

# Projet Programmation en classe avec Python

Imène Kerboua

M2 DS - 2022/2023

## 1 Énoncé du projet

### 1.1 Contexte

Dans ce projet, il vous est demandé de créer une application de gestion pour une bibliothèque.

La bibliothèque est caractérisée par son nom, son adresse et des horaires d'ouverture. Elle possède un stock de livres, plusieurs livres de différentes éditions, auteurs. Elle peut posséder également plusieurs copies d'un même livre.

La bibliothèque propose à ses usagers d'emprunter des livres. Un usager a le droit d'emprunter jusqu'à 5 livres à la fois. Chaque livre peut être emprunté pour une durée maximale de 25 jours.

Pour éviter l'afflux de personnes dans la bibliothèque, le personnel souhaite mettre en place un système de réservation de créneau pour emprunter un livre. Ainsi, si un utilisateur souhaite emprunter un livre, il devra réserver un créneau disponible pour se rendre à la bibliothèque. Pour retourner un livre, il n'est pas nécessaire de réserver un créneau.

La bibliothèque est gérée par un administrateur, qui ajoute de nouveaux livres à la bibliothèque, et en retire des plus anciens. Il peut également modifier les horaires d'ouverture de la bibliothèque, et les créneaux de réservation (ouvrir des créneaux de réservation et en fermer).

Pour pouvoir consulter la collection de livres disponibles dans la bibliothèque, les utilisateurs peuvent effectuer des recherches de livres par titre, auteurs, édition (si existante).

Chaque utilisateur possède sa propre session (s'y connecte et se déconnecte), où il pourra consulter les livres disponibles dans la bibliothèque, gérer ses emprunts (consulter, emprunter et retourner un livre), gérer ses réservations de créneaux (annuler un créneau, réserver un nouveau créneau, consulter les créneaux disponibles, etc.).

## Application Web

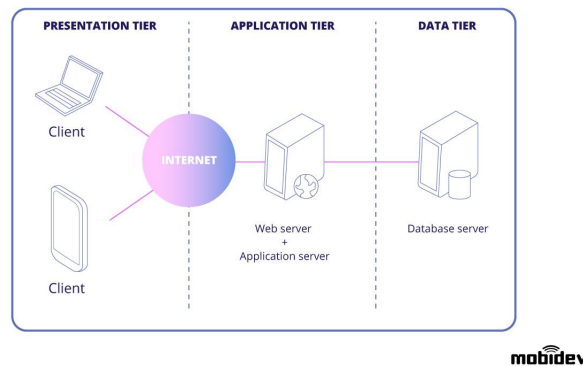


Figure 1: Application Web.

Nous allons simuler une application web, alors il nous faudra simuler un serveur qui gère tout le backend de l'application. Pour cela vous créerez une classe **Serveur** qui va gérer les utilisateurs, leurs actions, la bibliothèque, l'administrateur et toutes les actions liées à la gestion de la bibliothèque.

Les données de l'application devront être stockées, de sorte à ce qu'à chaque démarrage de l'application nous ayons toujours les utilisateurs inscrits, les livres disponibles, l'administrateur, etc. Utilisez alors des fichiers **pickle** ou **yaml** pour stocker vos structures de données (si vous utilisez **yaml**, il faudra convertir vos structures de données en **JSON**).

## Système de recommandation

Les gestionnaires de la bibliothèque souhaitent intégrer une fonctionnalité pour permettre de proposer (recommander) des livres à ses utilisateurs, en fonction de leurs emprunts. Dans un premier temps, la recommandation se fera que lorsque l'utilisateur a au moins un livre emprunté dans sa collection (pas de gestion de l'historique des emprunts). L'idée est de proposer des livres similaires à l'utilisateur, en fonction du thème (genre) du livre et auteur. Utilisez la similarité de Jaccard pour rapprocher les livres entre eux, vous devrez construire des vecteurs représentatifs des livres en utilisant les caractéristiques du thème (genre) des livre et l'information sur les auteurs.

### Similarité de Jaccard

La similarité de Jaccard est une méthode mathématique qui permet de calculer la similarité entre 2 ensembles A et B. Le coefficient de Jaccard équivaut à calculer un ratio entre le nombre d'éléments communs entre 2 ensembles sur la totalité de leurs éléments.

$$J(A, B) = \frac{|A \cap B|}{|A \cup B|} = \frac{|A \cap B|}{|A| + |B| - |A \cap B|} \quad (1)$$

Dans notre cas d'usage, nos deux ensembles représenteront, par exemple, les thèmes des livres de toute la bibliothèque et les thèmes des livres empruntés par l'utilisateur. Vous pouvez ajouter les noms des auteurs, ou créer des similarité entre les titres des livres (optionnel).

## 1.2 Diagramme entité-association

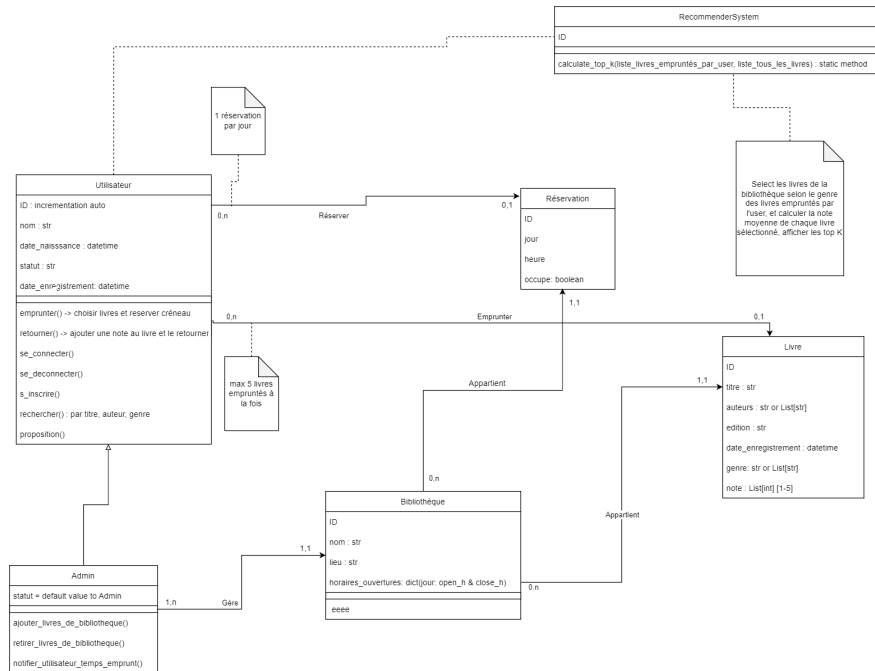


Figure 2: Diagramme de classes de l'application.

## 2 Consignes

- Créez un repository Git pour le projet, à me partager avant la date limite. Ce travail est à effectuer en binôme. La date limite de remise est avant de 05/01/2023.
- Créez l'application. Vous êtes libres de changer l'architecture proposée, ajouter des fonctionnalités, proposer une meilleure manière de faire de la recommandation.
- Pensez à bien utiliser la notion de modules et packages Python et toutes les notions de POO qu'on a vu en cours.