

COMPTE RENDU  
PROJET INFORMATIQUE

---

---

MÉDIATHÈQUE

---

---

*Auteurs:*

*Yann Colin*

*Grzegorz Maj*

---

10 janvier 2016

# Table des matières

<b>1</b>	<b>Introduction</b>	<b>2</b>
<b>2</b>	<b>La conception</b>	<b>2</b>
2.1	Diagramme de classes . . . . .	2
2.2	Diagramme de cas d'utilisation . . . . .	3
<b>3</b>	<b>Implémentation</b>	<b>3</b>
<b>4</b>	<b>Les tests</b>	<b>3</b>
<b>5</b>	<b>Façon de collaborer</b>	<b>4</b>
5.1	Partage du travail . . . . .	4
5.2	Logiciels utilisés . . . . .	4
5.3	Les problèmes . . . . .	5
<b>6</b>	<b>Conclusion</b>	<b>5</b>

L'objectif de ce projet est de créer un programme afin informatiser le système d'information d'une médiathèque. Celui-ci offrira aux adhérents des nouveaux services comme par exemple de réserver, d'emprunter via le logiciel des ressources. Nous avons dû utiliser la programmation orientée objet avec comme langage le Java.

## 2 La conception

## 2.1 Diagramme de classes

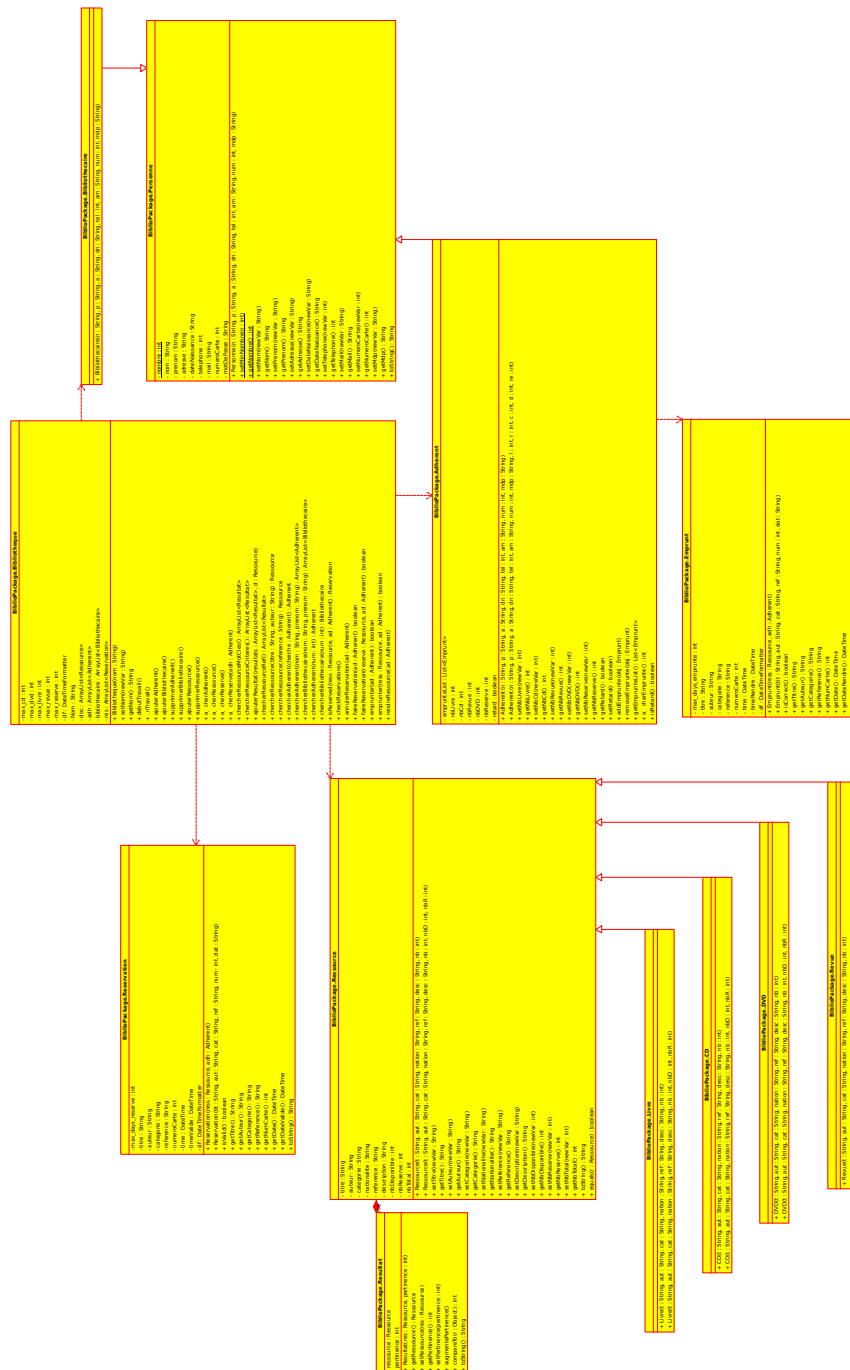


FIGURE 1 – Diagramme de classes

## 2.2 Diagramme de cas d'utilisation

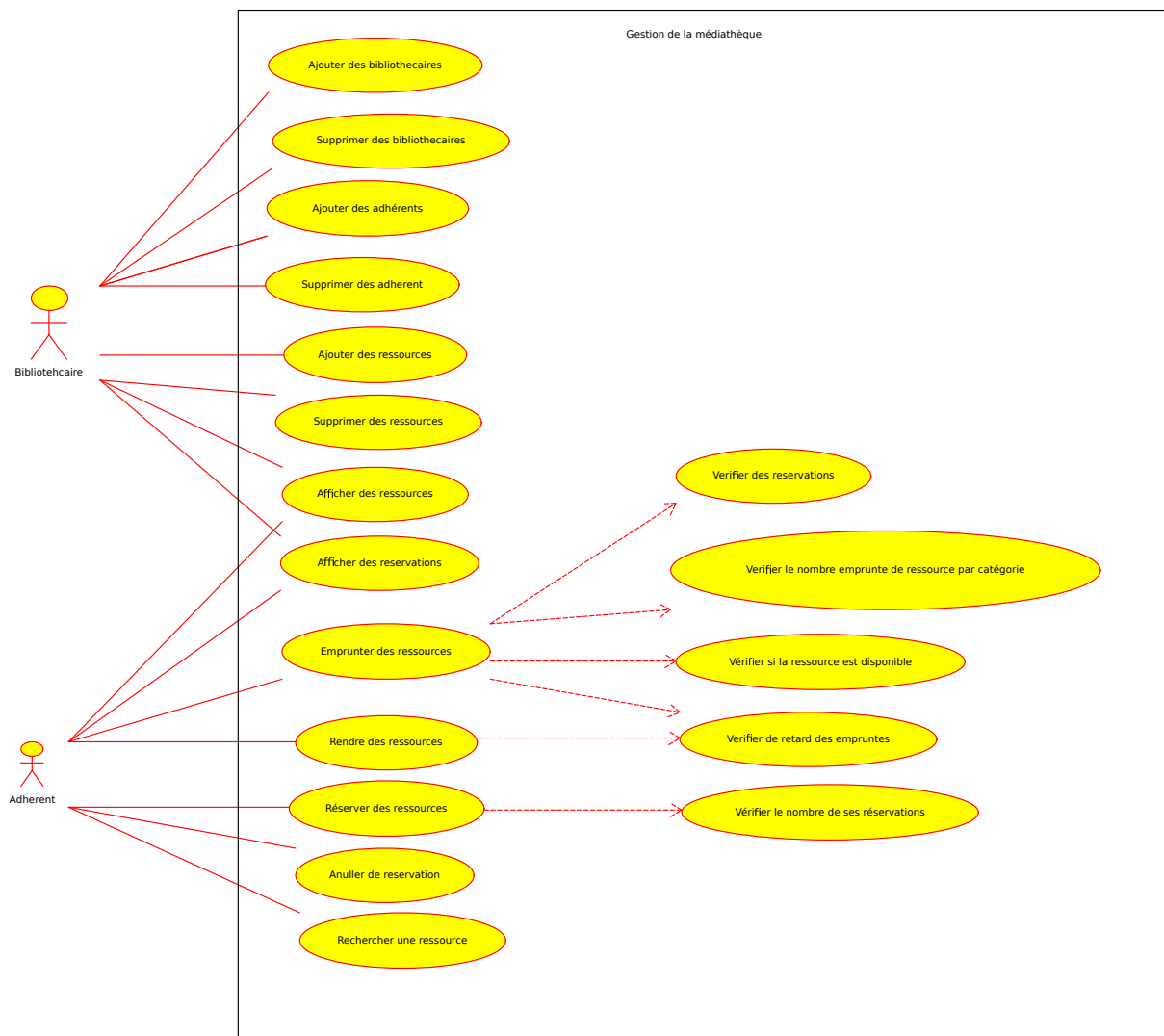


FIGURE 2 – Diagramme de cas d'utilisation.

## 3 Implémentation

Le logiciel est implémenté dans la langage Java. Pour lire les commandes de l'utilisateur on a utilisé la classe Lire.

Le projet de la médiathèque nécessite d'utiliser une base de données ou d'enregistrer les données dans des fichiers. On a considéré, comme option, l'utilisation du format XML ou du JSON. On a choisi le JSON, parce que il est plus léger et plus simple que le XML. Comme parser, on a choisi le JSON.simple.

Pour faire les emprunts et les réservations, on a besoin d'utiliser un certain format de la date. On a utilisé le JodaTime.

## 4 Les tests

Pendant les tests on a dû vérifier :

1. ajouter adhérent
2. supprimer adhérent
3. ajouter bibliothécaire
4. supprimer bibliothécaire

5. afficher adhérent
6. afficher ressource
7. ajouter ressource
8. supprimer ressource
9. emprunter
  - limites d'emprunt
  - durée d'emprunt
10. rendre
11. réservation
  - limites de la réservation
  - durée de la réservation
  - annuler une réservation
12. la recherche de ressources

Ci-joint, se trouve un exemple d'un test que nous avons réalisé.

Pour tester la date limite d'emprunt, et la limite de réservation, nous avons changé les durées limites à -1 jour. Ainsi nous avons pu voir tout de suite l'effet, de ne pas avoir rendu son document à temps, ou de ne pas aller chercher son document réservé.

## 5 Façon de collaborer

### 5.1 Partage du travail

Yann :

- diagramme de classes,
- diagramme de cas d'utilisation,
- classe Ressource + classes filles,
- classe Bibliothèque,
  - toutes les méthodes de recherche,
  - gestion des comptes d'adhérents et de bibliothécaires,
  - menu pour utilisateur,
- compte rendu.

Grzegorz :

- diagramme de classes,
- diagramme de cas d'utilisation,
- classe Personne + classes filles,
- classe Emprunt et classe Réservation,
- classe Bibliothèque,
  - gestion des réservations,
  - gestion des emprunts,
  - enregistre des données et lire de fichiers (JSON),
- compte rendu.

Nous avons aussi chacun lit la partie de code, créer par l'autre, et corrigé si besoin était.

### 5.2 Logiciels utilisés

Nous avons utilisé Netbeans pour implémenter le programme, car c'est un logiciel très complet par rapport à Geany.

Pour les diagrammes de classes et d'utilisation, nous avons utilisé Umbrello qui est un logiciel très simple à utiliser.

Pendant le travail, on a gardé tous les fichiers sur un serveur GitHub avec utilisation du système Git. Cela nous a permis de travailler en même temps, sur un même fichier sans avoir de conflits entre les fichiers. Il nous permet aussi de créer différentes branches, pour différentes options et ensuite d'en choisir la meilleur. Nous avons utilisé le serveur GitHub car il est gratuit, mais en contrepartie tout le monde peut voir notre projet.

Tout le projet est disponible sous l'adresse <https://github.com/grzegorzmaj/projetBibliotheque>.

### 5.3 Les problèmes

Le seul problèmes rencontré, fût de trouver une classe de date simple et stable. Il y avait un problème avec la gestion de date simple et stable. Le soutien pour la date dans le langage Java n'est pas bon. La solution a était d'utilisé le JodaTime qui est très connu et utilisé dans le langage Java.

## 6 Conclusion

Le programme rendu répond a toute les attentes du projet, et le logiciel fonctionne sans dysfonctionnement comme vous pouvez le voir dans le test ci-joint ou en essayant le programme.

Ce projet fût intéressant et il nous a permis de mobiliser les notions vu en cours d'informatique, car nous avons utilisé la programmation orientée objet avec des diagrammes de classes et de cas d'utilisation.