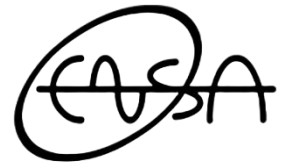




**Université Mohammed Premier
Ecole Nationale des Sciences
Appliquées Oujda**



Filière Génie Informatique

PROJET PYTHON

Réalisé par :

EL MNIAI Nissrine

Sous le thème :

**Mini projet d'application : Système de
Gestion de Bibliothèque**

Encadré par :

Mr. HAJA Zakaria

Année universitaire : 2024/2025

REMERCIEMENTS

Je souhaite débiter ce rapport en exprimant ma profonde gratitude envers mon professeur, Zakaria HAJA, qui a joué un rôle crucial dans le succès de ce projet. Ses conseils éclairés, son expertise et son engagement envers l'excellence académique ont été une source d'inspiration tout au long de ce parcours. Sa disponibilité, son soutien et sa capacité à transmettre des connaissances complexes ont grandement enrichi mon expérience d'apprentissage. Les échanges stimulants en classe, et l'attention portée aux détails ont contribué de manière significative à la qualité de ce projet. Je suis profondément reconnaissante d'avoir pu bénéficier de ses enseignements et de ses encouragements, qui marqueront durablement mon développement académique et professionnel.

Table des matières

1. Introduction	4
2. Analyse des besoins	4
1. Besoins fonctionnels	4
2. Besoins Techniques	4
3. Diagramme de classes UML	5
4. Modèle de données	5
5. Explication des algorithmes clés	5
1. emprunter_livre(ISBN, id_membre)	5
2. rendre_livre(ISBN, id_membre)	5
3. stats_genre() / stats_auteurs() / stats_emprunts()	6
6. Implémentation des fonctionnalités	6
7. Perspectives d'amélioration	8
8. Conclusion.....	8

1. Introduction

Ce projet consiste à développer un **système informatique simple et intuitif** permettant de gérer les opérations essentielles d'une bibliothèque, depuis l'ajout de livres et l'inscription de membres jusqu'au suivi des emprunts, retours et statistiques.

Ce projet a été réalisé en utilisant le langage **Python**, avec le module **Tkinter** pour l'interface graphique, **JSON** pour la gestion des données structurées, et **Matplotlib** pour la visualisation statistique. L'objectif principal est de fournir une solution légère, pédagogique et modulable, adaptée à un environnement scolaire ou personnel.

2. Analyse des besoins

Les besoins identifiés se répartissent comme suit :

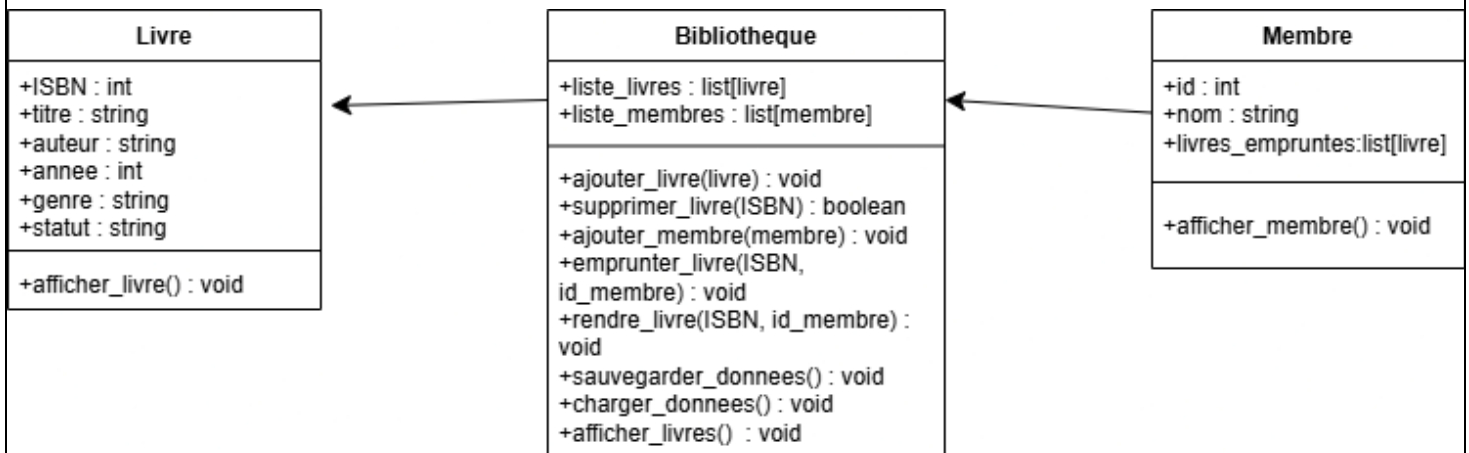
1. Besoins fonctionnels

- ☐ Ajouter, modifier et supprimer des livres et des membres.
- ☐ Enregistrer les opérations de prêt et de retour.
- ☐ Visualiser l'état de disponibilité des ouvrages.
- ☐ Afficher des statistiques sur les livres empruntés, les genres les plus consultés, et les auteurs les plus lus.

2. Besoins Techniques

- ☐ Stockage local des données via des fichiers JSON et CSV pour la simplicité et la portabilité.
- ☐ Interface graphique intuitive grâce à **Tkinter**, accessible à tout utilisateur sans connaissances techniques.
- ☐ Organisation modulaire du code en plusieurs classes (Livre, Membre, Bibliotheque, etc.) pour faciliter la maintenance.
- ☐ Affichage de statistiques graphiques à l'aide de **Matplotlib**

3. Diagramme de classes UML



4. Modèle de données

```
Src/
├── data/
│   ├── historique.csv
│   ├── livres.json
│   ├── log.csv
│   └── membres.json
├── bibliotheque.py
├── exceptions.py
├── livre.py
├── main.py
├── membre.py
└── visualisations.py
```

5. Explication des algorithmes clés

1. `emprunter_livre(ISBN, id_membre)`

- Recherche d'un membre et du livre.
- Vérification que le livre est disponible.
- Ajout du livre à la liste `livres_empruntes` du membre.
- Changement du statut du livre à "emprunté".
- Sauvegarde automatique dans l'historique.

2. `rendre_livre(ISBN, id_membre)`

- Recherche du membre et du livre dans ses emprunts.

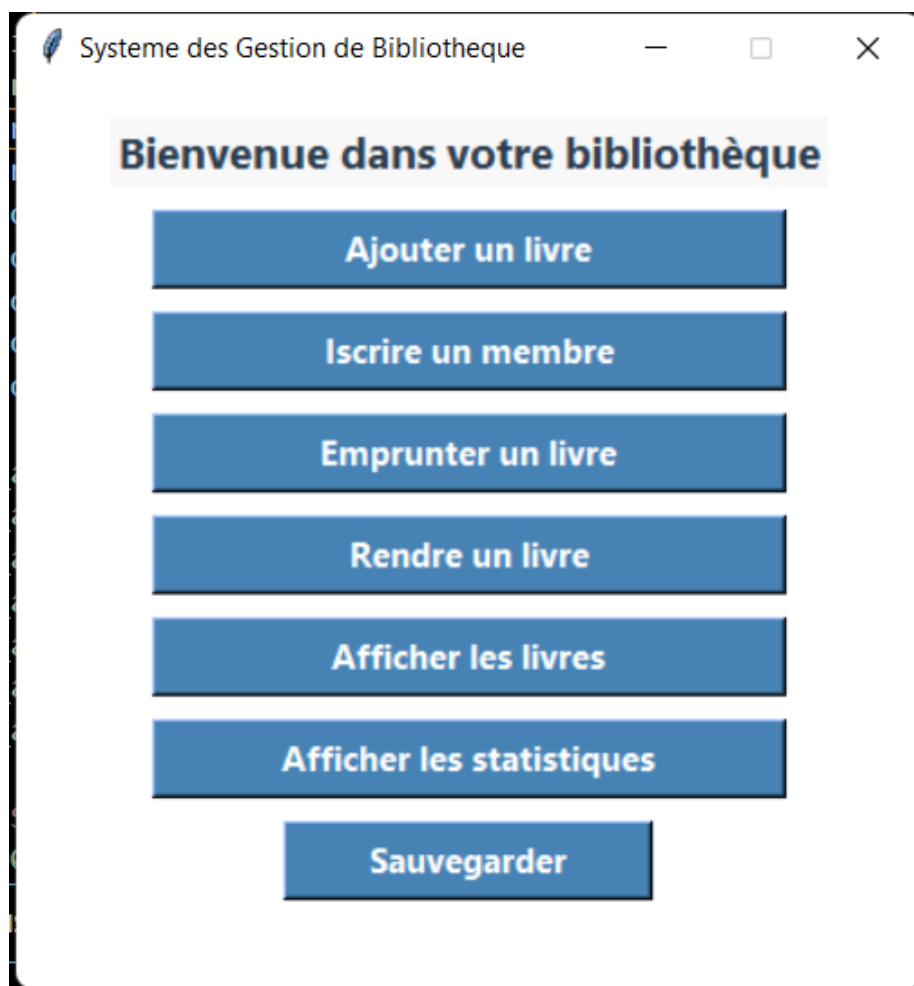
- Vérification que le livre figure bien dans livres_emprunts.
- Suppression de la liste.
- Remise du statut à “disponible”.

3. stats_genre() / stats_auteurs() / stats_emprunts()

- Récupération des données depuis les fichiers JSON ou CSV.
- Comptage via dictionnaires Python.
- Génération de graphiques via matplotlib.pyplot (barres, camemberts, courbes).

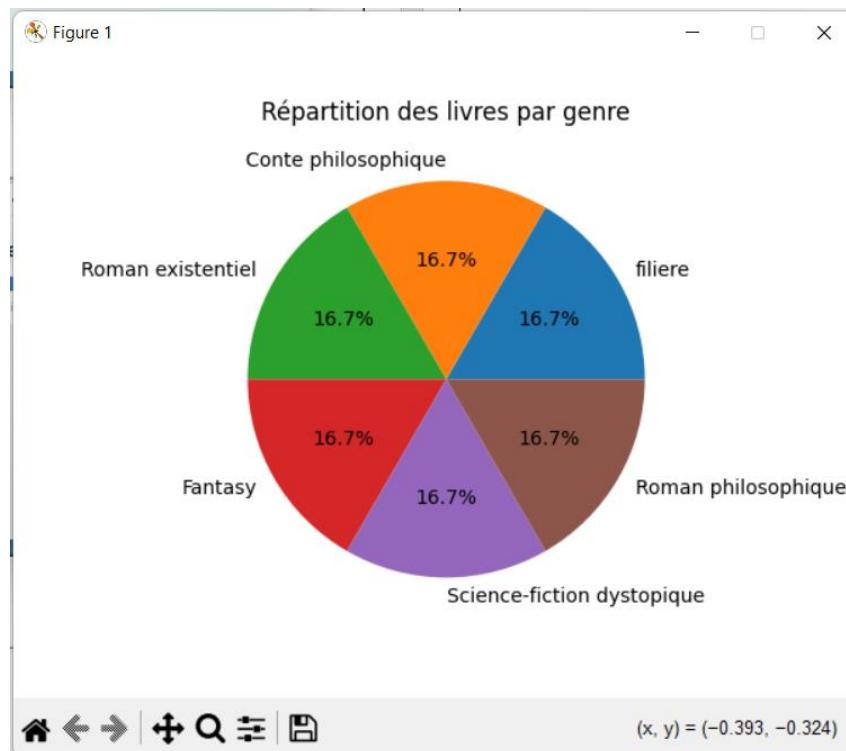
6. Implémentation des fonctionnalités

L'interface d'accueil se présente sous la forme d'une fenêtre principale conviviale, comprenant **sept boutons fonctionnels** permettant d'accéder aux différentes opérations du système, telles que l'ajout de livres, l'inscription de membres, la gestion des emprunts et des retours, l'affichage des données et la sauvegarde.





L'interface "Statistiques" offre à l'utilisateur la possibilité de choisir librement le type d'analyse qu'il souhaite consulter, qu'il s'agisse de la répartition des livres par genre, des auteurs les plus empruntés ou de l'évolution des emprunts dans le temps.



7. Perspectives d'amélioration

- Ajout d'un système de recherche dynamique (par titre, auteur, ID membre).
- Création d'un tableau interactif pour afficher les livres et les membres avec tri et filtrage.
- Exportation des statistiques ou de l'historique au format PDF ou Excel.
- Ajout d'un système d'authentification pour les administrateurs (mot de passe)
- Migration vers une base de données SQLite ou PostgreSQL pour une meilleure robustesse.

8. Conclusion

Ce projet m'a permis de renforcer mes compétences en Python, en interfaces graphiques avec Tkinter, et en gestion de données. J'ai appris à structurer un code clair, à anticiper les erreurs, et à concevoir une interface simple et efficace.

Au-delà de l'aspect technique, cette expérience m'a apporté rigueur, autonomie et confiance dans la réalisation d'un projet concret. Elle ouvre aussi la voie à de futures améliorations et extensions.