

# Gestion Documentaire

## Gestion des emprunts

Vous avez été sollicité pour réaliser l'analyse et la conception d'un système d'information pour la gestion d'une bibliothèque. Vous avez ci-après la transcription de l'interview de Melba Varde avec Hansy Clopédie, responsable de cette bibliothèque.

*MV – « Bonjour, j'appartiens à la société CPO qui est mandatée pour réaliser une analyse de vos problèmes de gestion d'ouvrages et vous proposer un système de gestion de prêts. »*

*HC – « C'est cela : comme vous le voyez, nous avons beaucoup d'ouvrages à gérer, et il en arrive tous les jours ! »*

*MV – « Justement, quelles sont les tâches principales que vous devez effectuer ? »*

*HC – « Quand des ouvrages nous parviennent comme ceux-ci, nous les inventorions et les identifions à l'aide d'une petite fiche comme cela. »*

*MV – « Et qu'indiquez-vous sur cette fiche ? »*

*HC – « Nous y reportons le titre de l'ouvrage, le nom de l'éditeur, la date de parution, le ou les noms des auteurs, le numéro ISBN, ainsi que le public auquel l'ouvrage s'adresse (enfant, ado ou adulte). »*

*MV – « Que signifie le numéro ISBN ? »*

*HC – « Les ouvrages sont identifiés de manière unique par un code international : le numéro ISBN (International Serial Book Number). »*

*MV – « Avez vous plusieurs exemplaires d'un même ouvrage ? »*

*HC – « Oui. Chaque exemplaire est identifié par le numéro ISBN de son ouvrage plus un numéro attribué séquentiellement. Nous conservons également la date de réception de l'exemplaire en bibliothèque ainsi que le nombre d'exemplaires par ouvrage gérés en bibliothèque. »*

*MV – « Tout exemplaire est-il empruntable ? »*

*HC – « Non, pour certains ouvrages, nous gardons des exemplaires consultables uniquement en bibliothèque. »*

*MV – « Un exemplaire non empruntable (resp. empruntable) peut-il devenir empruntable (resp. non empruntable) ? »*

*HC – « Non, la qualité d'être ou non empruntable est immuable »*

*MV – « Comment se passe l'emprunt en pratique ? »*

*HC – « C'est relativement simple : en fait pour les exemplaires empruntables, nous réalisons deux fiches. L'une est conservée dans notre fichier au comptoir, l'autre est placée dans l'ouvrage lui-même dans une pochette plastique à l'intérieur du livre. Lorsqu'un lecteur veut emprunter un exemplaire, il nous remet sa carte de lecteur et la fiche de l'ouvrage. Nous vérifions que le public visé pour l'ouvrage est conforme à l'âge du lecteur (pour un enfant, l'âge est inférieur ou égal à 10 ans, l'âge d'un adolescent est compris entre 10 et 16 ans, un adulte a un âge strictement supérieur à 16 ans). Nous indiquons sur la fiche le nom, le prénom, le numéro du lecteur, les dates d'emprunt et de retour (un exemplaire est empruntable pendant une durée fixe de 8 jours). Nous plaçons alors cette fiche dans le fichier des emprunts. Pour savoir si un exemplaire est emprunté, il nous suffit de vérifier dans le bon tiroir du fichier des emprunts si la fiche d'emprunt est présente. Généralement nous relançons par courrier nos lecteurs après 15 jours de retard (/ date de retour). »*

*MV – « Combien d'exemplaires chaque lecteur peut-il emprunter ? »*

*HC – « Au maximum cinq »*

MV – « Quelles informations souhaitez-vous conserver pour chaque lecteur ? »

HC – « Très peu : le numéro de lecteur, le nom, le prénom, la date de naissance, l'adresse et un mail ».

## Objectif

A la suite de cet entretien, la société CPO décide de faire appel à une équipe d'étudiants du département informatique de l'IUT2 pour développer cette application. Les fonctions attendues du système (Cas d'Utilisation) sont les suivantes :

- *NouveauLecteur* : ajout d'un nouveau lecteur. Toutes les informations relatives à ce lecteur doivent être insérées dans le système. Le numéro de lecteur est automatiquement attribué par le système.
- *NouvelOuvrage* : ajout d'un nouvel ouvrage. Toutes les informations relatives à cet ouvrage doivent être insérées dans le système. Un ouvrage peut être créé même si la bibliothèque n'a pas encore d'exemplaires de cet ouvrage.
- *NouvelExemplaire* : ajout d'un exemplaire. Toutes les informations relatives à cet exemplaire doivent être introduites dans le système. Un exemplaire doit obligatoirement correspondre à un ouvrage géré par le système. Ce cas d'utilisation peut permettre au choix la création d'un seul exemplaire ou de plusieurs exemplaires du même ouvrage.
- *ConsulterLecteur* : consultation d'un lecteur pour connaître son numéro, son nom, son prénom, son âge, son adresse et son numéro de téléphone.
- *ConsulterOuvrage* : consultation d'un ouvrage pour connaître son titre, son numéro ISBN, les auteurs, le nom de l'éditeur, la date d'édition.
- *ConsulterExemplairesOuvrage* : consultation des exemplaires d'un ouvrage pour afficher le titre de l'ouvrage et son numéro ISBN ainsi que ses exemplaires. Pour chaque exemplaire il s'agit de connaître le numéro de l'exemplaire, et dans le cas où il est emprunté le numéro, nom et prénom du lecteur ainsi que les dates d'emprunt et de retour.
- *EmprunterExemplaire* : Emprunt d'un exemplaire empruntable et disponible, par un lecteur compatible avec l'ouvrage et ayant moins de cinq emprunts en cours. On conservera la date d'emprunt et de retour.
- *RendreExemplaire* : Retour d'un exemplaire par un lecteur. Pour l'instant nous ne gérerons pas les pénalités de retard.
- *ConsulterEmpruntsLecteur* : consultation des emprunts d'un lecteur pour afficher le numéro, nom et prénom du lecteur ainsi que ses emprunts en cours. Pour chaque emprunt il s'agit de connaître le titre de l'ouvrage, le numéro ISBN, le numéro de l'exemplaire, les dates d'emprunt et de retour.
- *RelancerLecteur* : consultation de tous les exemplaires non retournés au bout de 15 jours. Pour chacun d'eux il s'agit de connaître les mêmes informations que dans la fonction *ConsulterEmpruntsLecteur*.

Le développement de ces fonctionnalités sera réalisé lors de 2 itérations. Vous pouvez rajouter quelques fonctionnalités utiles par exemple : l'édition de l'ensemble des lecteurs, des ouvrages, etc.

## Etapes de Développement – 1ère itération

Pour la première itération, les cas d'utilisations suivants ont été choisis : *NouveauLecteur*, *NouvelOuvrage*, *NouvelExemplaire*, *ConsulterLecteur*, *ConsulterOuvrage*, *ConsulterExemplaires-Ouvrage*. **Pour ConsulterExemplaires-Ouvrage, vous ne devez pour l'instant, pas afficher les informations sur un éventuel emprunt en cours.**

### 1. Expression des besoins et Analyse

- Etablir un diagramme de cas d'utilisation et réaliser la description des Cas d'Utilisation, exprimer ci-besoin des pré-conditions (vous pouvez réutiliser et compléter les descriptions de la page précédente)
- Pour chaque Cas d'Utilisation établir des Scénarios en langue naturelle ou des Diagrammes de Séquences de Haut Niveau mettant en évidence les interactions entre les acteurs et le système cible.
- Etablir un Diagramme de Classes faisant apparaître les classes, les associations, les attributs, la classe Application et ses méthodes correspondant aux cas d'utilisation. Exprimer les invariants dans des notes ou dans un document annexé au diagramme de classes.
- Si besoin affiner la description des cas d'utilisation et donc des Diagrammes de Séquences de haut niveau (ou des scénarios) pour prendre en compte les invariants.

### 2. Conception (sous VP pour assurer la cohérence entre les DS et le DC lisez la doc fournie)

Nous faisons le choix d'une architecture séparant l'IHM de la classe applicaton.

- Représenter les DS détaillés.
- Affiner les Diagrammes de Classes en ajoutant la classe IHM, les méthodes dans les classes, en finalisant les associations (mono ou bi-directionnelle, associations qualifiées...)

*Pour vous aider vous trouverez en annexe une proposition du DS détaillé de nouveauLecteur. Ne pas prendre « tel que » ce DS, en discuter dans le groupe et le modifier en fonction de vos conventions de groupe.*

### 3. Implantation

Définir les classes : transformer les associations en tenant compte des multiplicités et des types d'accès nécessaires. Coder en Java les algorithmes des méthodes en s'appuyant sur les DS détaillés et le DC complet.

*Pour vous aider récupérez sous Chamillo un début de projet NetBeans qui peut vous être utile bibliotheque.zip. Dans un premier temps :*

- *Appropriiez-vous le code et rendez le conforme aux conventions prises par le groupe*
- *Modifier le premier cas d'utilisation qui vous est donné : nouveauLecteur afin de le rendre conforme à vos spécifications*

#### **Le dossier à la fin de l'étape 1 / ce dossier ne sera pas relevé /**

- Conventions du groupe projet
- Expression des besoins et Analyse
  - ✓ Diagramme des cas d'utilisation
  - ✓ Descriptions des cas d'utilisation
  - ✓ Diagrammes de séquence de Haut niveau ou scénarios en langue naturelle
  - ✓ Diagramme de classes de niveau analyse et invariants
- Conception
  - ✓ Diagrammes de séquences détaillés
  - ✓ Diagramme de classes détaillé incluant les méthodes

**Pour passer à l'itération 2 vous devez avoir mon feu vert.**

## Etapes de Développement – 2ième itération

Pour la deuxième itération, les cas d'utilisations suivants ont été choisis : EmprunterExemplaire, RendreExemplaire, ConsulterEmpruntsLecteur, RelancerLecteur,

*Révision des autres cas d'utilisation en tenant compte des emprunts (par exemple ConsulterExemplaires-Ouvrage doit être complété, vous pouvez également fusionner les cas d'utilisation ConsulterLecteur et ConsulterEmpruntsLecteur, etc.)*

Dans cette deuxième itération, de nouvelles étapes sont à prendre en compte.

### 1. Expression des besoins et Analyse

- Etendre le diagramme de cas d'utilisation et réaliser la description en langue naturelle des Cas d'Utilisation (fonctions attendues), exprimer éventuellement des pré-conditions
- Pour chaque Cas d'Utilisation établir (ou revoir) les scénarios en langue naturelle ou les Diagrammes de Séquences de Haut Niveau mettant en évidence les interactions entre les acteurs et le système cible.
- Enrichir le Diagramme de Classes par de nouvelles classes, associations, attributs. Enrichir également la classe Application par les nouvelles fonctionnalités. Exprimer si besoin les invariants supplémentaires.
- **Définir les diagrammes d'état et décrire les états des objets évolutifs (bien choisir les objets dont l'évolution est importante dans le système)**
- Si besoin affiner la description des cas d'utilisation et des Diagramme de séquences de Haut niveau en enrichissant les pré-conditions (tirer partie des diagrammes d'états et états)

### 2. Conception

- Produire ou revoir les DS détaillés.
- Affiner les Diagrammes de Classes en ajoutant les méthodes dans les classes, en finalisant les associations (mono ou bi-directionnelle, associations qualifiées...)
- **Affiner les Diagrammes d'états en rajoutant les actions**

### 3. Implantation

- Modifier les squelettes des classes : transformer les associations en tenant compte des multiplicités et des types d'accès nécessaires
- Coder en Java les algorithmes des méthodes en s'appuyant sur les DS détaillés et le DC complet.

#### Le dossier à la fin de l'étape 2

- Introduction
- Conventions du groupe projet
- Expression des besoins et Analyse
  - ✓ Diagramme des cas d'utilisation
  - ✓ Descriptions des cas d'utilisation
  - ✓ Diagrammes de séquence de Haut niveau ou scénario en langue naturelle
  - ✓ Diagramme de classes de niveau analyse (incluant des invariants)
  - ✓ Diagramme d'états + définition des états (de niveau Analyse)
- Conception
  - ✓ Diagrammes de séquences détaillés
  - ✓ Diagramme de classes détaillé incluant les méthodes
  - ✓ Diagramme d'états (de niveau Conception c'est à dire incluant les actions)
- Implantation
  - ✓ Squelette des classes (attributs typés et signature des méthodes)
  - ✓ Code Java (ne mettre que ce qui vous paraît pertinent)
- Conclusion

**Le dossier (.pdf) ainsi que le code de l'application (.zip du projet netbeans) doivent être déposés sous Chamilo (répertoire travaux / projet) avant la démo.**

## Annexe : Proposition d'un DS détaillé de nouveauLecteur

