**Rapport sur la Gestion de Bibliothèque**

**Introduction**

Le système de gestion de bibliothèque est un outil essentiel pour organiser efficacement les ressources d'une bibliothèque, incluant le prêt de livres, la gestion des membres et la traçabilité des ouvrages. Ce rapport explore les principaux aspects fonctionnels et techniques d'un tel système, en mettant l'accent sur son architecture, ses fonctionnalités clés et son impact sur l'expérience utilisateur.

**Architecture du Système**

Le système de gestion de bibliothèque est généralement basé sur une architecture client-serveur, où plusieurs modules interagissent pour assurer un fonctionnement fluide. Les composants principaux incluent :

1. un fichier json : Centralise les informations sur les livres, les membres et les transactions.

2. Module de prêt et de retour : fait la gestion de prêt et retour de livre qui son stocker sur le fichier livres

**1.1 La gestion de livres**

**Fonctions de Validation des Données**

1. **Compter les Caractères (compterLesCaracteres)** :
   * Vérifie si une chaîne a entre 5 et 100 caractères.
2. **Doit Être une Chaîne (doitEtreUneChaine)** :
   * Vérifie si la chaîne n'est pas entièrement numérique.
3. **Valider Titre, Auteur, Genre (validerTitre, validerAuteur, validerGenre)** :
   * Utilisent les fonctions précédentes pour valider les chaînes entrées pour titre, auteur et genre.
4. **Valider Tout (validerTout)** :
   * Vérifie si toutes les informations sur un livre (titre, auteur, genre) sont valides.

**Gestion des Livres**

1. **Récupérer Tous les Livres (recupererTousLesLivres)** :
   * Lit le fichier JSON livres.json et retourne une liste de tous les livres.
2. **Ajouter Livre (ajouterLivre)** :
   * Demande à l'utilisateur de saisir les détails d'un livre.
   * Vérifie si le livre est déjà enregistré.
   * Génère un nouvel ID pour le livre et l'ajoute à la liste des livres.
   * Appelle preparerEnregistrement pour enregistrer le livre dans le fichier.
3. **Préparer Enregistrement (preparerEnregistrement)** :
   * Récupère les livres existants, ajoute le nouveau livre, et enregistre la liste mise à jour dans le fichier.
4. **Enregistrer Dans le Fichier (enregistrerDansLeFichier)** :
   * Enregistre une liste de livres dans le fichier livres.json en format JSON.
5. **Afficher Tout (afficherTout)** :
   * Affiche tous les livres disponibles ou archivés en fonction du choix de l'utilisateur.
6. **Rechercher un Livre (rechercherUnLivre)** :
   * Permet de rechercher un livre par titre, auteur ou genre.
7. **Archiver un Livre (archiverUnLivre)** :
   * Marque un livre comme non disponible (archivé).
8. **Supprimer un Livre (supprimerUnLivre)** :
   * Supprime définitivement un livre de la liste.

**Gestion des Emprunts et Retours de Livres**

1. **Emprunter Livre (emprunter\_livre)** :
   * Enregistre les détails d'un livre emprunté par un utilisateur dans users.json.
2. **Retourner Livre (retourner\_livre)** :
   * Marque un livre comme retourné et enregistre la date de retour dans users.json.

**Fonctionnalités Manquantes**

* Le code ne gère pas la vérification de la disponibilité d'un livre avant de l'emprunter.
* Il n'y a pas de gestion des erreurs détaillée pour les entrées utilisateur incorrectes ou manquantes.
* La gestion des utilisateurs (users.json) n'est pas intégrée complètement dans les opérations de prêt et de retour.

**Conclusion**

Ce rapport décrit une application de gestion de bibliothèque simple mais fonctionnelle en Python, utilisant des fichiers JSON pour le stockage des données. Il fournit des fonctionnalités de base telles que l'ajout, la recherche, l'archivage et la suppression de livres, ainsi que la gestion des emprunts et des retours. Pour une application plus robuste, des améliorations pourraient être apportées pour gérer plus efficacement les interactions utilisateur et les validations de données.

Haut du formulaire

Bas du formulaire