

Philippe Bertrand

Programmation et sécurité

Documentation du projet synthèse

Travail présenté à

Hasna Hocini

dans le cadre du cours

420-3P1 Développement d’applications Windows

Groupe 00001

Cégep de l’Outaouais

Campus Gabriel-Roy

Le 18 décembre 2022

**TABLE DES MATIÈRES**

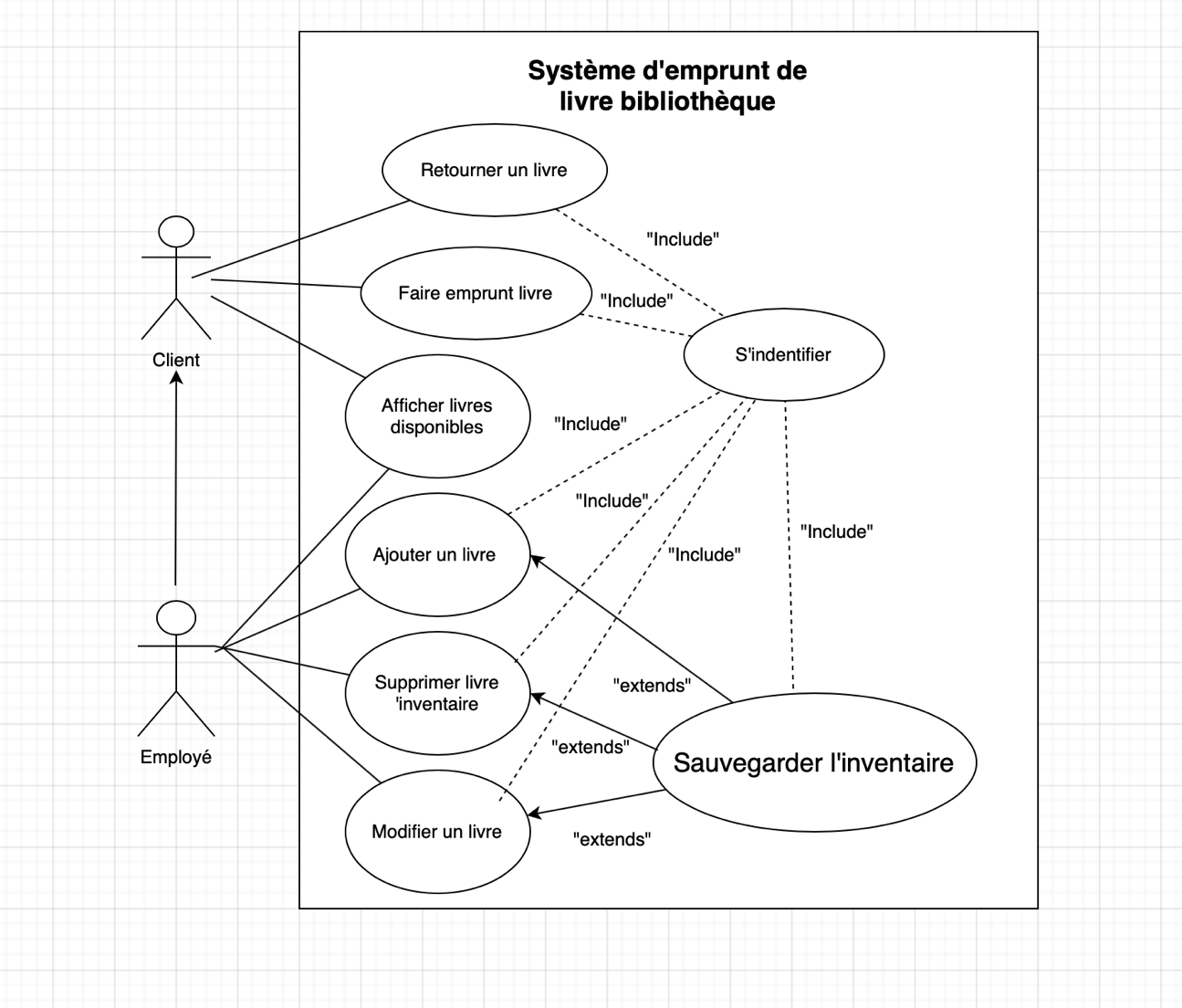
Table des matières

**Aucune entrée de table des matières n'a été trouvée.**

## Description du projet :

L’application choisi serait un système d’emprunt de livre dans une bibliothèque avec des clients et employés. D’un côté on a des clients qui peuvent se créer des comptes afin de pouvoir emprunter des livres et les retourner. De l’autre côté nous avons les employés qui s’occupe de la gestion des livres. Ils peuvent ajouter, modifier, supprimer des livres et ensuite sauvegarder les modifications apportées à l’inventaire. Toutes actions faites par les acteurs demandent une identification afin d’assurer la sécurité du système. Lors d’un emprunt, le livre choisi est retiré de l’inventaire jusqu’au retour de ce dernier par le client. Lorsqu’il est de retour, on re-ajoute le livre dans l’inventaire et il peut être réutiliser à nouveau par d’autres clients.

## Diagramme de cas d’utilisation :



## Description détaillée des cas d’utilisation :

Nom : Emprunt

Acteur : Client

Données d’entrée : Livre choisi par le client

Le cas d’utilisation démarre lorsque le client clique sur le bouton « Emprunter »

### Scénario principal :

1. Le système veut que le client saisi son identifiant et le rentre dans le champ « Client ID »
2. Le client entre son identifiant et le système vérifie si l’identifiant est bon.

* 2a. Scénario d’erreur ou l’identifiant est incorrect et que le client doit retaper un identifiant valide pour pouvoir continuer. Le système lui affiche également un message d’erreur.

1. Le système vérifie si les informations entrées du livre existent réellement pour vérifier si le livre est disponible ou tout simplement qu’il existe.

* 3a. Scénario d’erreur ou le client n’a pas rentré une information valide sur le livre voulu. Le client reçoie alors un message d’erreur qui l’informe de la problématique et recommence l’étape 3.

1. Lorsque l’emprunt est validé, le système enregistre le temps que cela va prendre pour retourner le livre emprunté et envoie un message de confirmation au client qui l’informe que son emprunt a fonctionné.

Nom : Retour

Acteur : Client

Données d’entrée : Livre choisi par le client

Le cas d’utilisation démarre lorsque le client clique sur le bouton « Retourner »

### Scénario principal :

1. Le client entre son identifiant dans le textbox « Client ID » et le système vérifie s’il est bon

* 1a. Scénario d’erreur si l’identifiant n’est pas valide, alors on envoie un message d’erreur au client qui l’informe de la problématique et on recommence l’étape 1.

1. Le client entre les informations du livre à retourner et le système vérifie si le client avait réellement emprunté le livre auparavant.

* 2a. Scénario d’erreur si les informations ne sont pas valides, on affiche un message d’erreur et on demande de recommencer l’étape 2.

1. SI les 2 étapes précédentes sont validées, on retourne le livre dans l’inventaire et on affiche un message au client en lui montrant le temps passé avec le livre.

Nom : Ajouter livre

Acteur : Employé

Données d’entrée : Livre à ajouter

Le cas d’utilisation démarre lorsque l’employé clique sur le bouton « Ajouter un livre »

### Scénario principal :

1. L’employé doit entrer l’identifiant admin afin de pouvoir modifier toutes données dans le système.

* 1a. Scénario d’erreur si l’identifiant n’est pas valide, alors on envoie un message d’erreur à l’employé qui l’informe de la problématique et on recommence l’étape

1. L’employé entre toutes sortes d’informations sur le livre à ajouter comme son nom, l’auteur, la date de sortie, l’identifiant donné et autres.

* 2a. Scénario d’erreur si les informations données ne passent pas les exigences voulues, on affiche un message d’erreur et on efface les textbox ou on détecte une erreur et on recommence l’étape 2.

1. Si toutes les informations sont bonnes et l’identifiant admin est correct, on ajoute le nouveau livre dans la liste d’inventaire de livre afin de pouvoir être emprunter. On affiche également un message de confirmation de l’ajout du livre.

Nom : Modifier livre

Acteur : Employé

Données d’entrée : Livre à modifier

Le cas d’utilisation démarre lorsque l’employé clique sur le bouton « Modifier un livre »

### Scénario principal :

1. L’employé doit entrer l’identifiant admin afin de pouvoir modifier toutes données dans le système.

* 1a. Scénario d’erreur si l’identifiant n’est pas valide, alors on envoie un message d’erreur à l’employé qui l’informe de la problématique et on recommence l’étape

1. L’employé entre toutes sortes d’informations sur le livre à modifier comme son nom, l’auteur, la date de sortie. Il est impossible de modifier l’identifiant du livre afin de garder l’identité unique du livre.

* 2a. Scénario d’erreur si les informations données ne passent pas les exigences voulues, on affiche un message d’erreur et on efface les textbox ou on détecte une erreur et on recommence l’étape 2.

1. Si toutes les informations sont bonnes et l’identifiant admin est correct, on modifie le livre dans la liste d’inventaire de livre afin de pouvoir être emprunter. On affiche également un message de confirmation de la modification du livre.

Nom : Supprimer livre

Acteur : Employé

Données d’entrée : Livre à supprimer

Le cas d’utilisation démarre lorsque l’employé clique sur le bouton « Supprimer un livre »

### Scénario principal :

1. L’employé doit entrer l’identifiant admin afin de pouvoir modifier toutes données dans le système.

* 1a. Scénario d’erreur si l’identifiant n’est pas valide, alors on envoie un message d’erreur à l’employé qui l’informe de la problématique et on recommence l’étape

1. L’employé entre l’identifiant du livre à supprimer et le système si le livre existe.

* 2a. Scénario d’erreur si l’identifiant donné est inexistant, un affiche un message d’erreur pour informer l’employé et on doit recommencer l’étape 2 jusqu’à l’envoi d’une bonne valeur.

1. Si l’identifiant du livre et l’identifiant admin est correct, on supprime livre de la liste d’inventaire de livre afin de ne plus pouvoir l’emprunter. On affiche également un message de confirmation de la suppression du livre.

Nom : Sauvegarder

Acteur : Employé

Données d’entrée : Sauvegarder les données

Le cas d’utilisation démarre lorsque l’employé clique sur le bouton « Sauvegarder »

### Scénario principal :

1. L’employé doit entrer l’identifiant admin afin de pouvoir sauvegarder toutes données dans le système.

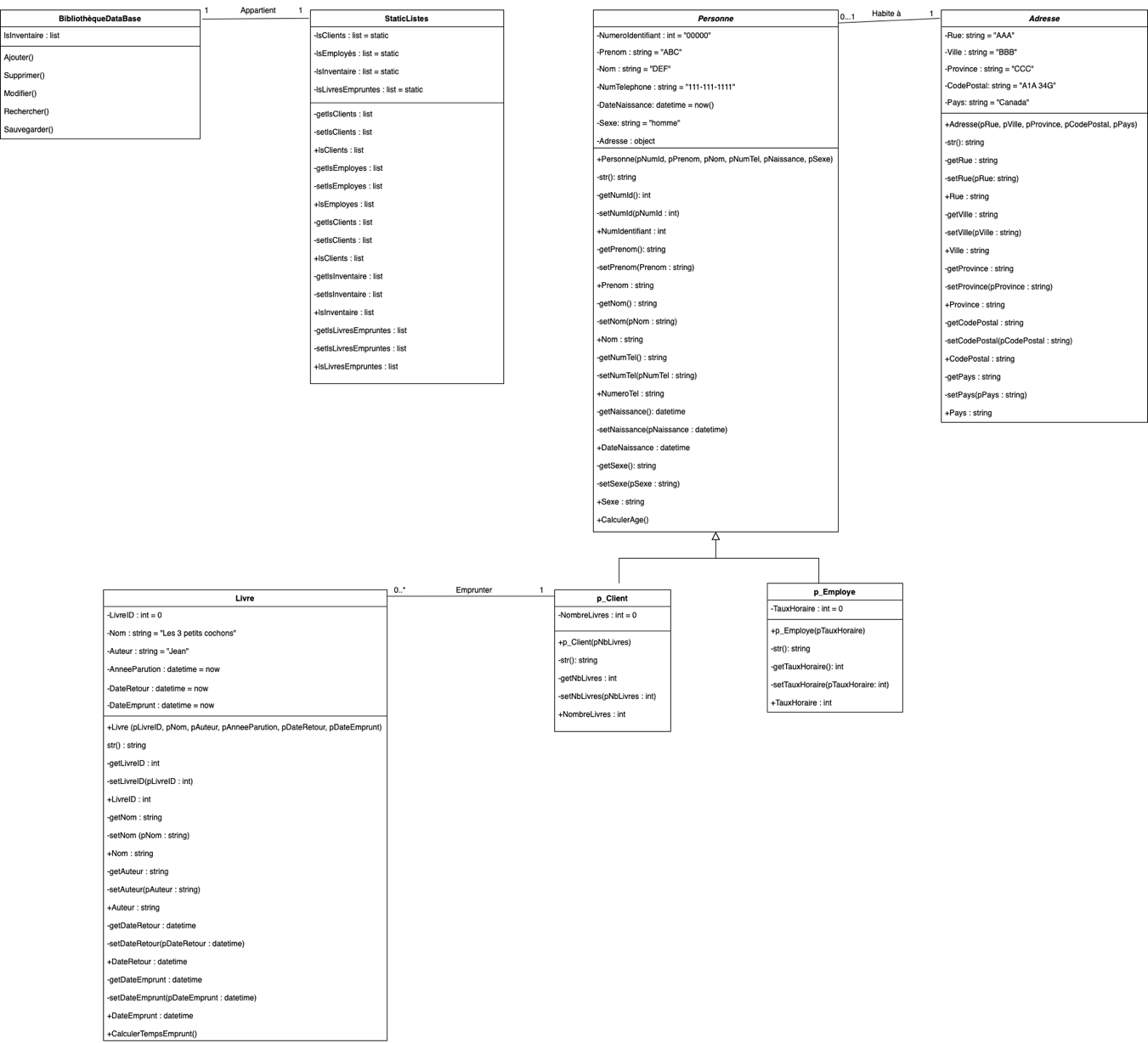
* 1a. Scénario d’erreur si l’identifiant n’est pas valide, alors on envoie un message d’erreur à l’employé qui l’informe de la problématique et on recommence l’étape

1. L’employé entre toutes sortes d’informations sur le livre à ajouter comme son nom, l’auteur, la date de sortie, l’identifiant donné et autres.

* 2a. Scénario d’erreur si les informations données ne passent pas les exigences voulues, on affiche un message d’erreur et on efface les textbox ou on détecte une erreur et on recommence l’étape 2.

1. Si toutes les informations sont bonnes et l’identifiant admin est correct, on sauvegarde les nouvelles données faites durant la session d’utilisation du système. On affiche également un message de confirmation de la sauvegarde de données.

## Diagramme UML des classes :



Lien pour voir l’UML clairement :

## Difficultés rencontrées et solutions :

* Non-fonctionnement de certains regex pour la validation de donnés :

Utiliser un site d’expressions régulières pour régler les problèmes rencontrés.

* Problème lors du chargement de certains formulaires avec un dataGridView

Demander assistance au professeure afin d’éviter de rester bloquer trop longtemps.

* Erreur exception lors de la suppression d’une ligne dans un datagridView :

Il faut sauvegarder la base de données après avoir ajouté où modifier une ligne du dataGridView afin que le programme détecte une modification dans la table en question.

* Problème de chargement d’un formulaire avec un dataGridView :

Le nom de la table donnée n’était pas le bon dans le load du formulaire.

## Journal du temps de travail :

|  |  |  |  |
| --- | --- | --- | --- |
| Date | Nombres d’heures | Tâche | Cumulatif |
| 1 dec | 1 | Recherche sur le sujet | 1 |
| 1 dec | 1 | Création du diagramme des cas d’utilisations | 1 |
| 1 dec | 2,5 | Description détaillée des cas d’utilisations | 2,5 |
| 3 dec | 3 | Création du UML des classes | 3 |
| 5 dec | 0,5 | Correction UML | 0,5 |
| 10 dec | 5 | Création de tous les formulaires du projet et de relier les pages entre elles avec les tools | 5 |
| 11 dec | 6 | Création de toutes les classes du projet | 6 |
| 12 dec | 5 | Début de la programmation des formulaires | 5 |
| 13 dec | 0,5 | Création de la base de données | 0,5 |
| 15 dec | 7 | Suite de la programmation du programme principal | 7 |
| 17 dec | 8 | Suite de la programmation du programme principal | 8 |
| 18 dec | 5 | Finaliser le projet | 5 |
| 18 dec | 1 | Documentation du code | 1 |
| 18 dec | 0,5 | Installation de l’application avec setup.exe | 0,5 |
| 18 dec | 3 | Faire le rapport de projet | 3 |

### Total : 49 heures

## Lien du versionnage du projet :

<https://github.com/PhilippeBertrand1/ProjetFinal_PhilippeB>

## Bibliographie :

Exercice électeur : Hasna Hocini

Laboratoire Ado.Net mode déconnecté : Hasna Hocini

Générateur de regex : <https://regex-generator.olafneumann.org/?sampleText=2020-03-12T13%3A34%3A56.123Z%20INFO%20%20%5Borg.example.Class%5D%3A%20This%20is%20a%20%23simple%20%23logline%20containing%20a%20%27value%27.&flags=i>

Projet Intra : Philippe Bertrand