Khelil Youcef
Marchetti Pierre
Robert François
Roffet Antoine
Salamani Léandre
Saule Laura

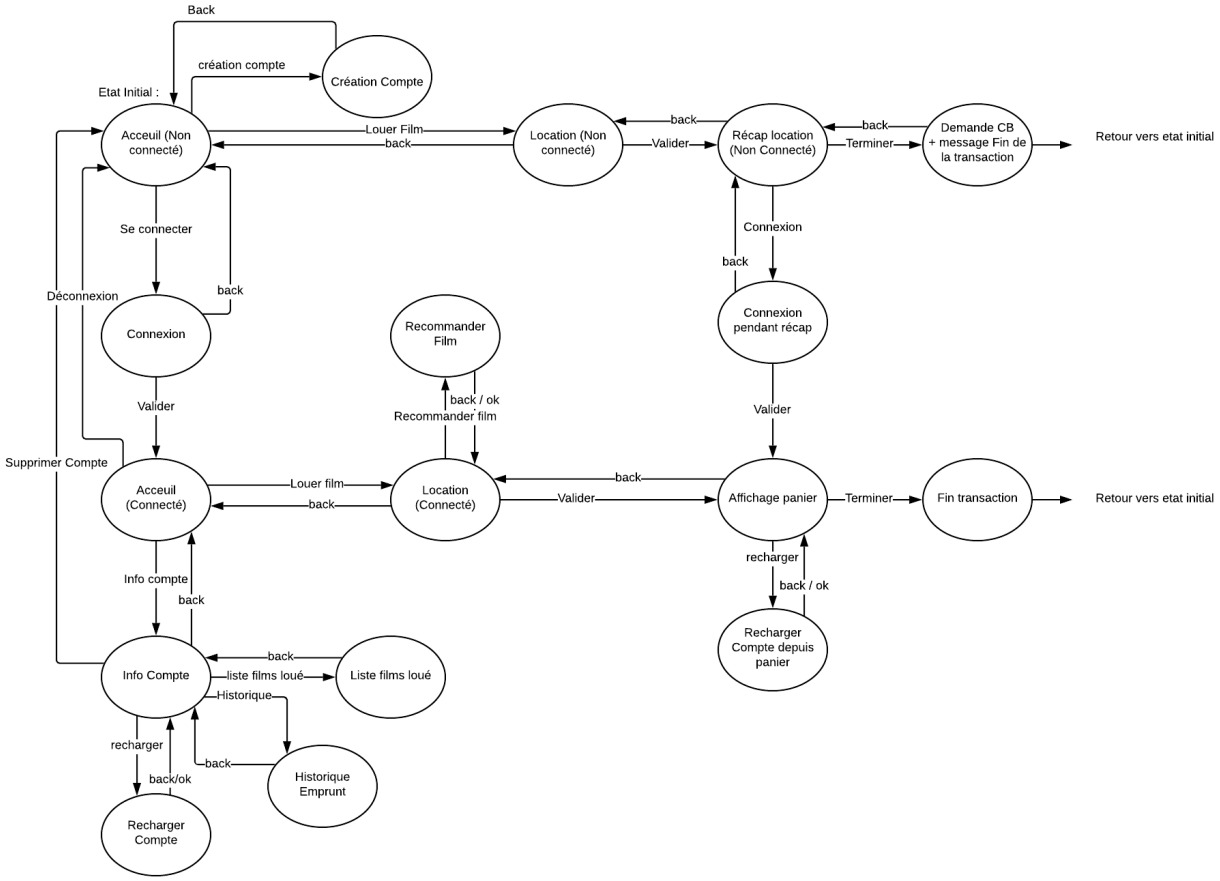
Projet CyberVideo

Objectifs :
L'objectif de ce projet est de réaliser le logiciel d'un automate de réservation pour un loueur de DVD CyberVideo.
Pour cela, plusieurs tâches ont été définies.

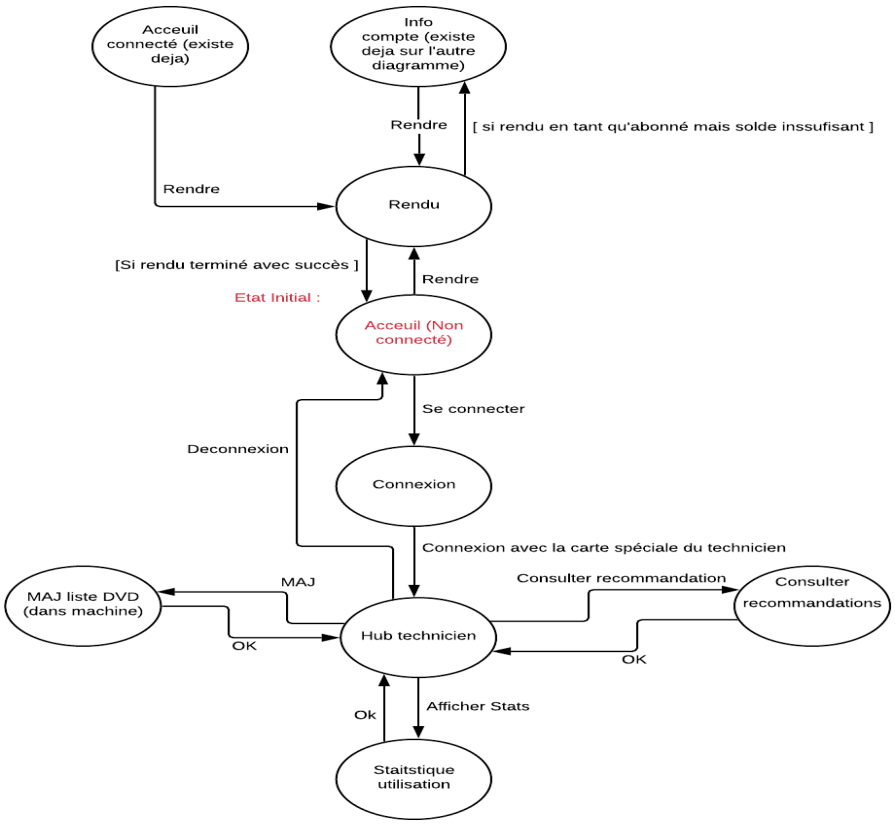
- interview du client pour rédiger la SRS
- modélisation UML pour concevoir notre projet.
- implémentation de la conception UML en Java

Réalisations :

Notre programme exécutable final se présente de la façon suivante :
L'utilisateur rentre en ligne de commande des commandes afin d'agir sur le programme, et se voit afficher des informations dans le terminal en fonction de ses actions. Ci dessous un diagramme qui représente les différentes « fenêtre » du programme.



Voici également un diagramme représentant d'autres « fenêtres » de la machine, notamment le rendu d'un DvD ainsi que l'interface proposé au technicien.



Ces diagrammes peuvent être utilisés comme aide à l'utilisation du programme, l'interface sur le terminal n'étant pas aussi agréable à utiliser qu'une interface classique.

Nous vous présentons également deux nouveaux diagrammes de classes qui sont accessible dans l'archive avec ce rapport. Ces diagrammes sont bien différents de ceux prévu avant l'implémentation, suite aux lacunes de ces derniers, nous exprimerons ce qu'il nous manquait dans notre première conception.

Nous avons rapidement remarqué lorsque nous avons commencé l'implémentation que ne pas avoir pris en compte la partie « base de données » (que nous avons fait au final avec un système de fichier), ni la partie « interface utilisateur » pendant notre conception allait changer beaucoup de choses. En effet il nous manquait beaucoup de fonctions dans nos classes, des attributs auxquels nous n'avions pas pensé également, pour remédier à ce problème nous avons décider d'implémenter notre application avec un design patern de type « Modèle-Vue-Contrôleur ».

Notre vue se présente sous la forme d'une machine à état, en fonction de l'entrée saisie au clavier par l'utilisateur on change d'état dans la machine à état, ou bien on fait un appel au contrôleur qui va pouvoir faire des appels de fonctions qui vont modifier le modèle.

Le contrôleur est représenté par les classes ModeleEmprunteur et ModeleTechnicien, ces deux classes contiennent l'ensemble des fonctions qui seront appelé dans les différents états de la vue, afin de pouvoir modifier le modèle.

Le modèle est représenté par les données stocké dans les fichiers auquel va accéder notre classe BD, les instances d'objets de nos classes DVD, Film, Emprunt, … sont temporaires. On s'en sert pour représenter les données présente dans les fichiers, les modifier avec les appels du contrôleur, puis enregistrer les modifications sur les fichiers correspondants. La classe BD contient donc toutes les fonctions nécessaire à la lecture et écriture dans nos fichiers de données en fonction des modification que peut avoir besoin de faire notre contrôleur.

Cette façon de faire nous a permis de bien diviser le travail au sein du groupe :

Un groupe s'occupait de la machine à état, faire les transitions entre les états, quels appels de fonction il faudrait faire, dans quel cas, quels paramètres donner et quelle est la valeur de retour souhaitée etc.

Un autre sous groupe s'occupait de la partie contrôleur, en implémentant les fonctions dont avait besoin la vue, utilisant des fonctions faisant appel à la base de donnée, on retrouve toute la partie « algorithmique » que nous avions le plus développer lors de notre première conception.

Un autre groupe s'est occupé de la partie base de donnée, afin de fournir au contrôleur les accès aux données dont il avait besoin, les fonctions de recherche, d'écriture, définir le format des « tables » en fonction des attributs de nos classes, etc..

L'ensemble des fonctions des classes ModeleEmprunteur et ModeleTechnicien ont été testé avec junit.

Cependant nous avons rencontré certains obstacle, nous allons donc lister ici les fonctionnalités spécifié dans la SRS que nous avons pu mettre en œuvre, le reste sera a considéré comme non terminée ou non fonctionnel et a donc été retiré du code qui sera rendu.
Au lancement du programme l'utilisateur peut :

décider de se connecter (en rentrant son id de carte abonné)

décider de créer un compte en rentrant son nom, prénom et numéro de carte bancaire (pour obtenir son id de carte il faut aller voir dans le fichier « BDD/abonne » ce qui correspondrait dans la réalité à aller chercher sa carte auprès du vendeur dans le magasin)

se connecter en tant que technicien, pour cela l faut utiliser l'id 1 (cela correspondrait dans la réalité à se connecter avec une carte spéciale que seul le technicien possède)

Louer un film, il sera redirigé vers la sélection des films, ou il pourra remplir son panier (selon s'il est connecté ou non la taille du panier varier), valider son panier (un récapitulatif lui sera afficher à l'écran), il pourra finalement valider sa transaction ce qui le renverra à l'accueil.

Si l'utilisateur s'est connecté il peut

avoir accès aux infos de son compte :

la liste des films qu'il a emprunté, son historique d'emprunts, il pourra également recharger son compte (en rentrant son numéro de carte bancaire ainsi que le montant désiré)

se déconnecter