

Gestion patient

Réalisé par :

Rachidy bader-eddine

Encadre par :

Mr. YOUSSEFI Mohamed

Introduction:

Dans le cadre du développement d'une application web JEE pour la gestion des patients, nous avons utilisé plusieurs technologies telles que Spring MVC, Spring Data, JPA et Hibernate.

Cette application a été conçue dans le cadre des activités pratiques n°2, n°3 et n°4 et elle est composée d'une couche DAO avec une entité JPA Patient, une interface PatientRepository basée sur Spring Data, ainsi qu'un test de la couche DAO.

La couche Web contient un contrôleur Spring MVC pour les patients et des vues basées sur Thymeleaf.

En outre, la partie sécurité est également incluse dans l'application avec Spring Security, conformément à l'activité pratique n°4.

Ce rapport présentera les différentes étapes de développement de l'application, ainsi que les choix technologiques et les considérations de conception qui ont été pris en compte. Nous décrirons également les fonctionnalités de l'application, notamment la recherche, l'édition, la suppression et l'ajout de patients, ainsi que les résultats obtenus lors des tests de performance de l'application.

Description de projet :

L'application de gestion des patients que nous avons développée est une application web basée sur Spring MVC, Spring Data, JPA et Hibernate. Elle permet de gérer les informations des patients, telles que le nom, le prénom, la date de naissance, le numéro de sécurité sociale, l'adresse, le numéro de téléphone, etc. Elle offre également une fonction de recherche pour faciliter la navigation dans la liste des patients.

L'application est divisée en deux couches : la couche DAO (Data Access Object) et la couche Web. La couche DAO est responsable de l'accès aux données de la base de données. Elle est composée d'une entité JPA Patient, d'une interface Patient Repository basée sur Spring Data et d'un test de la couche DAO. La couche Web est responsable de l'interface utilisateur. Elle comprend un contrôleur Spring MVC appelé Patient Controller, qui gère les requêtes HTTP et les réponses, ainsi que les vues basées sur Thymeleaf, telles que template.html, patients.html, formPatient.html et editPatient.html.

En outre, la partie sécurité de l'application est gérée par Spring Security, qui permet de protéger les données des patients contre toute utilisation abusive.

Dans l'ensemble, l'application de gestion des patients que nous avons développée est un projet complet qui utilise des technologies modernes pour offrir des fonctionnalités avancées et une sécurité accrue.

Problématique:

La problématique qui se pose avec cette application de gestion des patients est de savoir comment gérer efficacement les informations des patients dans un environnement web sécurisé. En effet, la gestion des informations de santé est une question délicate qui nécessite une approche rigoureuse pour garantir la confidentialité et la sécurité des données des patients.

Par conséquent, comment pouvons-nous concevoir une application web qui offre une interface utilisateur conviviale tout en protégeant les informations sensibles des patients contre tout accès non autorisé ? Comment pouvons-nous utiliser les technologies modernes pour faciliter la gestion des informations de

santé tout en maintenant un haut niveau de sécurité et de fiabilité ? Ce sont les questions auxquelles nous avons essayé de répondre dans le développement de cette application de gestion des patients.

Objectifs de projet:

1. Développer une application web de gestion des patients basée sur les technologies modernes de développement web et de sécurité informatique.
2. Offrir une interface utilisateur conviviale pour permettre aux professionnels de la santé de gérer efficacement les informations des patients.
3. Implémenter une couche DAO basée sur JPA et Hibernate pour permettre la persistance des données dans une base de données relationnelle.
4. Utiliser Spring Data pour faciliter l'accès aux données et permettre des opérations de recherche et de tri efficaces.
5. Concevoir une couche de sécurité robuste basée sur Spring Security pour protéger les informations des patients contre tout accès non autorisé.
6. Mettre en œuvre une pagination pour faciliter la navigation dans les pages affichant les informations des patients.
7. Valider les données saisies par les utilisateurs pour garantir la qualité des données stockées dans la base de données.
8. Écrire des tests pour assurer la qualité et la fiabilité du code source de l'application.

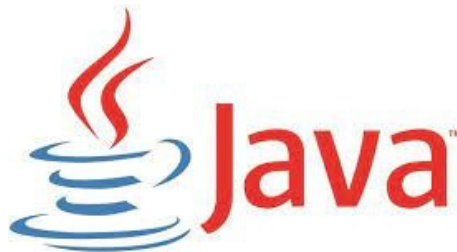
Fonctionnalités:

L'application de gestion des patients que nous avons développée offre plusieurs fonctionnalités permettant de visualiser, d'ajouter, de modifier et de supprimer des patients. Voici une liste des principales fonctionnalités :

1. Recherche de patients : l'application permet de rechercher des patients en utilisant différents critères, tels que le nom, le prénom ou le numéro de sécurité sociale.
2. Ajout de patients : l'application permet également d'ajouter de nouveaux patients en saisissant les informations requises, telles que le nom, le prénom, la date de naissance, etc.
3. Modification de patients : il est possible de modifier les informations d'un patient existant, telles que son adresse, son numéro de téléphone, etc.
4. Suppression de patients : l'application permet de supprimer un patient de la base de données.
5. Pagination des patients : la liste des patients peut être paginée pour faciliter la navigation et la recherche.
6. Validation des données : l'application vérifie la validité des données saisies avant de les enregistrer dans la base de données.
7. Sécurité : la partie sécurité de l'application est gérée par Spring Security, permettant de protéger les données des patients contre toute utilisation abusive.

Technologies et les outils utilise:

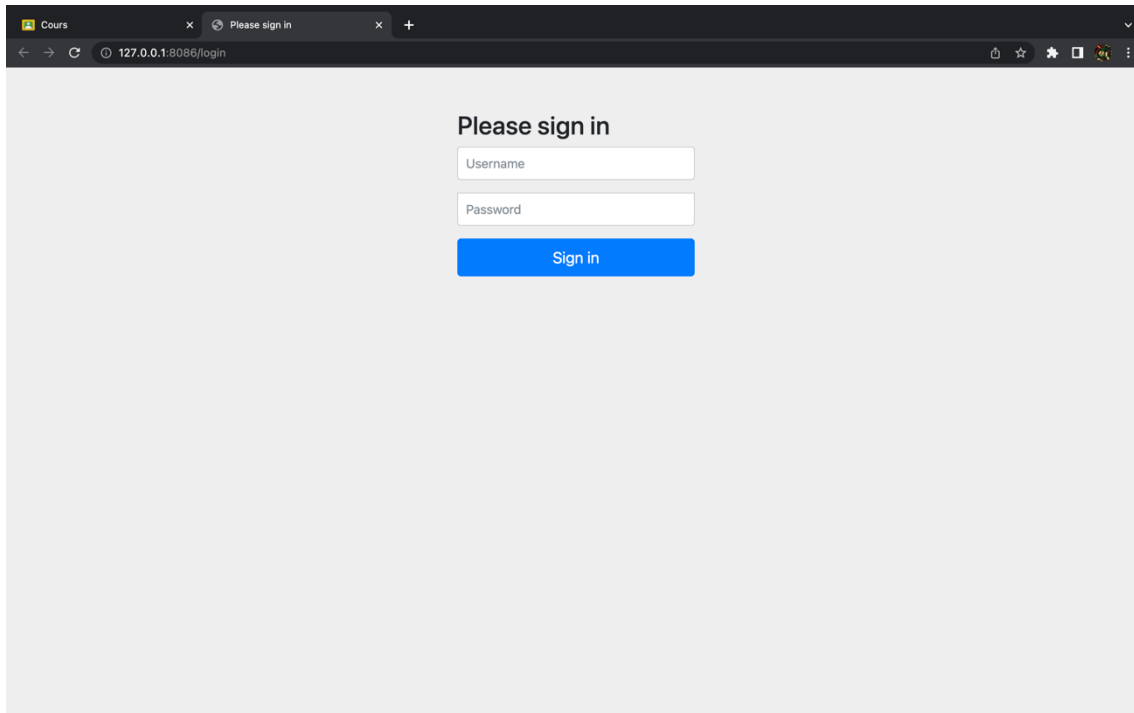
En utilisant ces technologies et les outils , nous avons pu développer une application web robuste, sécurisée et facile à maintenir.





Interfaces :

Page d'authentification



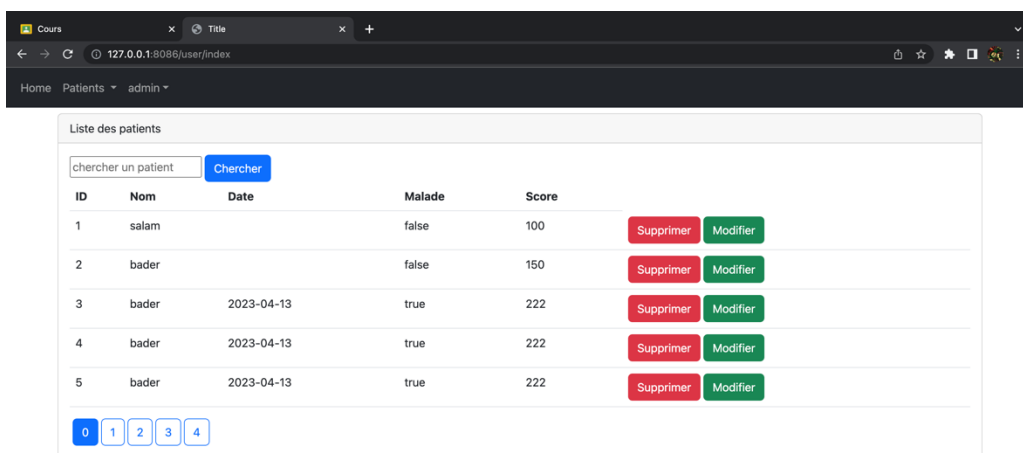
Please sign in

Username

Password

Sign in

Index (avec le rôle admin)



Liste des patients

chercher un patient [Chercher](#)

ID	Nom	Date	Malade	Score		
1	salam		false	100	Supprimer	Modifier
2	bader		false	150	Supprimer	Modifier
3	bader	2023-04-13	true	222	Supprimer	Modifier
4	bader	2023-04-13	true	222	Supprimer	Modifier
5	bader	2023-04-13	true	222	Supprimer	Modifier

[0](#) [1](#) [2](#) [3](#) [4](#)

Interfaces :

Index (avec le rôle user1)

ID	Nom	Date	Malade	Score
1	salam		false	100
2	bader		false	150
3	bader	2023-04-13	true	222
4	bader	2023-04-13	true	222
5	bader	2023-04-13	true	222

Modifier Patient

ID 3

Nom

bader

Date Naissance

13/04/2023

Malade ☒

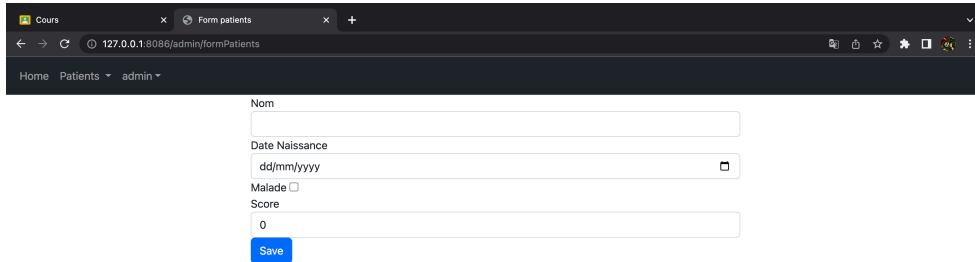
Score

222

Save

Interfaces :

Ajouter un patient



The screenshot shows a web browser window with two tabs: 'Cours' and 'Form patients'. The address bar displays '127.0.0.1:8086/admin/formPatients'. The browser's navigation bar includes links for 'Home', 'Patients', and 'admin'. The form itself is titled 'Ajouter un patient' and contains the following fields:

- Nom**: A text input field.
- Date Naissance**: A date picker with a calendar icon and the format 'dd/mm/yyyy'.
- Malade**: A checkbox.
- Score**: A text input field containing the value '0'.
- Save**: A blue button.

Conclusion:

En conclusion, nous avons développé une application de gestion des patients basée sur Spring MVC, Spring Data, JPA, Hibernate et Spring Security. L'application offre une interface utilisateur conviviale et des fonctionnalités avancées pour gérer les informations des patients, telles que l'affichage, la recherche, la mise à jour et la suppression des données des patients. En outre, la sécurité est prise en compte à travers l'utilisation de Spring Security pour protéger les informations sensibles des patients contre tout accès non autorisé.

Le développement de cette application a été un défi, mais il nous a permis de mettre en pratique les connaissances acquises en matière de développement web et de sécurité informatique. Nous espérons que cette application pourra être utile pour les professionnels de la santé et les patients qui cherchent à gérer efficacement les informations de santé tout en respectant les normes de sécurité et de confidentialité.