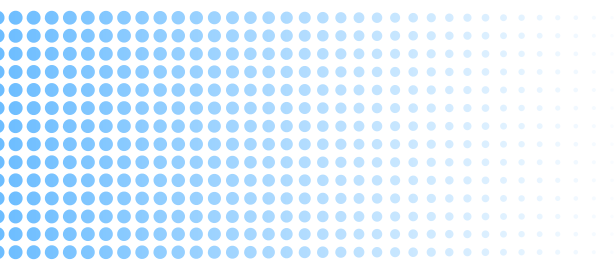




MODULE JAVA JEE

RAPPORT

SYSTÈME DE RÉSERVATION DE SALLES



Préparé par :
Hala TOB

Introduction

Ce rapport présente le développement d'une application web dédiée à la gestion des réservations de salles dans un centre d'affaires. L'application a été conçue pour répondre aux besoins des utilisateurs, notamment les administrateurs et les clients, en leur permettant de gérer efficacement les réservations. L'application utilise des technologies modernes telles qu'Hibernate pour la gestion des données, des Servlets pour la logique métier, et JSP pour l'interface utilisateur. Ce document est structuré par interface, détaillant les fonctionnalités mises en œuvre et les choix techniques adoptés.



Enoncé Du Projet

Gestion d'un Système de Réservation de Salles

Objectif:

Créer une application web pour la gestion des réservations de salles dans un centre d'affaires. Le projet doit utiliser Hibernate pour la gestion des données, les Servlets pour la logique métier, et JSP pour l'interface utilisateur. Exigences fonctionnelles :

Utilisateurs :

Admin : Gestion des salles et des utilisateurs.

Client : Réservation et consultation des salles disponibles.

Modules à développer:

Authentification :

Connexion et déconnexion (admin et client).

Gestion des salles (Admin) :

Ajouter, modifier et supprimer une salle.

Consulter la liste des salles.

Enoncé Du Projet

Réservation des salles (Client) :

Rechercher des salles par critères (date, capacité, équipements).

Réserver une salle disponible.

Annuler une réservation.

Afficher l'historique des réservations.

Tableau de bord (Admin) :

Consulter les statistiques (nombre de réservations, salles les plus utilisées, etc.).

Exigences techniques :

Base de données :

Créer une base de données relationnelle pour stocker les informations sur les utilisateurs, les salles, et les réservations.

Utiliser Hibernate pour le mapping objet-relationnel (ORM).

Back-End :

Utiliser des Servlets pour gérer la logique métier.

Implémenter un modèle MVC (Modèle-Vue-Contrôleur).

Enoncé Du Projet

Front-End :

Utiliser JSP pour les vues et les formulaires.

Ajouter un minimum de CSS pour une interface utilisateur agréable.

Validation et Sécurité :

Implémenter des validations côté serveur et côté client (HTML/JavaScript).

Protéger les pages avec un filtre servlet pour vérifier les rôles (admin/client).

Livrables :

Code source de l'application.

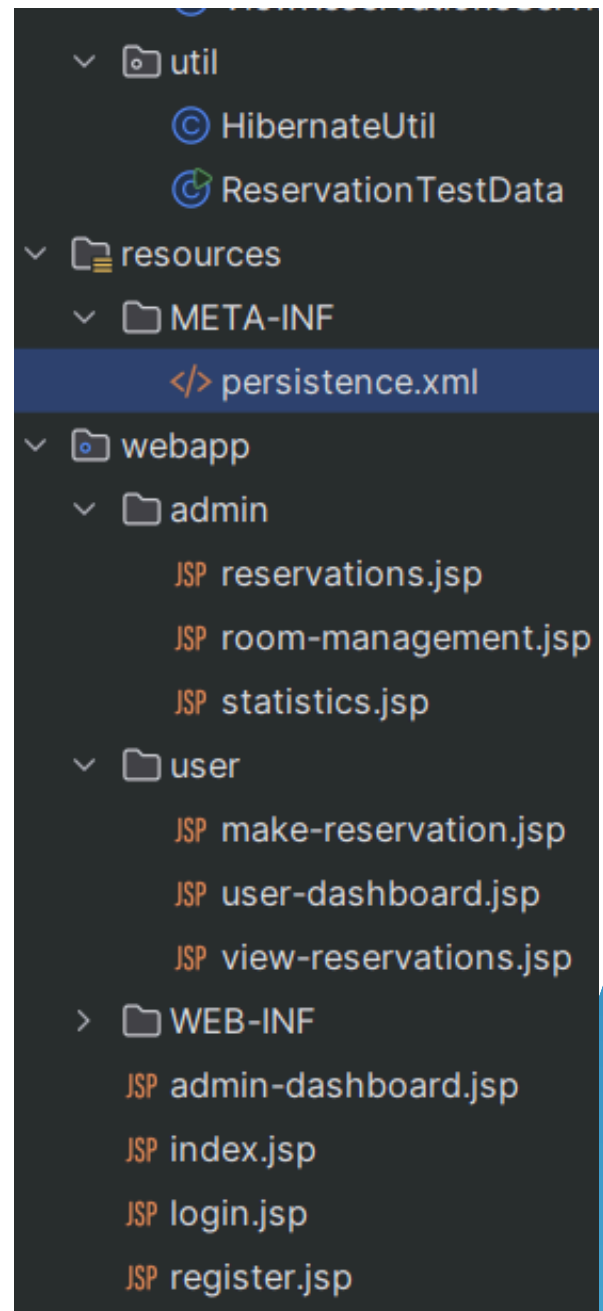
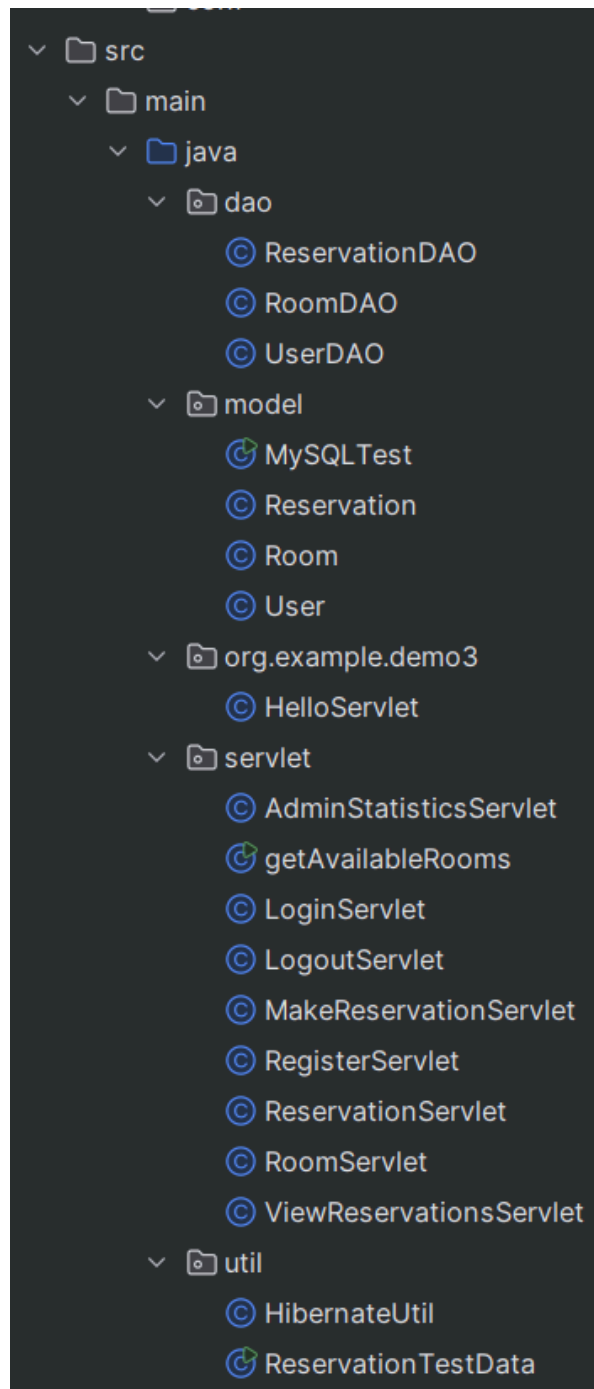
Script SQL pour la création de la base de données.

Manuel utilisateur décrivant les fonctionnalités de l'application.

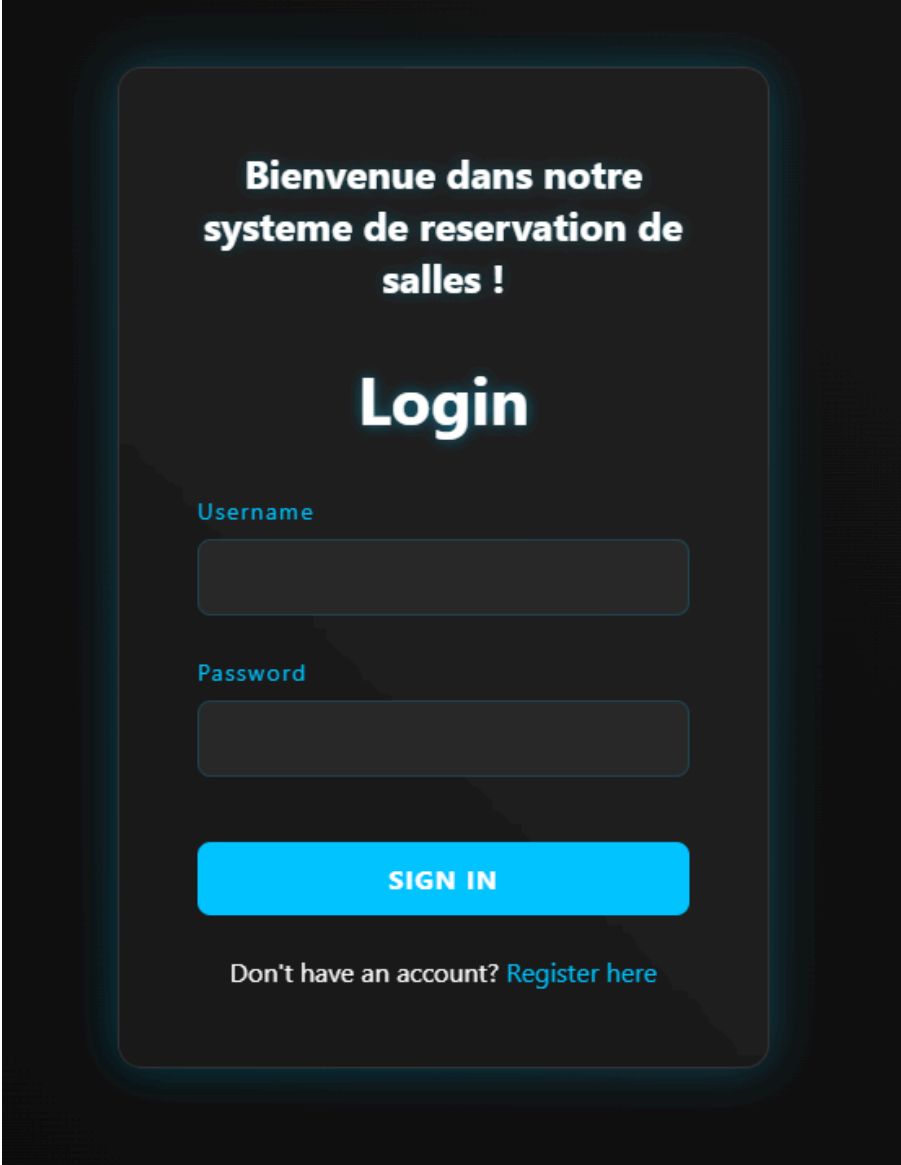
ENVIRONNEMENT



STRUCTURE DU PROJET



Interface LOGIN

A dark-themed login interface with a glowing blue border. It features a welcome message in French, a 'Login' title, input fields for 'Username' and 'Password', a bright blue 'SIGN IN' button, and a link to 'Register here' for new users. The interface is set against a background with blue geometric shapes in the bottom right corner.

**Bienvenue dans notre
systeme de reservation de
salles !**

Login

Username

Password

SIGN IN

Don't have an account? [Register here](#)

Interface Utilisateur

1. Page de Connexion (login.jsp)

Fonctionnalité : Permet aux utilisateurs (admin et clients) de se connecter à l'application.

Technologies Utilisées :

JSP : Pour l'affichage dynamique du formulaire de connexion.

HTML/CSS : Pour le design et la mise en forme de la page.

Servlets : Pour traiter les demandes de connexion.

Détails du Code :

Utilisation d'un formulaire HTML pour recueillir le nom d'utilisateur et le mot de passe.

Validation côté serveur pour assurer la sécurité des informations d'identification.

2. Gestion des Salles (manageRooms.jsp)

Fonctionnalité : Permet aux administrateurs d'ajouter, modifier ou supprimer des salles.

Technologies Utilisées :

JSP : Pour afficher les formulaires de gestion des salles.

Servlets : Pour gérer les requêtes liées aux opérations CRUD (Créer, Lire, Mettre à jour, Supprimer).

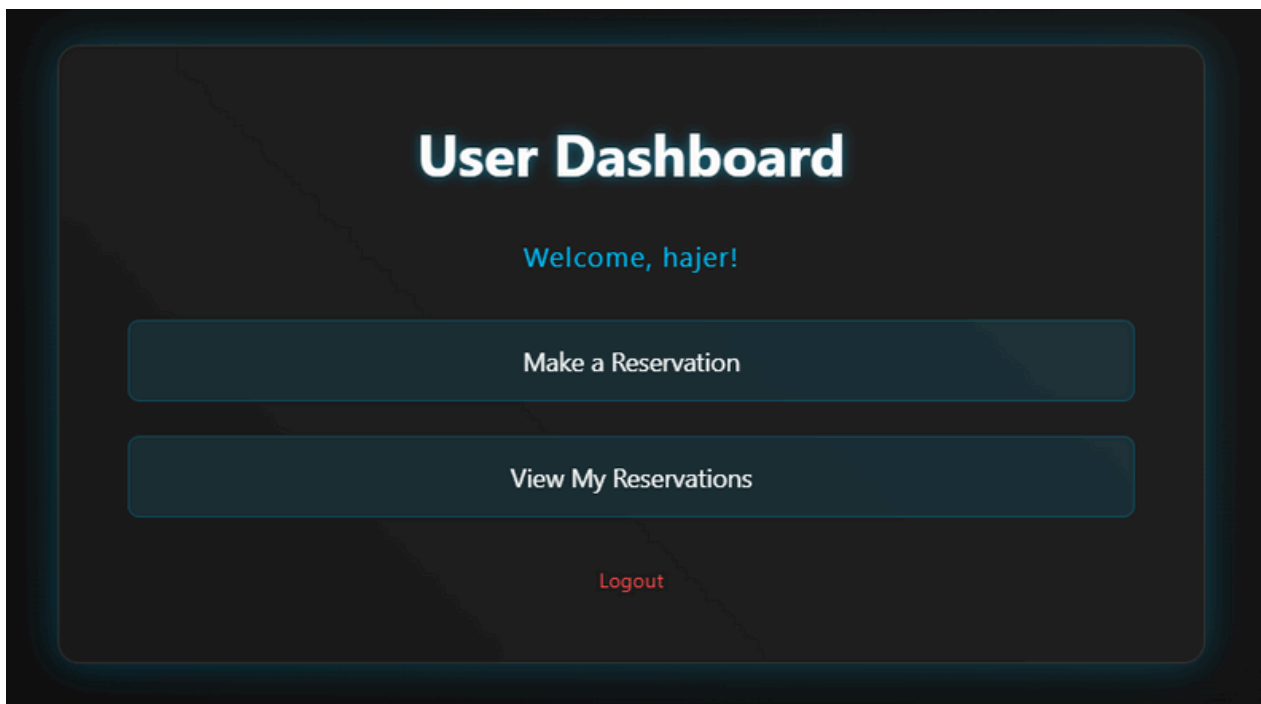
Détails du Code :

Formulaires HTML pour entrer les détails de la salle (nom, capacité, équipements).

Utilisation de tables pour afficher les salles existantes avec des options d'édition et de suppression.



Interface user-dashboard



3. Réservation des Salles (reserveRoom.jsp)

Fonctionnalité : Permet aux clients de rechercher et réserver des salles disponibles.

Technologies Utilisées :

JSP : Pour présenter un formulaire de recherche et une liste des salles disponibles.

Servlets : Pour traiter les demandes de réservation.

Détails du Code :

Formulaire permettant aux utilisateurs de spécifier des critères tels que la date et la capacité.

Affichage dynamique des résultats de recherche avec possibilité de réservation.

4. Tableau de Bord Administratif (adminDashboard.jsp)

Fonctionnalité : Fournit aux administrateurs un aperçu des statistiques concernant les réservations.

Technologies Utilisées :

JSP : Pour générer dynamiquement le tableau de bord avec des statistiques pertinentes.

Servlets : Pour récupérer les données nécessaires à partir de la base de données via Hibernate.

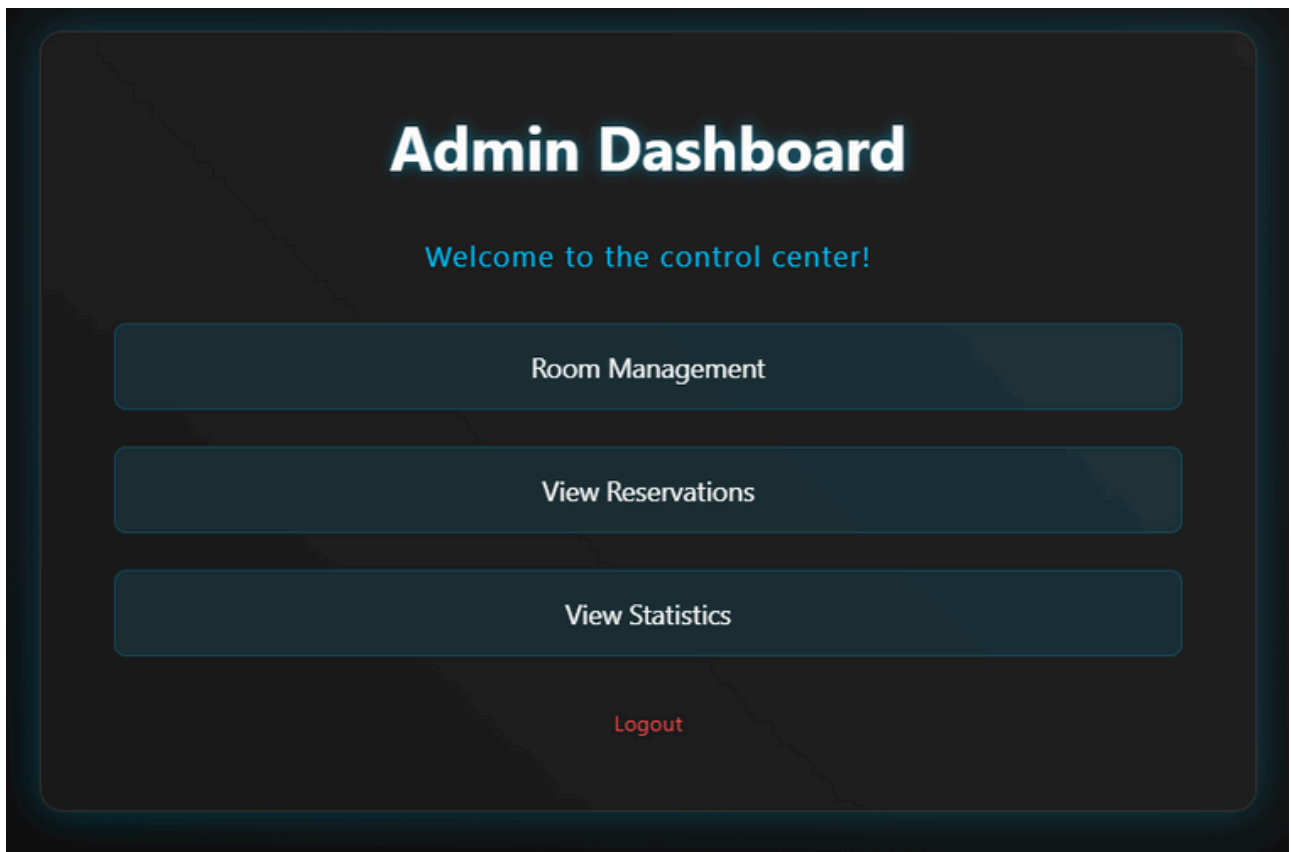
Détails du Code :

Tables affichant le nombre total de réservations, les salles les plus réservées, etc.

Graphiques ou tableaux pour visualiser les données.



Interface admin-dashboard



Backend

Utilisation d'Hibernate

Hibernate est utilisé comme ORM (Object Relational Mapping) pour faciliter l'interaction entre l'application Java et la base de données MySQL. Cela permet une gestion efficace des entités telles que User, Room, et Reservation.

Le fichier persistence.xml configure la connexion à la base de données, spécifiant le driver JDBC, l'URL, l'utilisateur et le mot de passe.

Servlets

Les Servlets sont utilisés pour gérer la logique métier. Chaque action utilisateur déclenche un Servlet correspondant qui traite la demande, interagit avec la base de données via Hibernate et renvoie une réponse appropriée à l'utilisateur.

Base de Données

Structure Relationnelle

Une base de données relationnelle a été créée pour stocker les informations sur les utilisateurs, les salles et les réservations. Les tables principales comprennent users, rooms, et reservations.

Les relations entre ces tables sont gérées par Hibernate, permettant une manipulation simple des données.

Validation et Sécurité

Validation Côté Client et Serveur

Des validations sont mises en place tant côté client (via HTML/JavaScript) que côté serveur (via Servlets) pour garantir que toutes les entrées utilisateur sont correctes avant traitement.

Sécurité

Un filtre Servlet est utilisé pour protéger certaines pages en vérifiant le rôle de l'utilisateur (admin ou client) avant d'accorder l'accès.

Détails Techniques Supplémentaires

Configuration Technique

Structure du Projet

src/main/java

dao: Contient les classes Data Access Object pour interagir avec la base de données.

model: Contient les classes représentant les entités (User, Room, Reservation).

servlet: Contient les Servlets qui gèrent la logique métier.

util: Contient HibernateUtil pour gérer la configuration Hibernate.

META-INF

persistence.xml: Configuration d'Hibernate définissant l'unité de persistance.


webapp

Dossier contenant toutes les pages JSP (admin, user).

Base de Données

Utilisation de XAMPP avec phpMyAdmin pour gérer MySQL.

La base est configurée avec trois tables principales: users, rooms, et reservations.



Conclusion

Le projet "Gestion d'un Système de Réservation de Salles" a permis de développer une application web fonctionnelle qui répond aux besoins spécifiques d'un centre d'affaires. Grâce à l'utilisation conjointe d'Hibernate, Servlets et JSP, nous avons créé une interface utilisateur intuitive tout en assurant une logique métier robuste. Ce système facilite non seulement la gestion des réservations par les administrateurs mais offre également une expérience fluide aux clients lors de leurs réservations. À l'avenir, il serait bénéfique d'envisager l'intégration d'autres fonctionnalités telles que des notifications par e-mail ou un calendrier interactif pour améliorer encore plus l'expérience utilisateur.

