository.save(new Patient( id: null, nom: "Said", new Date(), malade: true, score: 10));
            //retourner la listes des patient aux demarage de l'appliocation
            patientRepository.findAll().forEach(p->{
                System.out.println(p.getNom());
            });
        };
    }
}
```

## H2-database :



localhost:8082/h2-console/login.do?sessionId=25259a2082bdfaf15507cc3cf0de4e7

Max rows: 1000 | Auto complete: Off | Auto select: On

Run | Run Selected | Auto complete | Clear | SQL statement:

SELECT \* FROM PATIENT

SELECT \* FROM PATIENT;

ID	DATE_DE_NAISSANCE	MALADE	NOM	SCORE
1	2022-03-27	FALSE	Abdenasser	51
2	2022-03-27	TRUE	Ahmed	25
3	2022-03-27	FALSE	ikram	71
4	2022-03-27	TRUE	Said	10

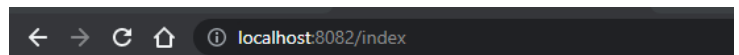
(4 rows, 6 ms)

Edit

## Template :

```
<meta charset="utf-8">
<title>Title</title>
</head>
<body>
<h3>Listes des patients</h3>
<table>
<thead>
<tr>
<th>id</th><th>Nom</th><th>Malade</th><th>Score</th>
</tr>
</thead>
<tbody>
<tr th:each="p:${listepatients}">
<td th:text="${p.id}"></td>
<td th:text="${p.nom}"></td>
<td th:text="${p.malade}"></td>
<td th:text="${p.score}"></td>
</tr>
</tbody>
</table>
</body>
</html>
```

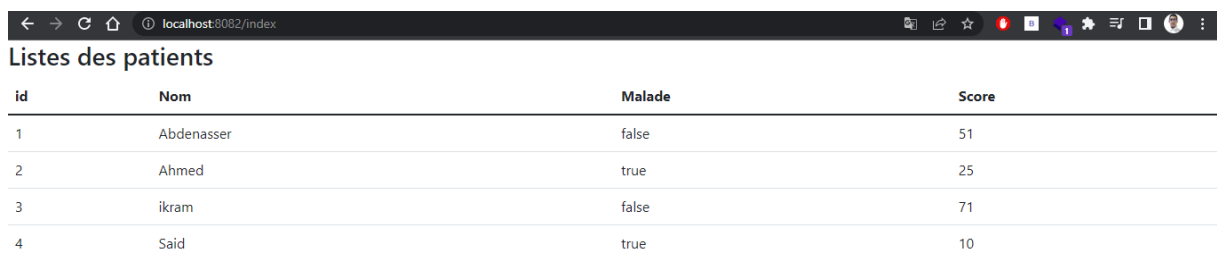
## Result :



### Listes des patients

id	Nom	Malade	Score
1	Abdenasser	false	51
2	Ahmed	true	25
3	ikram	false	71
4	Said	true	10

## Avec bootstrap 5 :



localhost:8082/index

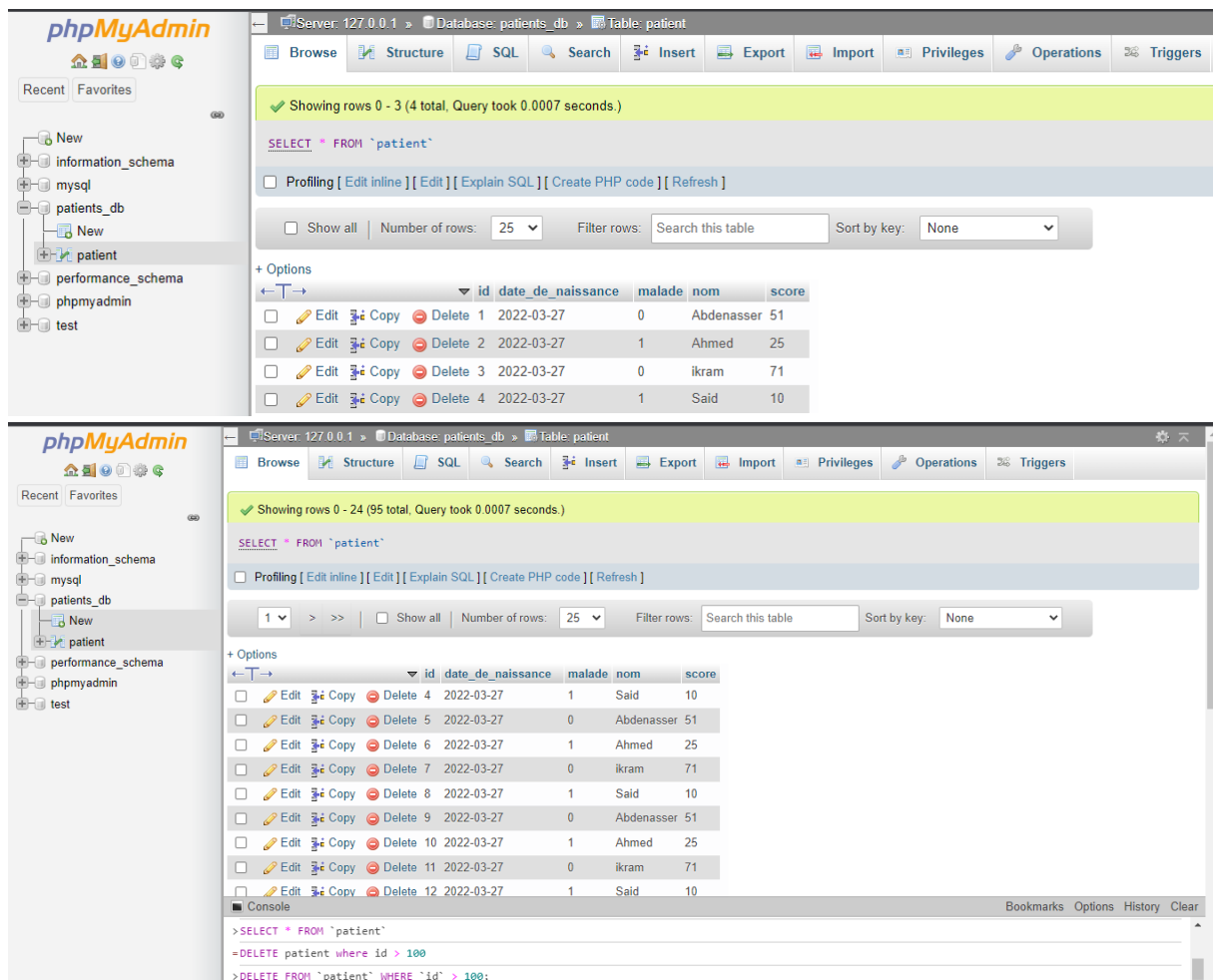
### Listes des patients

id	Nom	Malade	Score
1	Abdenasser	false	51
2	Ahmed	true	25
3	ikram	false	71
4	Said	true	10

On continue notre travail avec la base de données Mysql

```
#spring.datasource.url=jdbc:h2:mem:hospital_db
#spring.h2.console.enabled=true
server.port=8082

spring.datasource.url=jdbc:mysql://localhost:3306/Patients_DB?createDatabaseIfNotExist=true
spring.datasource.username=root
spring.datasource.password=
spring.jpa.hibernate.ddl-auto=update
spring.jpa.properties.hibernate.dialect=org.hibernate.dialect.MariaDBDialect
spring.jpa.show-sql=true
```



The screenshot displays the phpMyAdmin interface for a MySQL database named 'patients\_db'. The 'patient' table is selected, showing columns: id, date\_de\_naissance, malade, nom, and score. The first screenshot shows the first four rows of the table. The second screenshot shows the table after a DELETE operation, with rows 0-24 displayed. The console at the bottom shows the SQL commands used:

```
>SELECT * FROM `patient`
=DELETE patient where id > 100
>DELETE FROM `patient` WHERE `id` > 100;
```



## Les interface de l'application :

- Home : cette page contient tous les patients existents avec leurs informations :

Active Link Link Disabled Patients ▾ Username ▾

### Listes des patients

Mot clé

id	Nom	Malade	Date de naissance	Score		
6	Ahmed	true	2022-03-27	25	<input type="button" value="Delete"/>	<input type="button" value="Edite"/>
7	ikram	false	2022-03-27	71	<input type="button" value="Delete"/>	<input type="button" value="Edite"/>
8	Said	true	2022-03-27	10	<input type="button" value="Delete"/>	<input type="button" value="Edite"/>
9	Abdenasser	false	2022-03-27	51	<input type="button" value="Delete"/>	<input type="button" value="Edite"/>
10	Ahmed	true	2022-03-27	25	<input type="button" value="Delete"/>	<input type="button" value="Edite"/>

0 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24 25 26 27 28 29

- illustration de la confirmation de suppression d'un Patient :

Active Link Link Disabled Patients ▾ localhost:8080 indique vous etes sure ?? OK Annuler Username ▾

### Listes des patients

Mot clé

id	Nom	Malade	Date de naissance	Score		
6	Ahmed	true	2022-03-27	25	<input type="button" value="Delete"/>	<input type="button" value="Edite"/>
7	ikram	false	2022-03-27	71	<input type="button" value="Delete"/>	<input type="button" value="Edite"/>
8	Said	true	2022-03-27	10	<input type="button" value="Delete"/>	<input type="button" value="Edite"/>
9	Abdenasser	false	2022-03-27	51	<input type="button" value="Delete"/>	<input type="button" value="Edite"/>
10	Ahmed	true	2022-03-27	25	<input type="button" value="Delete"/>	<input type="button" value="Edite"/>

0 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24 25 26 27 28 29

- illustration de la recherche d'un patient avec un mot clé :

## Listes des patients

Mot clé

id	Nom	Malade	Date de naissance	Score		
9	Abdenasser	false	2022-03-27	51	<input type="button" value="Delete"/>	<input type="button" value="Edite"/>
13	Abdenasser	false	2022-03-27	51	<input type="button" value="Delete"/>	<input type="button" value="Edite"/>
17	Abdenasser	false	2022-03-27	51	<input type="button" value="Delete"/>	<input type="button" value="Edite"/>
25	Abdenasser	false	2022-03-27	51	<input type="button" value="Delete"/>	<input type="button" value="Edite"/>
29	Abdenasser	false	2022-03-27	51	<input type="button" value="Delete"/>	<input type="button" value="Edite"/>