

I. Introduction

La gestion de magasin de livres (GML) constitue un exemple concret et suffisamment représentatif des projets e-commerce, pour cela nous désirons créer un logiciel permettant la gestion de magasin de livres. L'objectif fondamental du futur logiciel est de permettre aux clients de rechercher des ouvrages par thème, titre, ISBN, etc, de constituer un panier, puis de pouvoir les commander et les payer directement.

Dans ce chapitre nous introduisons l'analyse et la conception de notre système en utilisant le langage UML, par le processus unifié UP. Ce processus permettra de modéliser d'une manière claire et précise la structure et le comportement de notre système.

II. Les exigences du GML

Le magasin devra regrouper toutes les fonctionnalités nécessaires de :

➤ Recherche

La première étape pour le client consiste à trouver le plus rapidement possible un ouvrage recherché dans l'ensemble du catalogue; les références de cet ouvrage pouvant être plus ou moins précises, il faut lui fournir plusieurs méthodes de recherche. Le client doit pouvoir saisir un critère : titre, thème, ISBN, etc, ou même plusieurs critères à la fois. Les résultats de la recherche sont disponibles sur une page particulière, et doivent pouvoir être facilement parcourus et reclassés.

➤ Découverte

Chaque livre recherché est présenté en détail sur sa propre page du livre sélectionné. On y trouvera en particulier le prix et la disponibilité.

➤ Sélection

Dans notre magasin, le client choisit ses articles les uns à la suite des autres, les dépose dans son panier, puis se rend à la caisse pour régler le tout. Lorsque le client est intéressé par un ouvrage, il peut l'enregistrer dans son panier.

Il a ensuite à tout moment la possibilité d'ajouter des ouvrages, d'en supprimer ou encore de modifier les quantités avant de passer la commande.

➤ Commande d'ouvrages

A tout moment, le client peut accéder au formulaire du bon de commande, dans lequel il confirme ses coordonnées et les informations nécessaires au paiement et à la

livraison. Pour garantir la sécurité et la confidentialité des échanges, il est impératif que l'envoi des données soit crypté. Dans le cas où le client le souhaiterait, le système doit être capable de lui imprimer un devis pour commander par fax ou par courrier.

Le client doit pouvoir ensuite suivre ses commandes récentes, voire les modifier avant expédition, de façon sécurisée. [2]

III. Conception du système

Dans cette partie nous allons introduire les concepts UML fondamentaux pour la spécification des exigences et plus précisément nous allons utiliser le processus unifié qui permettra de modéliser d'une manière claire et précise la structure et le comportement du système. Cette méthodologie est illustrée dans la figure II.1

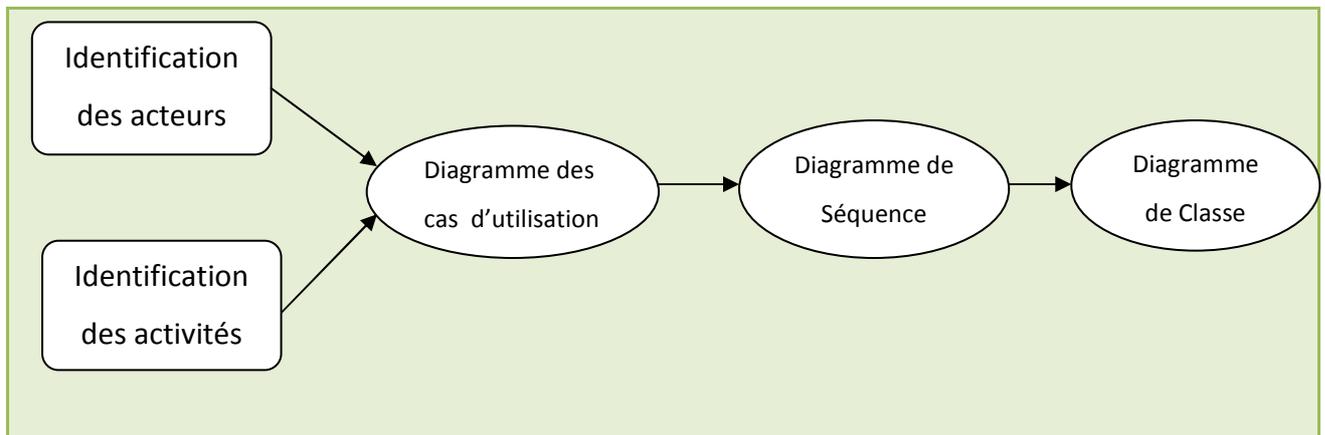


Figure II.1 : Méthodologies de modélisation

III.1. Identification des acteurs

Un acteur représente un rôle joué par une entité (utilisateur humain, dispositif matériel,..), qui interagit directement avec le système étudié. Les acteurs humains pour notre système sont :

- ❖ **Le client** : personne qui visite le magasin pour rechercher des ouvrages et éventuellement passer une commande. Il s'agit de l'acteur le plus important.
- ❖ **L'administrateur** : rôle des employés qui ont comme charge le bon fonctionnement et la maintenance du système.
- ❖ **Le service clients** : rôle des employés qui s'occupent du suivi des commandes des clients.

❖ **Le libraire** : rôle des employés qui sont responsables du contenu rédactionnel du système.

Nous allons aussi prendre en compte les systèmes informatiques connectés au système. A savoir :

-Le système « **nouveautés** » qui sert à alimenter la base avec tous les nouveaux ouvrages.

-Le système « **gestion des stocks** » qui sert à mettre à jour les données concernant le prix et l'état du stock des livres du catalogue.

C'est deux sources seront automatiquement chargées dans la base de données de façon périodique.

III.2. Identification des activités

Pour bien encadrer notre du système nous avons définie un ensemble d'activités pour chaque acteur qui seront développées par la suite :

❖ **Activités du client**

- Recherche des ouvrages.
- Gérer le panier.
- Effectuer une commande.

❖ **Activités d'administrateur**

- Maintenir le système.

❖ **Activités du service clients**

- Suivre les commandes.
- Réalisation des commandes.

❖ **Activités du libraire**

- Maintenir le catalogue, qui fait intervenir les deux systèmes « nouveautés » et « gestion des stocks ».
- Maintenir les informations éditoriales

Le diagramme des cas d'utilisation (figure II.2) illustre les activités du système.

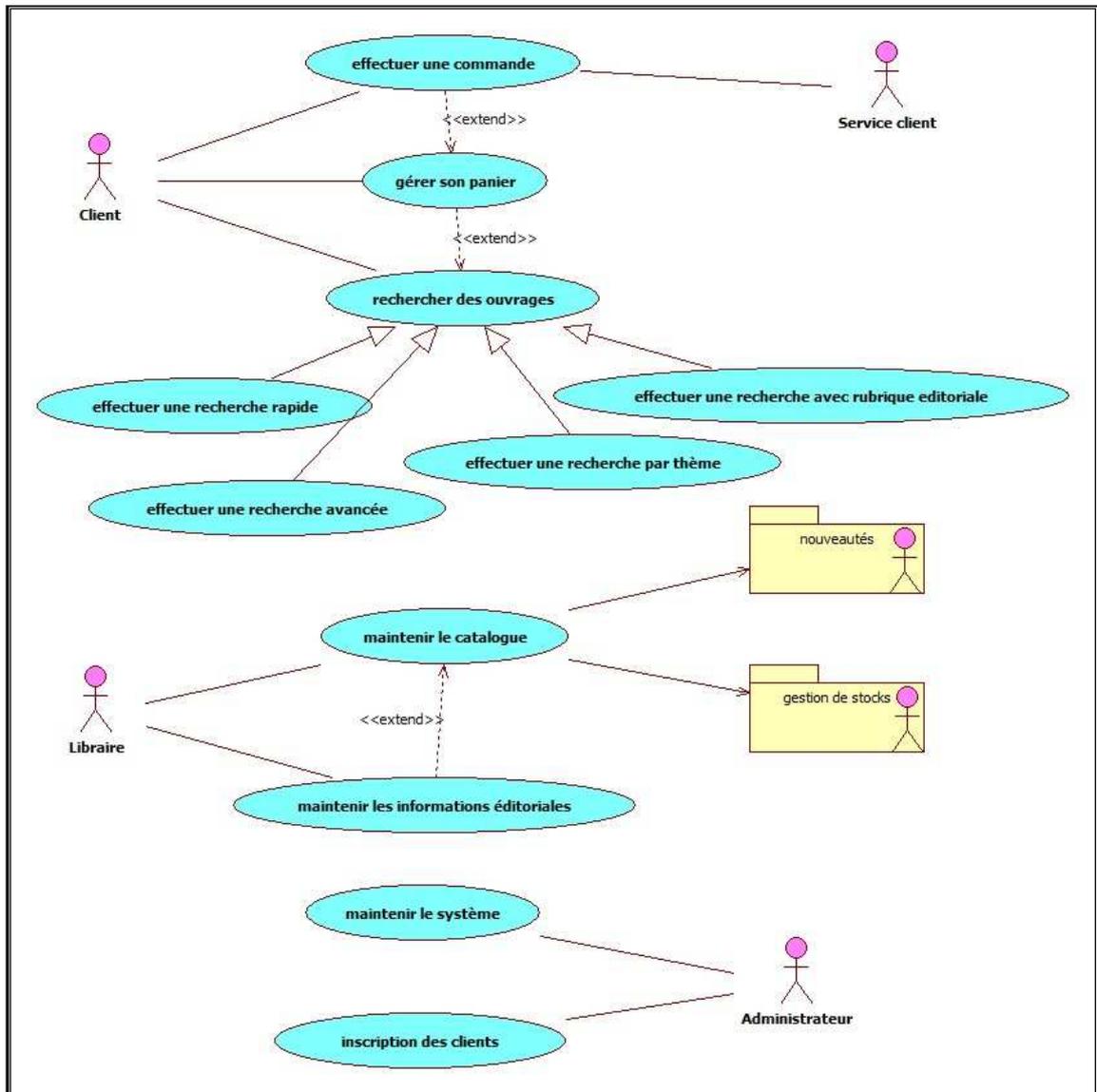


Figure II.2 : Diagramme des cas d'utilisation du système GML.

III.3. Détermination des cas d'utilisation

Pour donner une autre définition du cas d'utilisation, on peut dire que c'est une collection de scénarios de succès ou d'échec qui décrit la façon dont un acteur particulier utilise un système pour atteindre un objectif. Pour détailler la dynamique du cas d'utilisation, la procédure la plus évidente consiste à recenser de façon textuelle toutes les interactions entre les acteurs et le système.

A partir de chaque scénario défini ci-dessous nous dresserons pour chacun d'eux le diagramme de séquence correspondant.

- ❖ rechercher des ouvrages.
- ❖ gérer son panier.
- ❖ effectuer une commande.

- ❖ maintenir le catalogue.

III.3.1. Rechercher des ouvrages

- **Acteur principale**

Client.

- **Objectif**

Le client veut trouver le plus rapidement possible un ouvrage recherché dans l'ensemble du catalogue. Il veut aussi pouvoir flâner et chercher des livres avec des critères variés.

- **Scénario**

1. Le client lance une recherche rapide à partir d'un mot clé : thème, titre, ISBN.
2. Le système affiche une page de résultat.
3. Le client sélectionne un ouvrage.
4. Le système lui présente une fiche détaillée pour l'ouvrage sélectionné.
5. Le client met l'ouvrage de côté dans son panier.

Le diagramme suivant (figure II.3) illustre ce scénario.

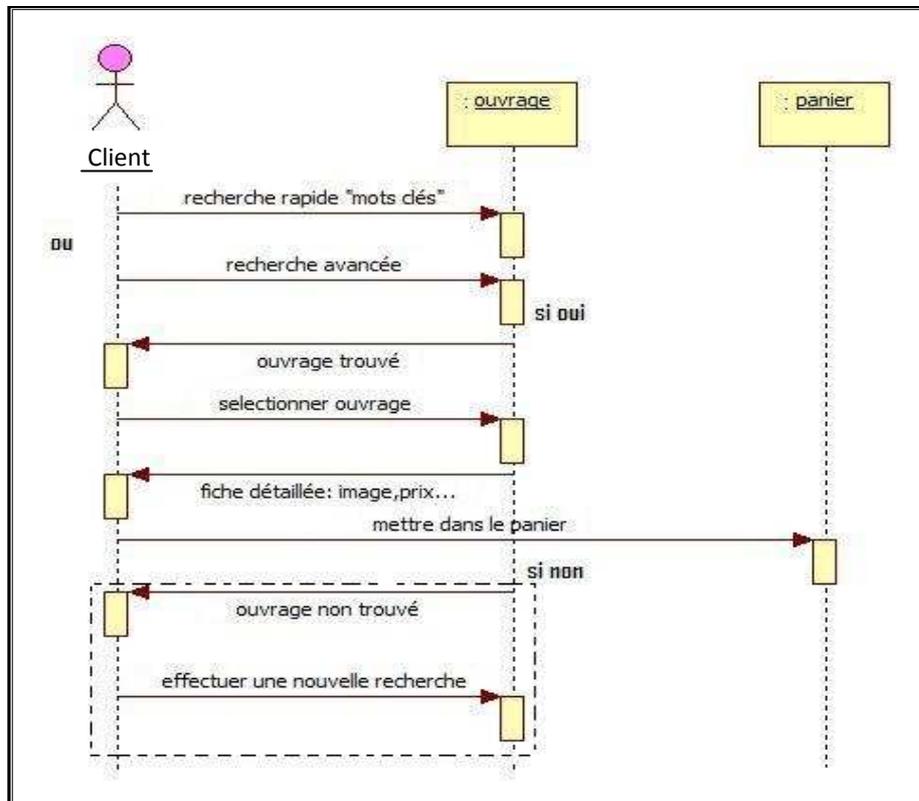


Figure II.3 : Diagramme de séquence de recherche des ouvrages.

III.3.2. Gérer son panier

- **Acteur principale**

Client.

- **Objectif**

Lorsque le client est intéressé par un ouvrage, il doit avoir la possibilité de l'enregistrer dans un panier, puis d'ajouter d'autres livres, en supprimer ou encore en modifier les quantités avant de passer commande.

- **Scénario**

1. Le client enregistre les ouvrages qui l'intéressent dans son panier (voir le cas d'utilisation **rechercher des ouvrages**).
2. Le client demande l'accès à son panier.
3. le système lui affiche l'état de son panier.
4. Le client valide son panier en demandant à **effectuer une commande**.

Ce cas d'utilisation est représenté dans la figure ci-dessous par le scénario suivant (figure II.4)

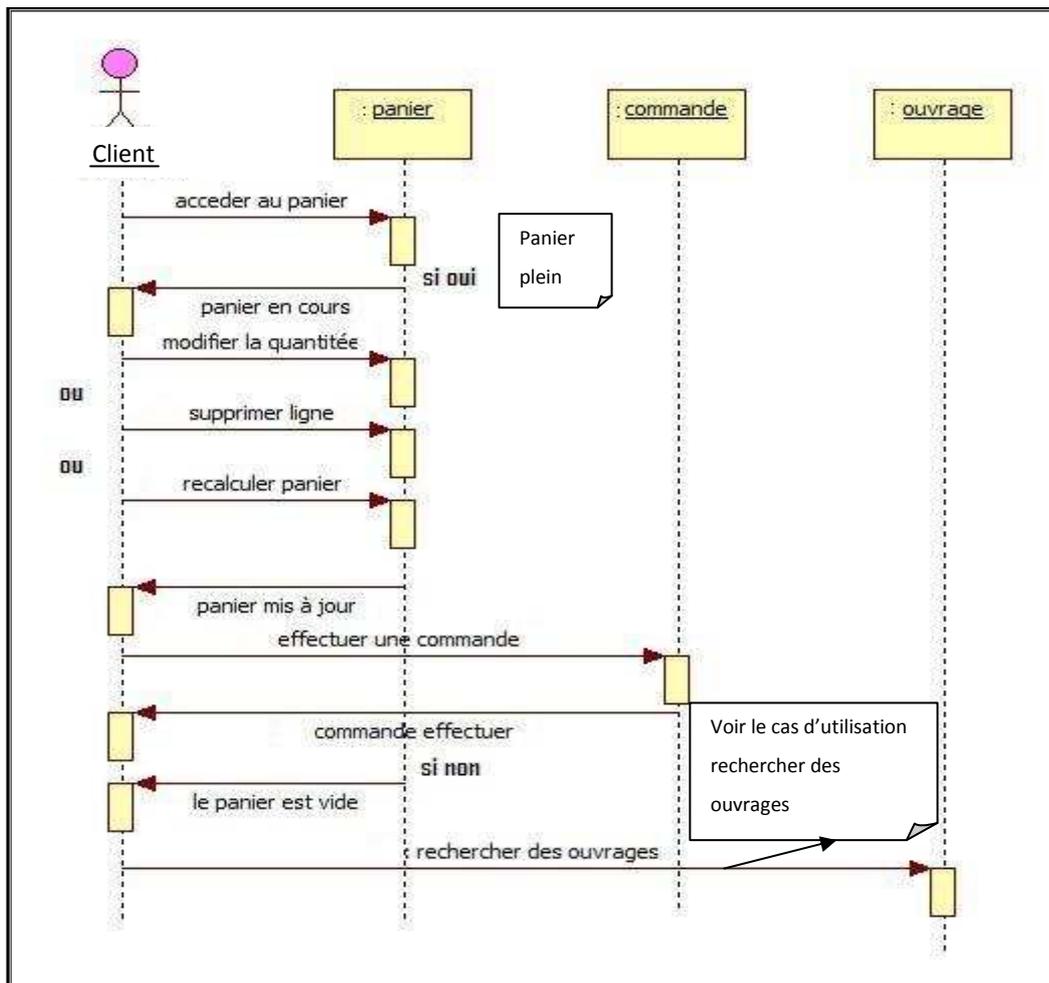


Figure II.4 : Diagramme de séquence de gérer son panier.

III.3.3. Effectuer une commande

- **Acteur principale**

Client.

- **Objectif**

A tout moment, le client doit pouvoir accéder au formulaire du bon de commande, dans lequel il peut saisir ses coordonnées et les informations nécessaires au paiement et à la livraison.

- **Scénario**

1. Le client saisit l'ensemble des informations nécessaire au paiement et à la livraison.
2. Le système affiche une récapitulatif de la commande.
3. Le client valide sa commande.
4. Le système envoi la commande valider au service clients du système GML
5. Le système confirme la prise de commande de client.

Ce cas d'utilisation est représenté dans la figure ci-dessous par le scénario suivant (figure II.5).

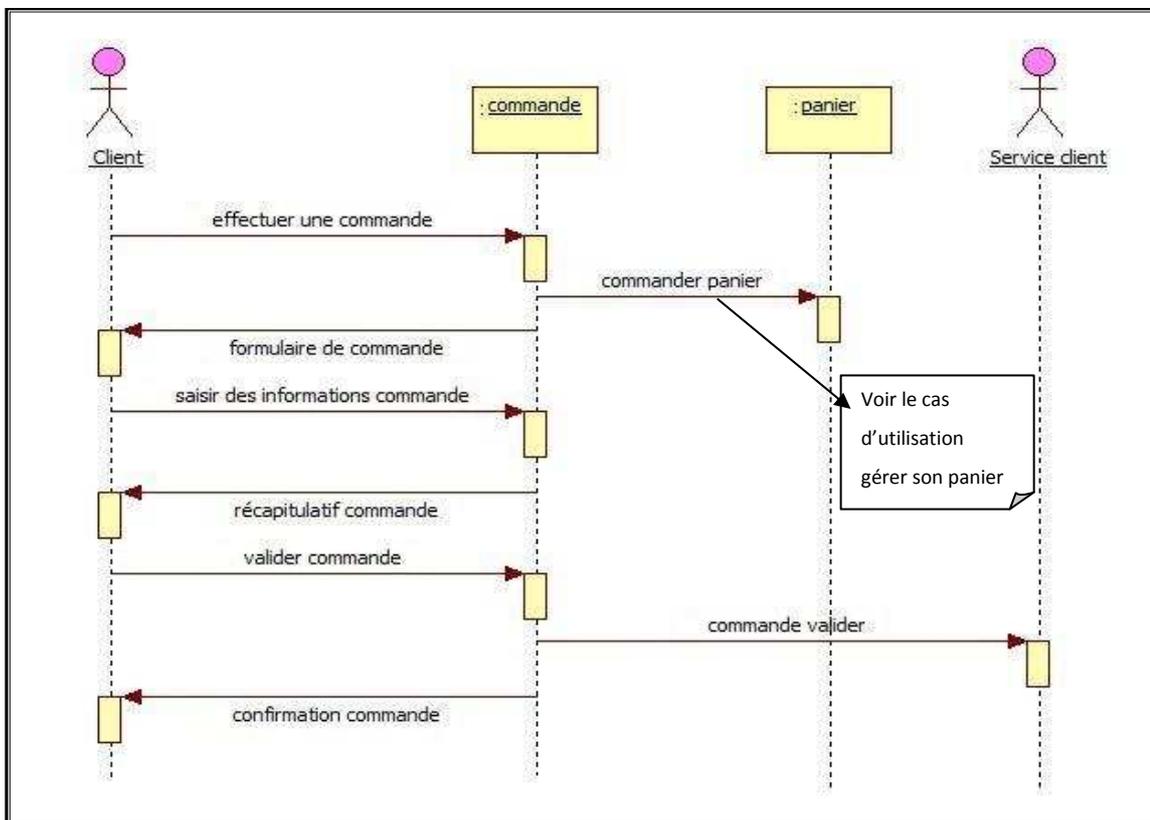


Figure II.5 : Diagramme de séquence d'effectuer une commande.

III.3.4. Maintenir le catalogue

- **Acteur principale**

Le libraire.

- **Acteur secondaire**

Les deux systèmes « nouveauté » et « gestion des stocks »

- **Objectif**

Le libraire veut pouvoir contrôler la mise à jour automatique du catalogue des ouvrages présentés sur le système.

- **Scénario**

1. Le système **nouveauté** alimente le site avec les nouveaux ouvrages.
2. Le système **gestion des stocks** met à jour les données qui concernent le prix et l'état du stock.
3. Le libraire valide la mise à jour de catalogue.

Le diagramme suivant (figure II.6) montre se scénario.

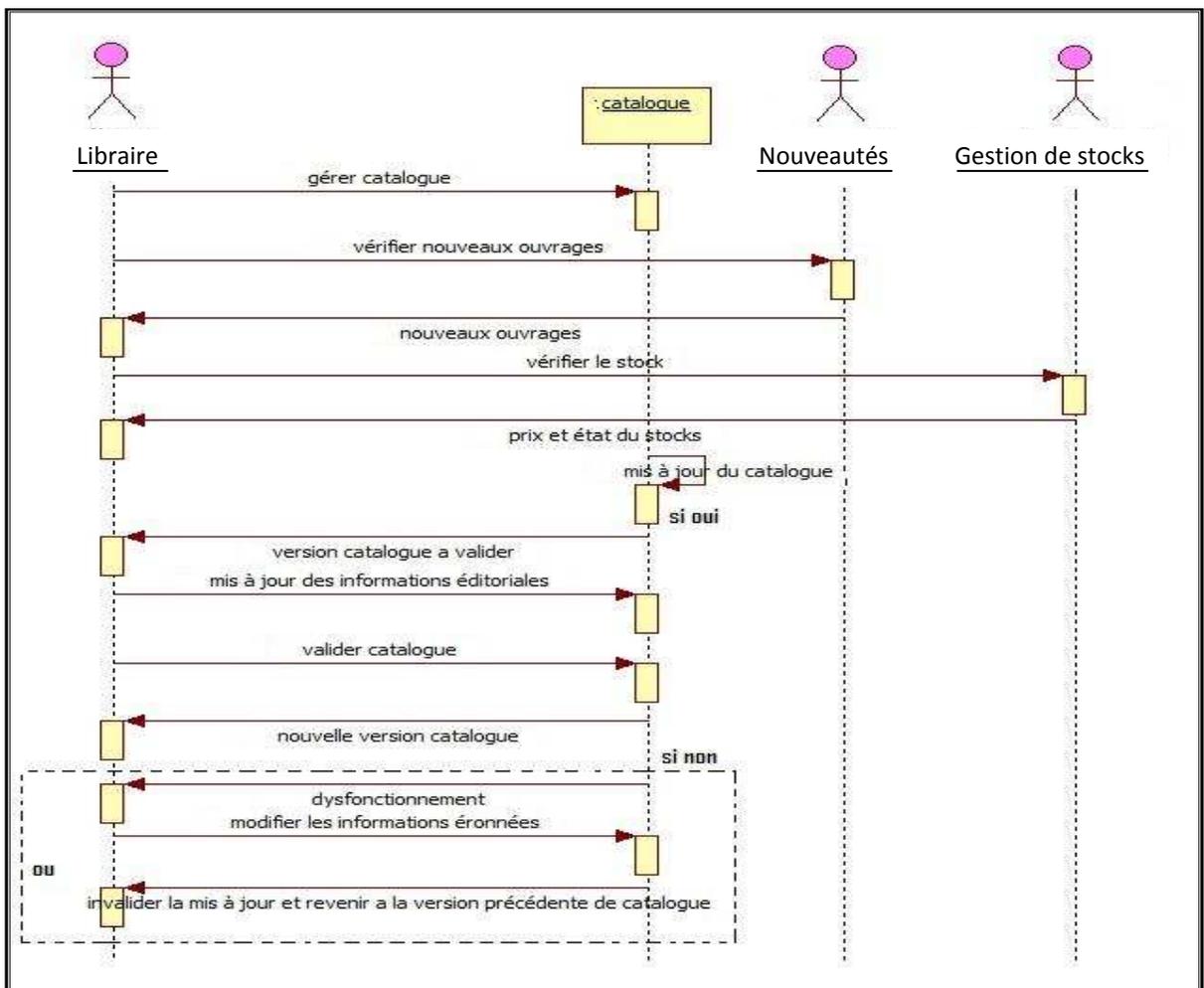


Figure II.6 : Diagramme de séquence de maintenir le catalogue

III.4. Diagramme de classes

A partir des diagrammes de séquence et de la liste d'objets tirés, nous pouvons dresser le diagramme de classe qui expose les relations entre ces objets.

La figure(II.7) illustre notre diagramme de classe :

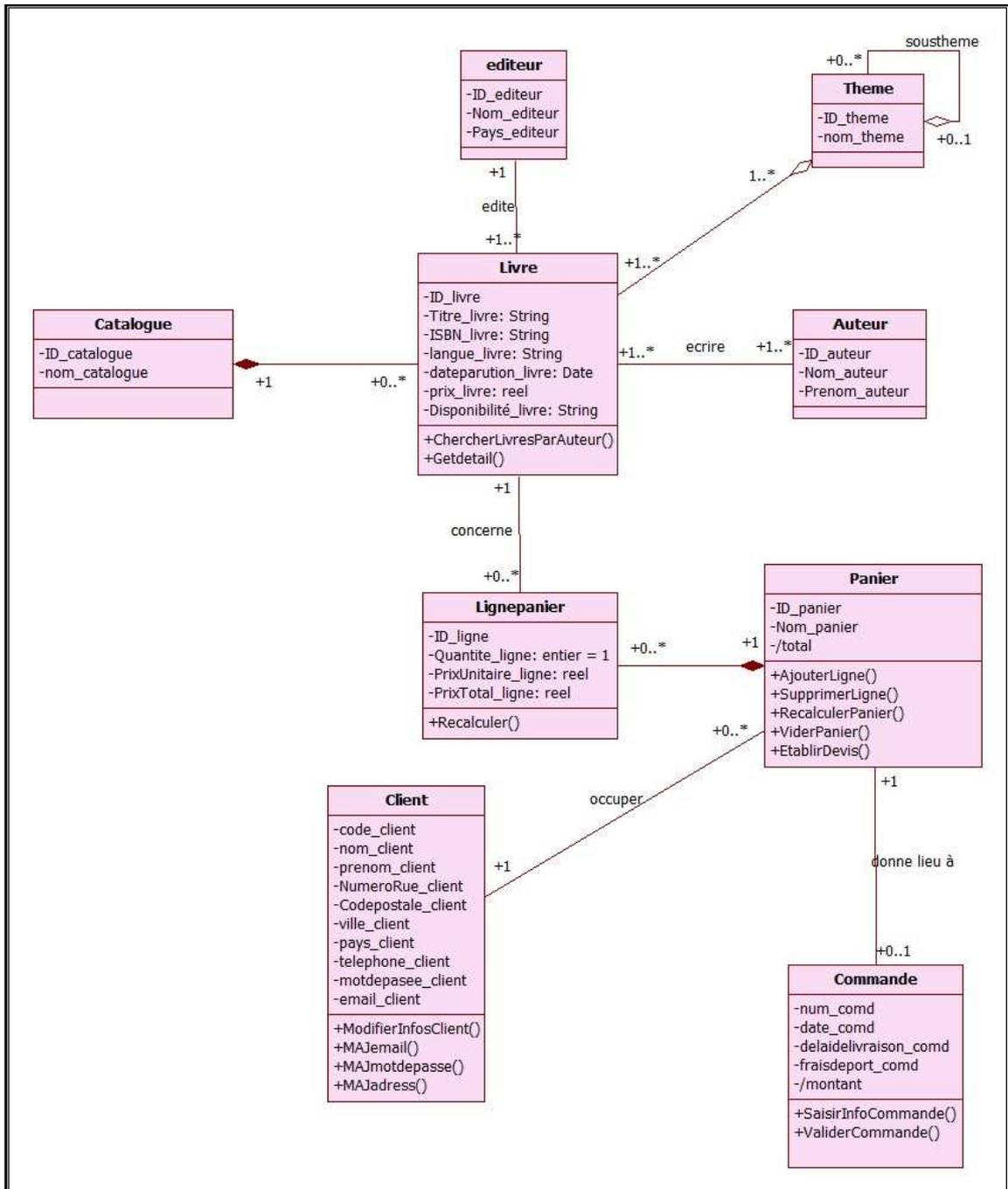


Figure II.7 : Diagramme de classe du système GML

III.5. Le model relationnel de données

III.5.1. Les règles de passage

❖ *Le cas des classes*

Pour chaque classe du model conceptuel, une table est créée dans le model relationnel, sauf dans le cas d'une association d'héritage où les attributs de la table en cause peuvent être combiné pour former une table. Les attributs de la classe deviennent les attributs de la table. Dans le cas d'une classe forte, son identifiant simple ou composé devient la clé primaire de la table, et que chaque attribut formant la clé porte la mention {PK}.

❖ *Le cas des entités d'associations*

En ce qui concerne une classe d'association, la table correspondante aura comme clé {PK} une clé obligatoirement composé de tout les identifiants des classes participant à l'association et que chaque attributs de la clé composée portera la mention {PK, FK}.

❖ *Les associations binaires : un à plusieurs*

Les associations un à plusieurs sont les plus nombreuses dans un modèle conceptuel. Or il s'avère que c'est aussi le type d'association comportant la règle de dérivation la plus simple. Elle se résume ainsi : la table dérivée de la classe du côté de la multiplicité maximale plusieurs (*) est considérée comme table fille. En conséquence, une copie de la clé primaire de la deuxième table, la table mère, est déposée dans la table fille à titre de clé étrangère

❖ *Les associations binaires : plusieurs à plusieurs*

Une association plusieurs à plusieurs sans classe d'association donne lieu a une table fille comportant comme clé primaire la combinaison des clés primaires des tables mères. Chaque composant de la clé primaire est aussi une clé étrangère. [6]

III.5.2. Le model logique de données

Nous appliquons les règles de passages mentionnées ci-dessus pour obtenir le modèle logique de données suivant :

Editeur (ID_editeur, nom_editeur , pays_editeur) ;

Livre (ID_livre,titre_livre,ISBN_livre,langue_livre,datedeparution_livre,prix_livre, disponibilité_livre, ID_editeur*, ID_catalogue*) ;

Theme (ID_theme,nom_theme,ID_Stheme*);

Contenir (ID_livre*, ID_theme*);

Ecrire (ID_auteur*, ID_livre*);

Auteur (ID_auteur, nom_auteur,prenom_auteur) ;

Catalogue (ID_catalogue,Nom_catalogue) ;

Ligne panier(ID_ligne,quantites_ligne,prixunitaire_ligne,prixtotal_ligne,ID_livre*,ID_panier*) ;

Panier (ID_panier,Nom_panier,total,code_client*) ;

Commande (num_comd,date_comd,delaidelivraison_comd,fraisdeport_comd,montant, ID_panier*) ;

Client (code_client,nom_client,prenom_client,numerorue_client,codepostale_client, ville_client,pays_client,telephone_client,motdepasse_client,email_client) ;

IV. Conclusion

Dans ce chapitre, nous avons détaillé le processus d'analyse et de conception de notre système GML. En premier lieu, nous avons décrit les différents cas d'utilisation du système. En deuxième lieu, ces cas d'utilisation sont interprétés en diagrammes de séquence qui ont permis d'aboutir, en dernier lieu au diagramme de classe.

Cette conception constitue un point de départ à l'implémentation mis en œuvre dans le chapitre suivant (chapitre 3).