RAPPORT DE PROJET

Mise sur pied d’une bibliothèque Universitaire



# INTRODUCTION

## Contexte

La bibliothèque par essence, est le centre de concentration du savoir, elle conserve une trace écrite des pensées et des prouesses humaines. Dans notre université, nous avons un système de gestion de bibliothèque encore archaïque qui, plutôt que de conserver le savoir, le pousse bien au contraire à se perdre. Nous constatons des lourdeurs dans les processus de traitement et de gestion de l’ensemble des opérations au sein de la bibliothèque. Le service est très lent, la gestion des stocks se fait encore au décompte et il est difficile de suivre l’historique des emprunts et des réservations pouvant entraîner des pertes de livres.

Tout ceci rend le processus de travail archaïque et inefficace face à la demande sans cesse croissante au sein de la bibliothèque due aux grands effectifs de l’université et qui sans doute augmenteront les rentrées prochaines considérablement.

## Objectifs

Notre initiative à travers ce projet que nous lançons vise la modernisation de notre bibliothèque ceci à travers un **“SYSTÈME DE GESTION EFFICACE ET DURABLE DE LA BIBLIOTHÈQUE”** avec une base de données que nous alimenterons au fil du temps en fonction du nombre d’abonnés qui viendront s'enrôler en salle de lecture.

# 

# 

# 

# MÉTHODOLOGIE

Pour parvenir à un système qui répond pleinement aux exigences susmentionnées, nous avons choisi de procéder méthodiquement.

## Choix des entités et définition des associations

Suivant les exigences de notre système qui sont entre autres :

1. **La gestion des livres :**

* Ajout, modification et suppression de livres ;
* Recherche de livres par titre, auteur, genre, catégorie ;
* Gestion des stocks et des exemplaires disponibles ;
* Notation des livres.

1. **La gestion des Utilisateurs :**

* Inscription des étudiants et des membres du personnel ;
* Authentification et gestion du compte ;
* Gestion des droits d’accès ;
* Gestion des informations des utilisateurs (nom, prénom, numéro d'étudiant, contacts…) ;
* Gestion des emprunts et des réservations par utilisateur.

1. **La gestion des prêts et retours :**

* Enregistrement des prêts de livres ;
* Suivi des dates de retour prévues et des retards ;
* Enregistrement des retours et mise à jour des stocks.

Nous avons répertorié 4 entités et 2 associations

1. **Entités**

* **Livres** : id\_livre, titre, auteur, annee\_publication, genre, nombre\_exemplaires
* **Étudiants** : id\_etudiant, nom, prenom, date\_naissance, email, telephone, date\_inscription
* **Emprunts** : id\_emprunt, id\_etudiant, id\_livre, date\_emprunt, date\_retour
* **Genres** : id\_genre, nom\_genre

**Associations**

| **Association** | **Cardinalité** | **Explication** |
| --- | --- | --- |
| emprunter | (0.N : 0.N) | 1 étudiant peut emprunter 0 ou plusieurs livres de même qu’un livre peut être emprunté par 0 ou plusieurs plusieurs étudiants |
| Appartenir | (0.N : 1.1) | 1 livre ne peut appartenir qu’à un et un seul genre par contre un genre peut contenir 0 ou plusieurs livres |

## Création de la Base de données et des tables

Suivant le rendu de notre MCD présenté dans la documentation technique, nous avons créé une base de données **“Bibliothèque”** ; des tables à base du langage SQL et nous leur avons attribué les caractéristiques suivantes.

1. **Table Livres**

| **Caractéristiques** | **Nom du champ** | **Type** | **NULL?** |
| --- | --- | --- | --- |
| Identifiant du livre | id\_livre | AUTO\_INCREMENT | Non |
| titre du livre | titre | VARCHAR (50) | Non |
| auteur du livre | auteur | VARCHAR (50) | Non |
| année de publication du livre | année\_public | INT | Non |
| nombre d’exemplaire du llivre | nbr\_exemplaire | INT (25) | Non |
| identifiant du genre du livre | id\_genre\_genres | INT (25) | Non |

1. **Table genre**

| **Caractéristiques** | **Nom du champ** | **Type** | **NULL?** |
| --- | --- | --- | --- |
| Identifiant du genre | id\_genre | AUTO\_INCREMENT | Non |
| Nom du genre | nom\_genre | VARCHAR (50) | Non |

1. **Table étudiants**

| **Caractéristiques** | **Nom du champ** | **Type** | **NULL?** |
| --- | --- | --- | --- |
| Identifiant de l’étudiant | id\_etudiant | AUTO\_INCREMENT | Non |
| nom de l’étudiant | nom | VARCHAR (25) | Non |
| prénom de l’étudiant | prénom | VARCHAR (25) | Non |
| date de naissance de l’étudiant | date\_naiss | DATE | Non |
| adresse email de l’étudiant | email | VARCHAR (50) | Non |
| téléphone de l’étudiant | tel | VARCHAR (50) | Non |
| date d’inscription de l’étudiant | date\_inscription | DATE | Non |

1. **Table emprunts**

| **Caractéristiques** | **Nom du champ** | **Type** | **NULL?** |
| --- | --- | --- | --- |
| Date de l’emprunt | date\_empt | DATE | Non |
| Date de retour de l’emprunt | date\_retour | DATE | Non |
| état du retour du livre | retour | BOLL | Non |
| identifiant du livre | id\_livre\_livres | AUTO\_INCREMENT | Non |
| identifiant de l’emprunteur | id\_etudiant\_étudiants | AUTO\_INCREMENT | Non |

## Ajout des données

La création de nos différentes tables étant faite, nous avons pour une opérationnalisation de notre base de données inséré une série de données pour tester son bon fonctionnement.

Chacune de ces données remplit pleinement les caractéristiques que nous avons définies dans les tables ci-dessus. Nous avons respectivement dans notre bibliothèque à l’heure actuelle :

* 06 Genres
* 10 étudiants inscrits
* 19 livres
* 15 emprunts déjà enregistrés dont 7 retours de livres effectifs et 8 en cours.

## Création des requêtes

Notre système en plus de gérer les emprunts aide le bibliothécaire à émettre des rapports et à faire un bilan de gestion ceci à travers des requêtes qu’il envoie au système.

Nous avons prévu des requêtes pour :

* **Trouver les étudiants ayant emprunté un certain livre**
* **Compter le nombre total d’emprunts par livre**
* **Lister les livres d’un certain genre**

## Définition des fonctions et procédures de notre système

Une fonction est un code qui effectue une série d'instructions bien précises, et renvoie le résultat de ces instructions.

En matière de fonctions et de procédures, notre système prend en compte :

* Une procédure d’ajout d’un emprunt
* Une fonction de vérification de la disponibilité d’un livre

## Mise en place des vues et triggers

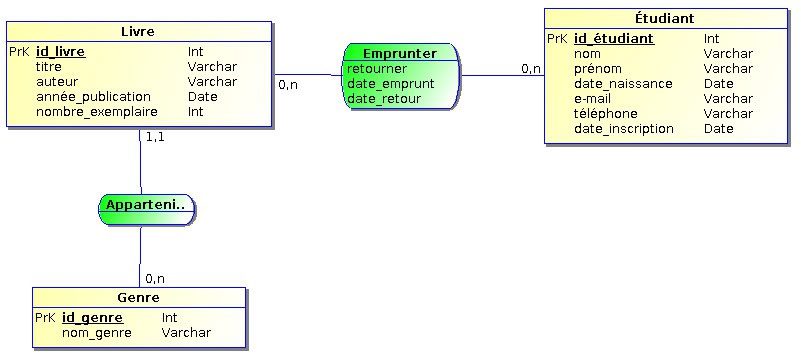
Les vues sont des objets de la base de données, constitués d'un nom, et d'une requête de sélection. Une fois qu'une vue est définie, on peut l'utiliser comme on le ferait avec une table ; table qui serait constituée des données sélectionnées par la requête définissant la vue. Pour notre projet nous avons défini une vue pour **afficher les détails d’un emprunt.**

Les triggers (ou déclencheurs) sont des objets de la base de données. Attachés à une table, ils vont déclencher l'exécution d'une instruction, ou d'un bloc d'instructions, lorsqu'une, ou plusieurs lignes sont insérées, supprimées ou modifiées dans la table à laquelle ils sont attachés. Pour notre projet nous avons définit un trigger pour **actualiser l’état du stock des livres quand un emprunt est achevé (pris et rendu)**

# Résultats

Dans cette section vous trouverez pour chaque étape de notre projet des captures d'écran montrant l’ensemble de nos actions sous forme de scripts SQL et le rendu lors de l’implémentation dans MySQL

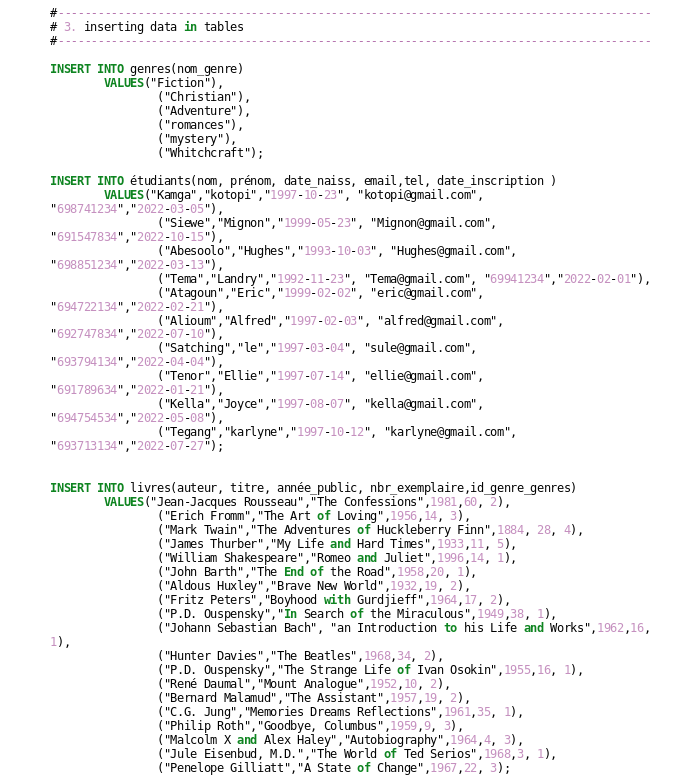
## Choix des entités et définition des associations



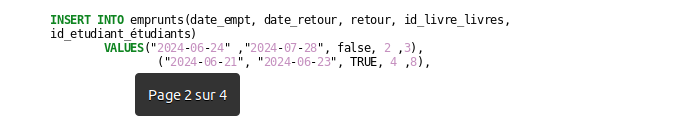
## Création de la base de données et des tables

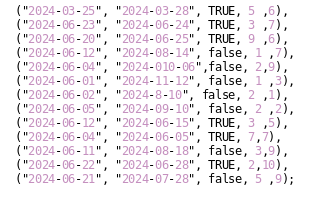
## 

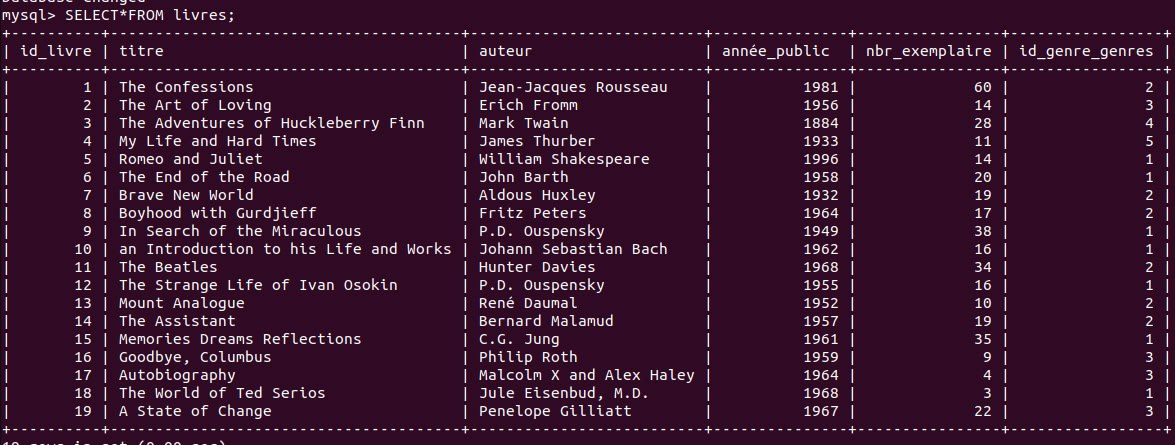
## Ajout des données



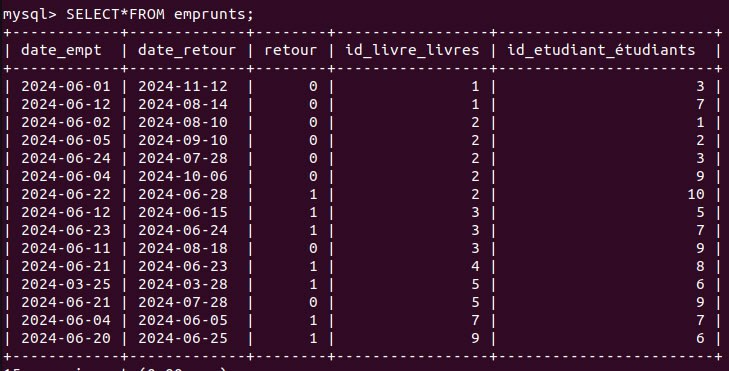
## 





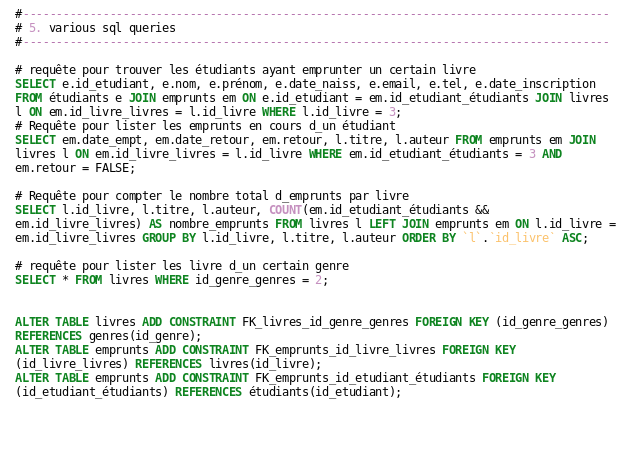


## 



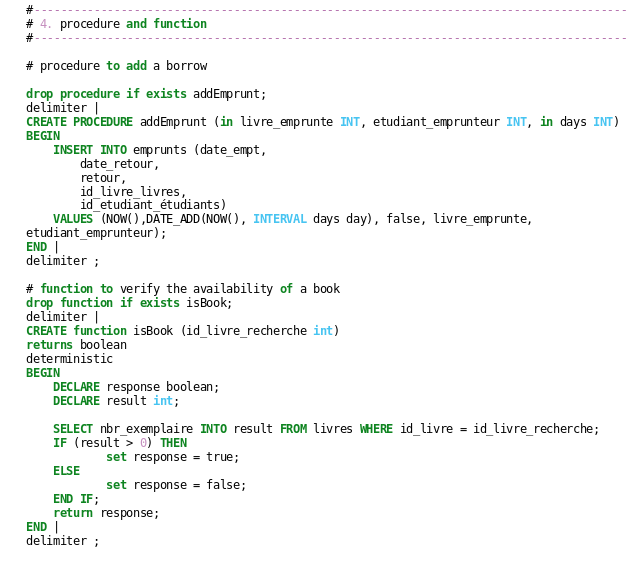
## 

## Création des requêtes



## 

## Définition des fonctions et procédures de notre système



## Mise en place des vues et triggers

# 

# 

# 

# 

# 

# 

# CONCLUSION

À travers notre base de données bibliothèque que nous avons créé, nous pouvons dire que la modernisation de notre bibliothèque va bon train. Nous avons pu dématérialiser la grande majorité des procédures au sein de la bibliothèque.

En utilisant notre système, nous pouvons avoir une traçabilité des actions menées au sein de la bibliothèque et nous pouvons gérer plusieurs générations d’apprenants.