

Projet DevOps – Gestion d’Abonnements

1. Introduction

L’application **Gestion d’Abonnements** permet aux utilisateurs de centraliser, gérer et analyser leurs abonnements récurrents. Cette version 0.2.0 s’inscrit dans la continuité de la version 0.1.0 et apporte des améliorations significatives au niveau de l’interface utilisateur, du tableau de bord et de la gestion des données.

2. Type d’application

Application Web avec back-end Java et API REST optionnelle.

3. Fonctionnalités pour la version 0.2.0

3.1. F1 : Nouvelle interface utilisateur (Glassmorphism 3.0)

L’interface graphique a été entièrement améliorée afin d’offrir une expérience utilisateur moderne et ergonomique :

- Effets de transparence et flou avancés (glassmorphism 3.0)
- Utilisation de gradients violets animés
- Cartes avec effets 3D et élévation dynamique
- Icônes animées lors des interactions
- Boutons avec animations et transitions fluides
- Barre de recherche avec animation de focus
- Micro-interactions modernes
- Interface responsive optimisée pour mobile, tablette et desktop

3.2. F2 : Système de thèmes personnalisables (version simple)

Un premier système de personnalisation visuelle a été introduit :

- Gestion des couleurs globales via des variables CSS
- Thème violet premium utilisé par défaut

- Préparation technique pour des thèmes plus avancés prévus dans les versions futures

3.3. F3 : Dashboard amélioré

Le tableau de bord a été enrichi afin d'améliorer la visibilité sur les abonnements :

- Affichage de quatre indicateurs clés (KPI) :
 - total des abonnements
 - abonnements actifs
 - abonnements expirés
 - alertes
- Mise en page moderne basée sur une grille
- Amélioration des couleurs de statut
- Calculs mis à jour dynamiquement à partir des données disponibles

3.4. F4 : Import / Export amélioré

De nouvelles fonctionnalités de gestion des données ont été ajoutées.

3.4.1. Export

- Export des abonnements au format JSON
- Export CSV simple compatible avec Excel et Google Sheets

3.4.2. Import

- Import des données au format JSON
- Validation basique des fichiers importés
- Prévisualisation des données avant importation

Les formats bancaires avancés (OFX, QIF) ne sont pas inclus dans cette version.

4. Données acquises / manipulées

- Données des abonnements (nom, dates, prix, catégorie)
- Statut des abonnements (actif, expiré, alerte)
- Données importées et exportées au format JSON et CSV

5. Interconnexions

- Front-end : HTML, CSS, JavaScript

- Back-end : Java
- API REST : Spark Java
- Persistance : fichier texte et fichiers JSON

6. Comment lancer le projet

6.1. Prérequis d'installation

- Java JDK 17
- Maven
- Un IDE (Visual Studio Code, IntelliJ IDEA, etc.)

6.2. Récupération du code

```
git clone <URL_DU_DEPOT>  
cd Projet-Dev-Ops
```

6.3. Compilation

```
mvn clean package
```

6.4. Lancement de l'application

```
mvn exec:java -Dexec.mainClass=com.example.abonnement.GestionAbonnements
```

6.5. Lancement de l'API REST (optionnel)

```
mvn exec:java -Dexec.mainClass=com.projet.api.ApiServer
```

7. Diagramme de classes

Première version du diagramme de classes du projet, illustrant les principales entités et leurs relations.