Sup Galilée – USPN

Programmation Web Avancée – S7

AppAssoSport
Rapport de projet

Élèves :

Baba SOW
Alioune SECK

Enseignant :

Benoît Villa

19 janvier 2025

Table des matières

1 Introduction

Le projet AppAssoSport est une application web conçue pour simplifier la gestion des adhérents et des groupes d'une association sportive. L'application propose des fonctionnalités de gestion via une interface utilisateur et des API REST, offrant ainsi une base pour des applications futures.

L'objectif principal est de centraliser les informations administratives tout en permettant une interaction fluide avec les données. Ce rapport présente les fonctionnalités de l'application, son architecture, les technologies utilisées, ainsi que les résultats obtenus.

2 Description des fonctionnalités

Pour répondre au cahier des charges, nous avons développé les fonctionnalités suivantes pour l'application AppAssoSport.

2.1 Gestion des adhérents

- **Ajouter un adhérent** : Enregistrement des informations personnelles et administratives (nom, prénom, email, téléphone, etc.).
- **Modifier un adhérent** : Mise à jour des données existantes.
- **Afficher les détails d'un adhérent** : Consultation des informations complètes.
- **Supprimer un adhérent** : Retrait définitif d'un adhérent de la base.
- **Gestion des groupes pour un adhérent** :
  - Assigner un adhérent à un groupe.
  - Retirer un adhérent d'un groupe.

2.2 Gestion des groupes

- **Créer un groupe** : Définition du nom d'un groupe.
- **Modifier un groupe** : Mise à jour des informations.
- **Lister les groupes** : Affichage des groupes existants et des adhérents associés.
- **Supprimer un groupe** : Gestion des adhérents rattachés lors de la suppression.

2.3 API REST

Pour faciliter l'intégration avec des applications futures, des API REST ont été développées :

- **Gestion des adhérents** :
  - Récupérer tous les adhérents : `GET /api/adherents`
  - Ajouter un adhérent : `POST /api/adherents`
  - Modifier un adhérent : `PUT /api/adherents/{id}`
  - Supprimer un adhérent : `DELETE /api/adherents/{id}`
- **Gestion des groupes** :
  - Récupérer tous les groupes avec leurs adhérents : `GET /api/groupes`

2.4 Interface utilisateur

L'interface utilisateur est faite pour être claire et ergonomique, avec une navigation intuitive entre les différentes fonctionnalités. La mise en page est améliorée grâce à l'utilisation d'une feuille de style CSS.

3 Technologies utilisées

Pour le développement de l'application **AppAssoSport**, nous avons utilisé les technologies suivantes :

— **Langages** : Java, HTML, CSS.

— **Frameworks et bibliothèques** : Jakarta EE (JPA, Servlet, JSP, REST).

— **Base de données** : MariaDB.

— **Serveur** : Apache Tomcat.

— **Outils de développement** : Maven pour la gestion des dépendances, Postman pour le test des API.

4 Architecture et organisation de l'application

4.1 Organisation générale

L'architecture de l'application suit une organisation structurée basée sur les principes du modèle MVC. Elle est divisée en deux parties principales : Backend et Frontend.

4.2 Fichier de configuration de persistance

pom.xml : La configuration principale de l'application est définie dans le fichier `pom.xml`, situé à l'emplacement `/AppAssoSport/pom.xml`. Ce fichier gère les dépendances nécessaires au projet, notamment pour la persistance, les servlets, les JSP et les API REST.

persistence.xml : Le fichier `persistence.xml` est situé à l'emplacement `/src/main/resources/META-INF/persistence.xml`. Ce fichier configure l'unité de persistance pour Hibernate et JPA. Il définit les paramètres de connexion à la base de données MariaDB (`AssoSport`) via l'utilisateur `assoSportAdmin`. De plus, il synchronise automatiquement les tables avec les entités JPA grâce à l'option `update`. Enfin, il affiche les requêtes SQL dans les logs pour faciliter le suivi et le débogage.

4.3 Backend

4.3.1 Entités JPA

Adherent.java : Le fichier est dans : /src/main/java/entity/Adherent.java Cette entité représente un adhérent avec ses attributs (nom, prénom, email, numéro de téléphone, adresse, date de naissance, date de paiement) et sa relation avec un groupe. Chaque adhérent est associé à un groupe via une relation `@ManyToOne`.

Groupe.java : Le fichier est dans /src/main/java/entity/Groupe.java. Cette entité représente un groupe d'adhérents. Chaque groupe contient une liste d'adhérents. Chaque groupe peut contenir plusieurs adhérents via une relation `@OneToMany`.

4.3.2 DAO (Data Access Object)

AdherentDAO.java : Le fichier se trouve dans /src/main/java/dao/AdherentDAO.java. Il permets de gèrer les interactions avec la base de données pour les adhérents. Les méthodes principales implémentées sont :

- `create(Adherent adherent)` : Ajoute un nouvel adhérent.
- `findById(Long id)` : Trouve un adhérent par son ID.
- `findAll()` : Récupère tous les adhérents.
- `delete(Long id)` : Supprime un adhérent.

GroupeDAO.java : Le fichier se trouve dans /src/main/java/dao/GroupeDAO.java. Il permets de gèrer les interactions avec la base pour les groupes. Les méthodes principales implémentées sont :

- `create(Groupe groupe)` : Ajoute un nouveau groupe.
- `findById(Long id)` : Trouve un groupe par son ID.
- `findAll()` : Récupère tous les groupes.
- `delete(Long id)` : Supprime un groupe.

4.3.3 Servlets

AdherentServlet.java : Le fichier se trouve dans /src/main/java/servlet/AdherentServlet.java Il permets de gèrer les requêtes HTTP pour les opérations liées aux adhérents. Les méthodes principales implémentées sont :

- `doGet` : Affiche la liste des adhérents ou les détails d'un adhérent.
- `doPost` : Gère l'ajout, la modification ou la suppression d'un adhérent.

GroupeServlet.java Le fichier se trouve dans /src/main/java/servlet/GroupeServlet.java. Il permets de gèrer les requêtes HTTP pour les groupes.

4.3.4 API REST

AdherentResource.java Le fichier se trouve dans /src/main/java/resource/AdherentResource.java. Il fournit des points d'entrée REST pour gérer les adhérents.

4.3.5 GroupeResource.java

Le fichier se trouve dans /src/main/java/resource/GroupeResource.java. Il fournit des points d'entrée REST pour gérer les groupes.

4.4 Frontend

Le dossier parent pour les documents du Frontend est `/src/main/webapp/`.

Pages JSP

- `index.jsp` : Page d'accueil avec des liens vers la gestion des adhérents et des groupes. Pour chaque entité dans /src/main/webapp/WEB-INF, nous avons les pages suivantes :
- `list.jsp` : Liste les adhérents/groupes avec des actions disponibles (modifier, supprimer, etc.).
- `form.jsp` : Formulaire pour ajouter ou modifier un adhérent/groupe.
- `details.jsp` : Affiche les détails complets d'un adhérent/groupe.

Un fichier style.css est utilisé pour améliorer la mise en page des pages JSP.

5 Résultats et démonstration

L'application **AppAssoSport** a été testée avec succès pour les fonctionnalités principales (NB : Pour la suppression on est obligé de restart le server Tomcat après suppression). Voici une démonstration des fonctionnalités de l'application (voir Annexes).

6 Conclusion

Le projet AppAssoSport constitue une solution complète et efficace pour la gestion des adhérents et des groupes au sein d'une association sportive. En combinant une interface utilisateur intuitive, des fonctionnalités robustes de gestion et des API REST extensibles, l'application répond aux besoins actuels et ouvre la voie à des développements futurs. Les principales réalisations incluent :

- Une architecture solide basée sur le modèle MVC, facilitant la maintenance et l'évolution du projet.
- Des fonctionnalités complètes de gestion, allant de l'ajout à la suppression des adhérents et groupes, avec une gestion des relations entre eux.
- L'intégration des API REST, permettant une extension future vers des applications mobiles ou des services tiers.

Le projet a également permis de relever des défis techniques, tels que la configuration de la persistance avec JPA et Hibernate, la gestion des relations en base de données.

Perspectives d'avenir Pour aller plus loin, les axes d'amélioration suivants pourraient être explorés :

— Intégration d'un système d'authentification pour sécuriser l'accès à l'application.

— Rapports statistiques dynamiques pour un suivi détaillé des adhérents et groupes.

— Développement d'une application mobile exploitant les API REST existantes.

En conclusion, AppAssoSport constitue une base solide et évolutive, adaptée aux besoins d'une association sportive moderne. Ce projet illustre parfaitement l'alliance entre technologie et gestion efficace.

A Annexes

A.1 Ajout d'un adhérent

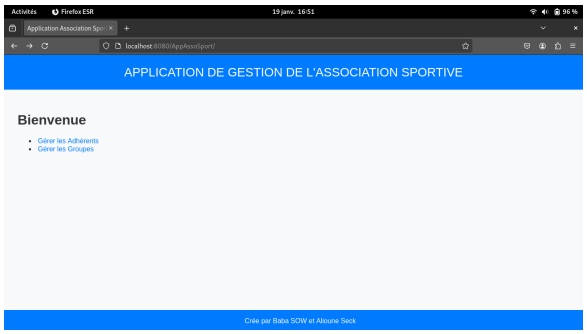

FIGURE 1 – Ajout d'un adhérent 1

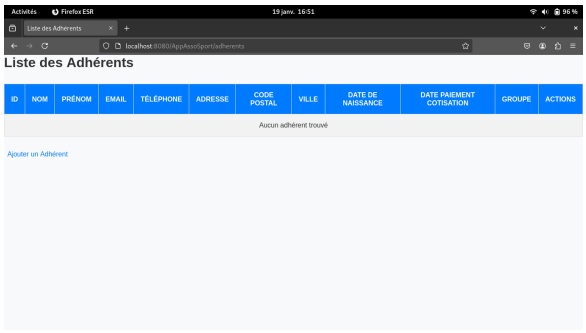

FIGURE 2 – Ajout d'un adhérent 2

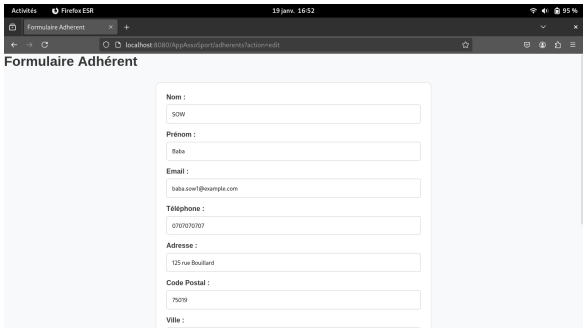

FIGURE 3 – Ajout d'un adhérent 3

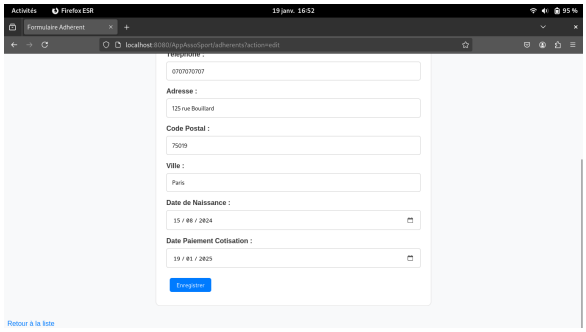

FIGURE 4 – Ajout d'un adhérent 4

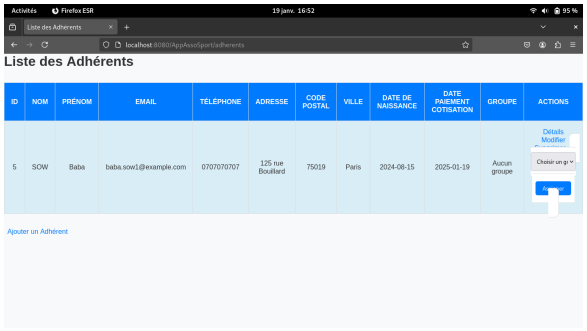

FIGURE 5 – Ajout d'un adhérent 5

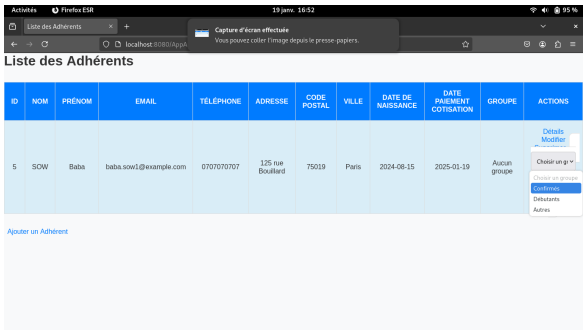

FIGURE 6 – Assigner à un groupe 1

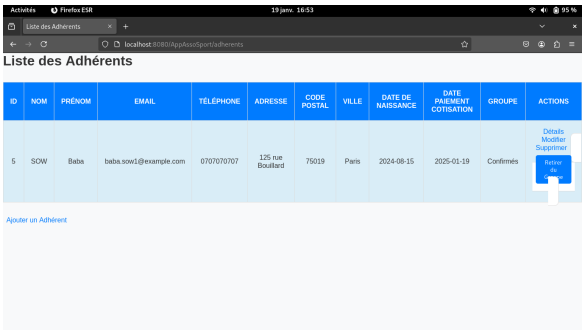

FIGURE 7 – Assigner à un groupe 2

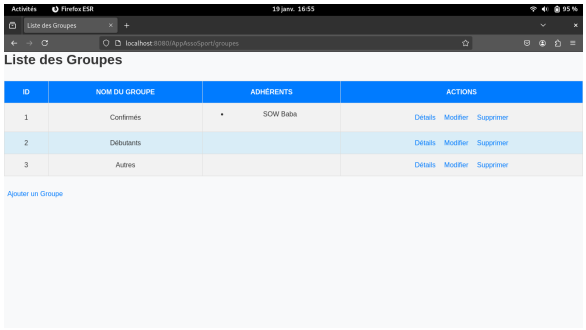

FIGURE 8 – Assigner à un groupe 3

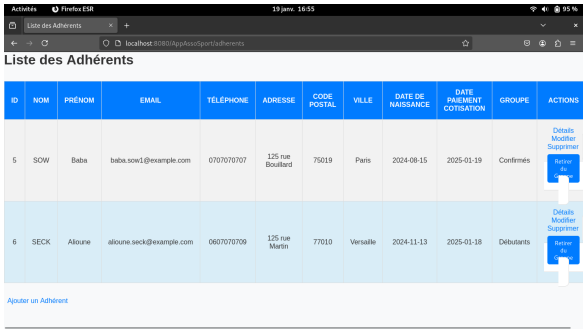

FIGURE 9 – Liste des adhérents d'un groupe 1

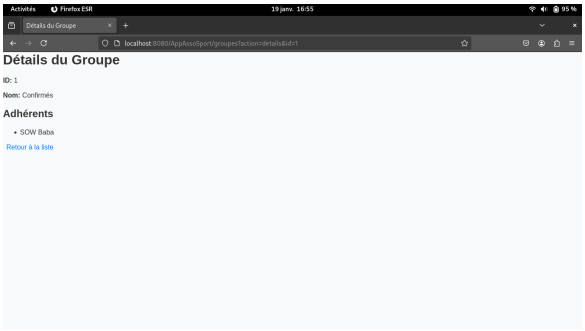

FIGURE 10 – Liste des adhérents d'un groupe 2

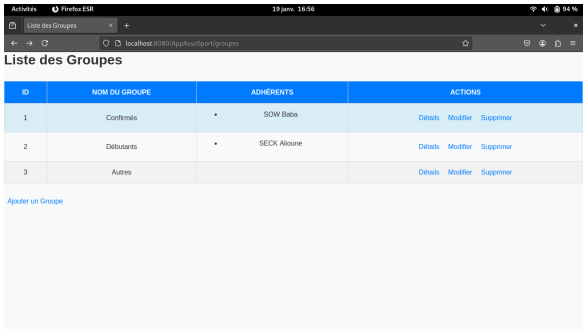

FIGURE 11 – Liste des adhérents d'un groupe 3

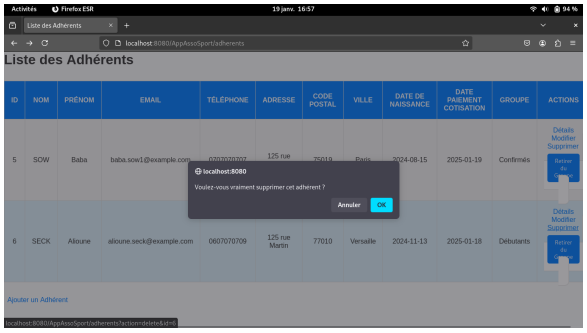

FIGURE 12 – Supprimer un adhérent 1

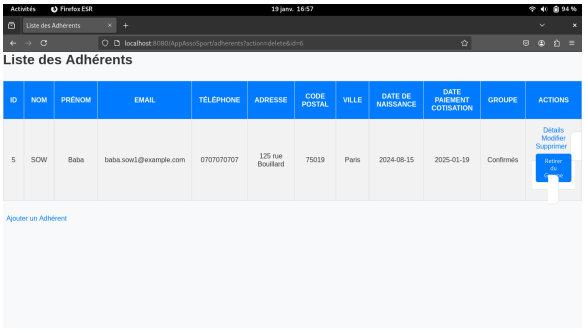

FIGURE 13 – Supprimer un adhérent 2

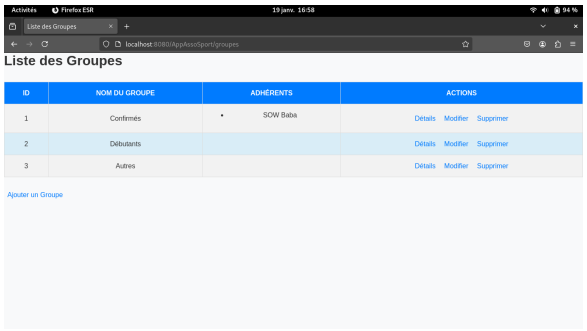

FIGURE 14 – Supprimer un adhérent 3

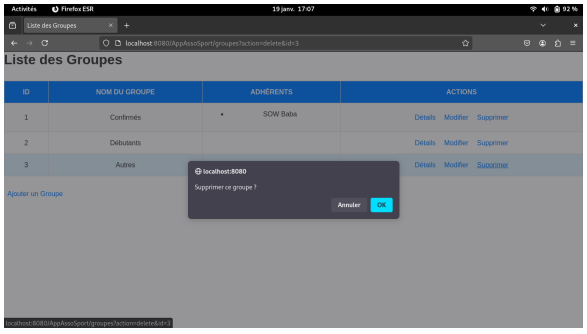

FIGURE 15 – Supprimer un groupe 1

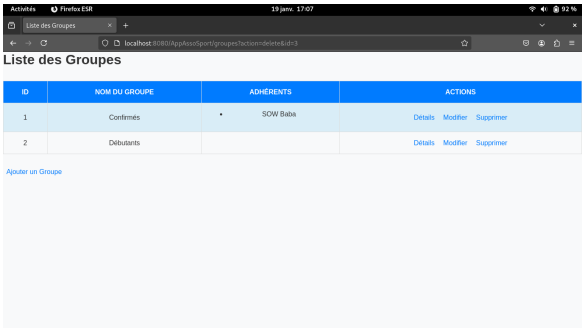

FIGURE 16 – Supprimer un groupe 2