

Sujet de Mini-Projet : Gestion des Dossiers de Mutuelle avec Spring Batch et Calcul de Remboursement

Contexte du Projet

Dans un système de mutuelle, chaque dossier déposé par un assuré peut inclure une consultation et des traitements médicaux. Ce projet consiste à créer une application de traitement par lots avec **Spring Batch** pour automatiser la gestion des dossiers, valider les données, et calculer les montants à rembourser à partir d'une base de médicaments référentiels.

Objectif

Développer une application batch en utilisant Spring Batch pour :

1. Lire des dossiers de mutuelle depuis une première source de données ; un fichier JSON.
2. Valider les informations essentielles de chaque dossier.
3. Calculer le total des remboursements, en appliquant un pourcentage sur le prix de la consultation et en utilisant des prix de référence et taux de remboursement pour chaque médicament prescrit.
4. Écrire les dossiers traités dans une base de données, contenant les données de l'assurée, pour archivage.

Fonctionnalités

1. **Lecture des Dossiers** : Utilisez un `JsonItemReader` pour lire les données JSON. Chaque dossier contient les informations suivantes :
 - **Assuré** : nom, numéro d'affiliation, immatriculation
 - **Bénéficiaire** : nom, lien de parenté avec l'assuré, date de dépôt du dossier
 - **Consultation et frais** : montant total des frais, prix de la consultation, nombre de pièces jointes
 - **Traitements** : chaque traitement comprend le code barre, nom, type de médicament, prix, et disponibilité. pourcentage base sur le referentiel et non pas le prix
2. **Base de Médicaments Référentiels** : Créez une source de données pour les médicaments référentiels, qui inclut :
 - Le nom du médicament
 - Le prix de référence
 - Le pourcentage de remboursement applicable
3. **Validation des Données** :
 - Le **nom de l'assuré** et le **numéro d'affiliation** ne doivent pas être vides.
 - Le **prix de la consultation** et le **montant total des frais** doivent être positifs.
 - La **liste des traitements** doit être présente et non vide.

4. Calcul du Remboursement :

- Appliquez un pourcentage fixe de remboursement sur le prix de la consultation.
- Pour chaque traitement, mappez le médicament prescrit à son équivalent référentiel pour calculer le remboursement en appliquant le pourcentage défini dans le référentiel.

5. Enchaînement des Processeurs :

- Utilisez un CompositeItemProcessor pour enchaîner un ValidationProcessor et un CalculProcessor.
- ValidationProcessor : Vérifie les contraintes de validation.
- CalculProcessor : Calcule le total du remboursement basé sur les médicaments référentiels et le pourcentage sur la consultation.

Structure des Sous-Processors

- **ConsultationProcessor** : Calcule le remboursement de la consultation en appliquant un pourcentage fixe sur le prix de la consultation.
- **TraitementMappingProcessor** : Mappe chaque traitement du dossier au médicament référentiel et vérifie sa disponibilité dans le référentiel.
- **TraitementRemboursementProcessor** : Calcule le montant remboursable pour chaque médicament référencé en appliquant le pourcentage de remboursement spécifié.
- **TotalRemboursementProcessor** : Additionne les remboursements de la consultation et des traitements pour calculer le total du remboursement.
- **Écriture des Données** : Enregistrez les dossiers traités avec le montant total de remboursement dans une base de données ou dans un fichier de sortie pour archivage.

6. **Écriture des Données** : Enregistrez les dossiers traités avec le montant total de remboursement dans une base de données ou dans un fichier de sortie pour archivage.

Exemple de fichier json :

```
[
  {
    "nomAssure": "Ibrahimi",
    "numeroAffiliation": "AFF123456",
    "immatriculation": "IMM098765",
    "lienParente": "fils",
    "montantTotalFrais": 150.0,
    "prixConsultation": 50.0,
    "nombrePiecesJointes": 3,
    "nomBeneficiaire": "Omar",
    "dateDepotDossier": "2024-11-10",
    "traitements": [
      {
        "codeBarre": "1234567890",
        "existe": true,
        "nomMedicament": "Paracétamol",
        "typeMedicament": "Antalgique",
        "prixMedicament": 5.0
      },
      {
        "codeBarre": "0987654321",
        "existe": false,
        "nomMedicament": "Ibuprofène",
        "typeMedicament": "Antalgique",
        "prixMedicament": 4.5
      }
    ],
    "date_consultation": "2024-11-10",
    "date_rejet": "2024-11-10"
  }
]
```

```

    {
      "codeBarre": "0987654321",
      "existe": false,
      "nomMedicament": "Ibuprofène",
      "typeMedicament": "Anti-inflammatoire",
      "prixMedicament": 8.0
    }
  ],
},
{
  "nomAssure": "Farouk",
  "numeroAffiliation": "AFF654321",
  "immatriculation": "IMM123098",
  "lienParente": "épouse",
  "montantTotalFrais": 200.0,
  "prixConsultation": 60.0,
  "nombrePiecesJointes": 2,
  "nomBeneficiaire": "Samira",
  "dateDepotDossier": "2024-11-11",
  "traitements": [
    {
      "codeBarre": "1122334455",
      "existe": true,
      "nomMedicament": "Amoxicilline",
      "typeMedicament": "Antibiotique",
      "prixMedicament": 12.0
    }
  ]
}
]

```

Remarque :

Ce mini-projet constitue un premier draft visant à fournir aux étudiants une base pour développer un système de gestion des remboursements de dossiers de mutuelle à l'aide de Spring Batch. L'architecture, les spécifications et les différents composants (lecteurs, processeurs, validateurs, etc.) sont présentés de manière modulaire pour initier le traitement par lots, avec une attention particulière sur la séparation des responsabilités. Ce draft peut être enrichi par des ajustements selon les besoins métier spécifiques et approfondi pour aborder des cas d'usage avancés, des contrôles qualité ou des règles de calcul plus complexes, permettant ainsi aux étudiants de personnaliser et d'affiner leur solution.