

Projet Informatique Appliqué

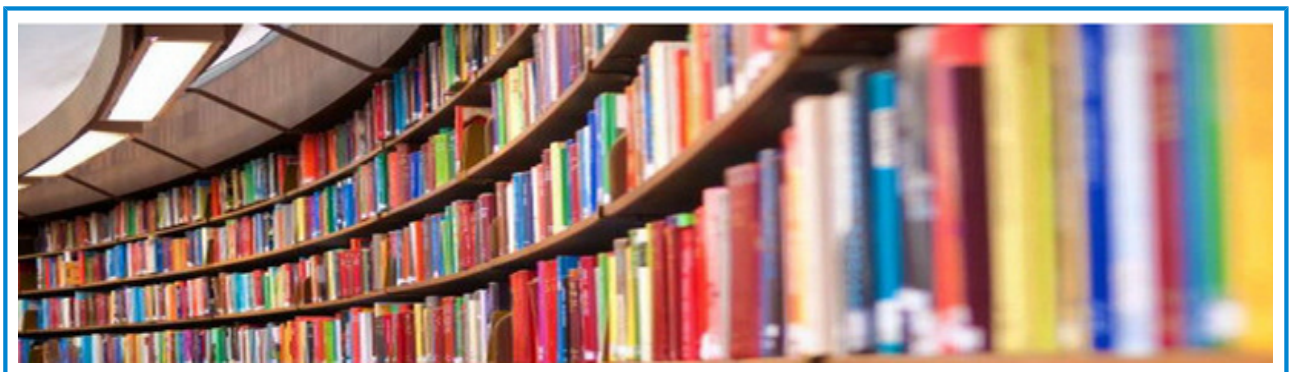
Gestion d'une bibliothèque

MDJASSIR Amirdine

NARRAN Maxime

SOW Abdoulaye

07 Mai 2014



SOMMAIRE

I - Gestion du projet	2
1.1 - Présentation du groupe	2
1.2 - Présentation du projet	2
1.3 - Diagramme de Grant	2
 II - Cahier des charges	 2
2.1 - Études de l'existant	3
2.2 - Études des besoins	4
2.3 - Études de faisabilité	5
2.4 - Études préalable	6
2.5 – Règles de gestion	8
2.6 - Modélisation Merise	9
2.7 - Analyse fonctionnelle	12
 III - Programmation de l'interface	 14
3.1 - Architecture de l'application	14
3.2 - Code source	15
3.3 - Interface graphique	18
 IV – Conclusion	 20

I - Gestion du projet

1.1 Présentation du groupe

Notre groupe de travail est composé de trois personnes. Toutefois, nous avons réussi à nous organiser et partager le travail en deux grandes parties, à savoir la création de la base de données puis la création de l'interface graphique en PHP comme première partie du travail et la rédaction du cahier des charges, création des formulaires en html, css, la page d'accueil, la saisie du rapport de projet comme la deuxième et dernière partie du travail. Nous avons rendez-vous au moins deux fois par semaine pour évaluer le travail et se poser des questions afin d'atteindre l'objectif qu'on s'est fixé pour le projet. Nous avons rencontré très souvent des problèmes tels que des problèmes d'organisation, de compréhensions du code PHP aussi bien que la saisie du cahier des charges. On prenait des rendez-vous avec notre professeur et encadreur Madame Line JAKUBIEK pour avoir des explications et quelques corrections. Les trois membres formant ce groupe de travail sont donc Sow Abdoulaye, Maxime Narran et Mdjassiri Amirdine tous trois en 3^{ème} année de licence en Informatique à l'université Aix Marseille II.

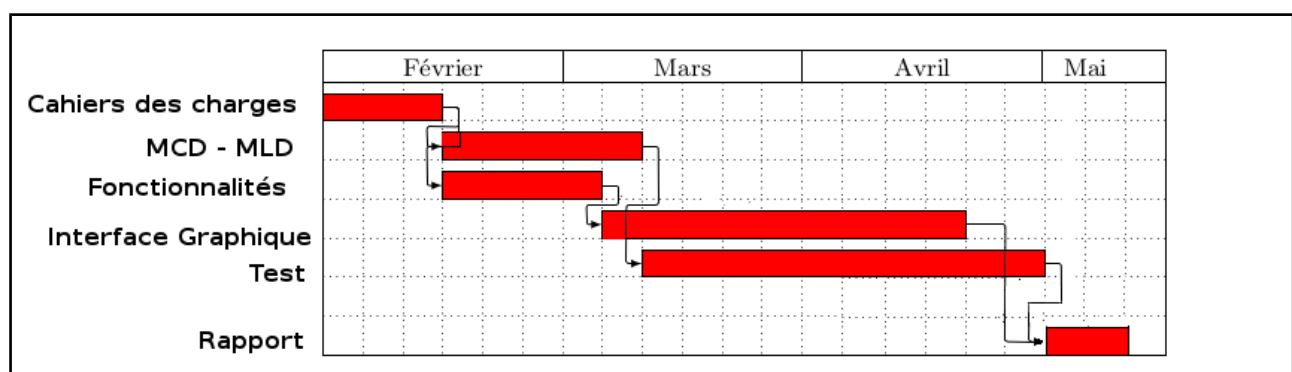
1.2 Présentation du projet

Nous cherchons à réaliser un logiciel de gestion d'une bibliothèque. Le fond d'une bibliothèque est composé d'un certain nombre d'oeuvres. Chaque œuvre est identifiée par une cote et caractérisée par un ou plusieurs noms d'auteurs, un titre, une date de parution et un certain nombre de mots-clefs permettant son classement analytique.

La bibliothèque possède en moyenne 3 ou 4 exemplaires de chaque œuvre, qu'on appelle LIVRE un exemplaire de chaque œuvre. Un livre est identifié par un numéro d'exemplaire et la cote de l'oeuvre correspondante; il est caractérisé par une date d'achat.

Les livres peuvent être empruntés par les adhérents de la bibliothèque ; un adhérent est repéré par un numéro, et caractérisé par son nom, son adresse, sa date d'adhésion et sa dernière date de paiement de cotisation. La bibliothèque(le bibliothécaire) gère les prêts et les réservations d'oeuvre. Elle garde l'historique des prêts effectués pour un livre donné.

1.3 Diagramme de Grant



II – Le cahier des charges

2.1 Études de l'existant

Au niveau de l'existant, nous avons une bibliothèque de taille moyenne, avec un certain nombre d'œuvres. Dans cette étude, il est important de faire la distinction entre une œuvre et un livre. Une œuvre est une production littéraire unique écrite par un ou plusieurs auteurs. Et un livre est un exemplaire d'une œuvre. Par exemple, pour une même œuvre, il peut y' avoir plusieurs exemplaires écrits par des maisons d'éditions différentes. Cette bibliothèque possède en moyenne trois ou quatre exemplaires pour chaque œuvre, que l'on appelle Livre. Un livre est identifié par un numéro d'exemplaire et la cote de l'œuvre correspondante. Il est caractérisé par une date d'achat.

Ces livres peuvent être empruntés par les adhérents de la bibliothèque. Chaque adhérent est repéré par un numéro et caractérisé par son nom, son adresse, sa date d'adhésion et sa dernière date de paiement de cotisation. La bibliothèque gère les prêts , et les réservations d'œuvre . Elle garde l'historique des prêts effectués pour un livre donné. Nous avons donc à gérer les données sur l'adhérent, les œuvres, les livres (exemplaires) et les prêteurs.

Concernant l'adhérent et les prêts, nous suggérons l'ajout de une ou plusieurs caractéristiques. Pour l'adhérent, il serait important d'ajouter l'âge, le sexe, son numéro de téléphone et son adresse e-mail. Et pour chaque adhérent , il faudrait limiter le nombre d'emprunt total à cinq livres et interdire l'emprunt de plusieurs exemplaires identiques.

2.2 Études des besoins

Pour des raisons d'efficacité, la bibliothèque désire une Base de Données permettant d'informatiser divers traitements comme l'enregistrement d'un prêt, d'un renouvellement d'un prêt ou le retour d'un livre. Lorsqu'un adhérent dépose une demande de prêt ou de prolongation, le service de la bibliothèque doit lui répondre en lui remettant soit le ou les livres demandés, soit une notification de réservation.

Pour les adhérents n'ayant pas rendu leur livre, le bibliothécaire envoie une fois par semaine aux adhérents retardataires une lettre leur rappelant que la durée d'un prêt ou d'un renouvellement est de un mois, et ce avec la liste des livres concernés. Quand un adhérent rend un livre emprunté, son emprunt passe dans l'historique du livre qui devient disponible pour un nouveau prêt.

Lors de la réception d'un nouveau livre provenant d'un achat, ce livre devient disponible.

La Bibliothèque souhaite aussi que chaque adhérent puisse consulter la base de livres existant à la bibliothèque, c'est-à-dire obtenir entre autres les œuvres auxquelles un auteur a participé, les œuvres traitant d'un sujet donné par une liste de mots-clefs.

Une demande de prêt non satisfaite est enregistrée sous forme de réservation, restant en attente d'une disponibilité. Chaque fois qu'un livre devient disponible, le bibliothécaire doit proposer par courrier , à l'adhérent dont la réservation est la plus ancienne . Si l'adhérent vient le chercher dans la semaine qui suit, ce prêt lui est accordé, s'il vient trop tard, il lui est refusé et sa réservation est annulée, rendant ce livre à nouveau disponible.

Dans cette étude des besoins, nous suggérons que les envois par courrier soit remplacés par des envois automatiques par e-mail.

2.3 Études de faisabilité

Les tâches seront réparties équitablement entre chaque membre du groupe.
Les données devront d'abord être modélisées et mise sous forme pour la base de données .
Notre modélisation merise et nos tables seront réalisées à l'aide du logiciel Jmerise.

La Base de données devra être réalisée sous MySql. Elle sera implémentée à partir de très nombreux exemples que nous aurons inventées .

La méthode hybride est celle qui est retenue. Les demandes de prêt, les consultations, les inscriptions , les enregistrements des retours des livres , les achats de nouveaux livres et les demandes de réservations, seront gérés par le bibliothécaire.

La relance des adhérents , les avertissements de réservations et la suppression des prêts seront faits automatiquement . Les mails seront privilégiés par rapport aux lettres.

Les informations contenues de notre base de données seront mises à jour en conséquence.
De toutes les étapes auxquelles, nous serons confrontés , c'est l'automatisation des démarches que nous redoutons le plus et qui nous prendra certainement le plus de temps.

2.4 Analyse préalable

Le dictionnaire de données

NOM	Nom de l'adhérent	A	Élément		Martin
PRENOM	Prénom de l'adhérent	A	Élément		Jean
ADRESSE	Adresse de l'adhérent	AN	Com	N° Rue + rue + code postal +ville	100 Rue de Rome 13001 Marseille
COURRIEL	Adresse e-mail de l'adhérent	AN	Élément	****@****.***	jean, martin@gmail.fr
DATE_ADHESION	Date d'adhésion de l'adhérent	Date	Élément		25 Jan 2010
NUMERO_ADHERENT	Numéro de l'adhérent	AN	Élément		2301400n
DD_PAIEMENT	Date de paiement de l'adhérent	Date	Élément		18 Dec 2014
ISBN	Cote du Livre	AN	Élément		VIC874
DATE_EMP	Date du prêt	Date	Élément	Inférieur ou égal à la date d'aujourd'hui	20 Mai 2014
DATE_RETOUR	Date du retour du prêt	Date	Élément		18 Juin 2014
NUMERO_EXEMPLAIRE	Numéro du livre (exemplaire)	N	Élément		1
DATE_ACHAT	Date d'achat du livre (exemplaire)	Date	Élément	Inférieur a la date du retour et à la date de prêt	31/07/90
CODE_BARRE	Code Barre du livre	N	Élément		
TITRE	Titre de l'œuvre	AN	Élément		Les Misérables
NOM_AUTEUR	Auteur de l'œuvre	A	Élément		Hugo
DATE_PARUTION	Date de parution de l'œuvre	Date	Élément	Inférieur à la date du prêt	1862
NOMBRE_EXEMPLAIRE	Nombre d'exemplaire disponible pour une œuvre	N	Calculée	Supérieur a zéro	5

DATE _RESERVATION	Date de réservation d'un livre	Date	Élément	Supérieur à la date du prêt et inférieur à la date de parution de l'œuvre	30 Juin 2014
PROLONGATION	Si prolongation de l'emprunt pendant 1 semaine	Bool	Élément		oui
ID_AUTEUR	Numéro de l'auteur	N	Élément		12006n
Prénom_AUTEUR	Prénom de l'auteur	A	Élément		hugo
Date_NAISSANCE	Date de naissance de l'auteur	Date	Élément		1926
NUMMOTSCLES	Numéro l'identification des mots clés.	A	Élément		15412
MOTS_CLES	Mots clés du livre	A	Comp		Guerre France Verdun Allié
DISPONIBILTE	Disponibilité du livre	Bool	Élément		oui
VILLE	Ville de résidence de l'adhérent	A	Élément		Paris
ID_EDITEUR	Identifiant de l'éditeur	N	Élément		155161658
NOM_EDI	Nom de l'éditeur	A	Élément		hachette
ADRESSE_EDI	Adresse de l'éditeur	A	Com		
SEXE	Sexe de l'adhérent	A	Élément	Masculin ou féminin	masculin
DATE_NAISSANCE	Date de naissance de l'adhérent	Date	Élément		31/02/92
DATE_LIM	Date à laquelle la réservation est annulée si l'adhérent ne confirme pas son prêt	Date	Élément		25/09/19
TELEPHONE	Numéro de téléphone de l'adhérent	N	Élément		0236897412

2.5 Règles de gestions

RG1:Chaque œuvre se compose d'un ou plusieurs livres.

RG2:Chaque adhérents ne peut pas avoir plus de 3 prêts actifs à la fois.

RG3:Chaque Adhérents ne peut faire plus de 5 réservations.

RG4:Un livre ne peut être emprunté que par un seul adhérent à la fois.

RG5:Un Adhérent ne peut prolonger son prêt qu'une seule fois.

RG6:Si l'Adhérent demande un livre qui n'est pas disponible il peut être mis sur réservation.

RG7:Si le livre est ou devient disponible, l'adhérent qui a la réservation la plus ancienne obtient le prêt.

RG8:Si le livre réservée devient disponible et que l'adhérent à qui il doit revenir ne le récupère pas au bout d'un certain temps :la réservation est annulée.

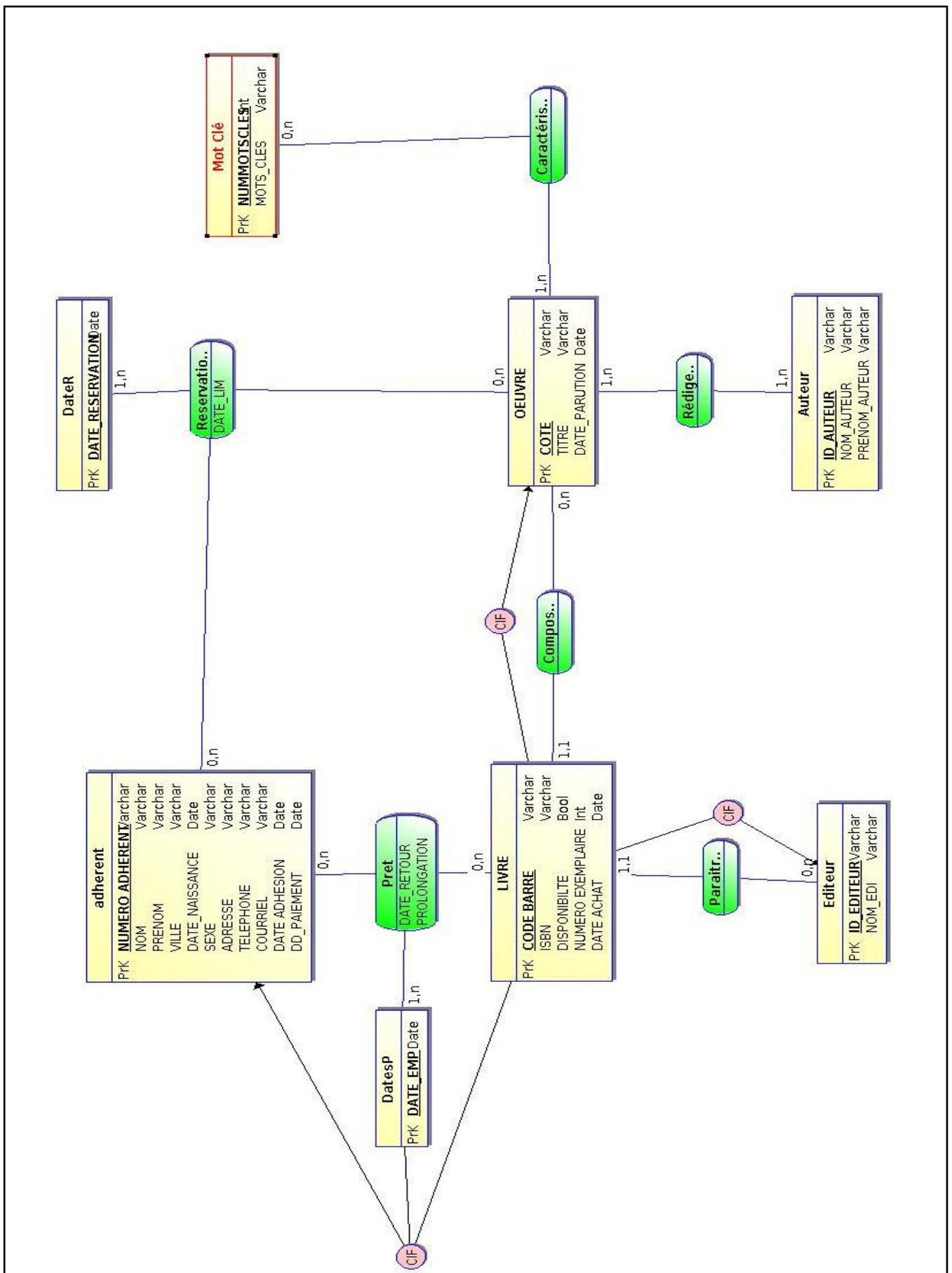
RG9:Une maison d'édition est responsable de plusieurs livres.

RG10:Une œuvre peut être écrite par un ou plusieurs auteurs.

RG11:Une œuvre est caractérisée par zéro ou plusieurs mots clés.

Note : La règle 2 et 3 n'ont pas été respectées, celles-ci jugées non obligatoires et par manque de temps.

2.6 Modélisation Merise



Modèle Physique

```
CREATE TABLE ADHERENT (  
Numero_ADHERENT Varchar (25) NOT NULL ,  
NOM Varchar (25) NOT NULL,  
PRENOM Varchar (25) NOT NULL,  
VILLE Varchar (25) NOT NULL,  
DATE_NAISSANCE DATE NOT NULL,  
SEXE Varchar (25) NOT NULL,  
ADRESSE Varchar (350) NOT NULL,  
COURRIEL Varchar (50) NOT NULL,  
TELEPHONE Varchar (25) NOT NULL,  
DATE_ADHESION DATE NOT NULL,  
DD_PAIEMENT DATE NOT NULL,  
PRIMARY KEY(Numero_ADHERENT)  
)ENGINE = INNODB;  
  
CREATE TABLE OEUVRE (  
COTE Varchar (25) NOT NULL ,  
TITRE Varchar (25) NOT NULL ,  
DATE_PARUTION Date NOT NULL ,  
PRIMARY KEY(COTE)  
)ENGINE = INNODB;  
  
CREATE TABLE EDITEUR (  
ID_Editeur Varchar (25) NOT NULL , NOM_EDITEUR Varchar (25) NOT NULL , PRIMARY KEY(ID_Editeur)  
)ENGINE = INNODB;  
  
CREATE TABLE LIVRE (  
CODE_BARRE Varchar (25) NOT NULL ,  
ISBN Varchar (25) NOT NULL ,  
DISPONIBILITE Bool ,  
NUMERO_EXEMPLAIRE int (30) NOT NULL ,  
DATE_ACHAT Date NOT NULL,  
ID_EDITEUR Varchar (25) NOT NULL,  
COTE Varchar (25) NOT NULL,  
PRIMARY KEY(CODE_BARRE),  
FOREIGN KEY(ID_EDITEUR)REFERENCES EDITEUR(ID_EDITEUR),  
FOREIGN KEY(COTE)REFERENCES OEUVRE(COTE)  
)ENGINE = INNODB;  
  
CREATE TABLE AUTEUR (  
ID_AUTEUR Varchar (25) NOT NULL ,  
NOM_AUTEUR Varchar (25) NOT NULL ,  
PRENOM_AUTEUR VARCHAR (25) NOT NULL,  
PRIMARY KEY(ID_AUTEUR)  
)ENGINE = INNODB;
```

```

CREATE TABLE PRET (
Numero_ADHERENT Varchar (25) NOT NULL ,
CODE_BARRE Varchar (25) NOT NULL ,
DATE_EMP Datetime NOT NULL,
DATE_RETOUR Date NOT NULL ,
Prolongation Bool,
PRIMARY KEY(CODE_BARRE,DATE_EMP, Numero_ADHERENT),
FOREIGN KEY(Numero_ADHERENT)REFERENCES ADHERENT(Numero_ADHERENT),
FOREIGN KEY(CODE_BARRE)REFERENCES LIVRE(CODE_BARRE)
)ENGINE = INNODB;

```

```

CREATE TABLE RESERVE (
Numero_ADHERENT Varchar (25) NOT NULL ,
COTE Varchar (25) NOT NULL ,
DATE_RESERVATION Datetime NOT NULL,
DATE_LIM Date NOT NULL ,
PRIMARY KEY(Numero_ADHERENT,COTE,DATE_RESERVATION),
FOREIGN KEY(Numero_ADHERENT)REFERENCES ADHERENT(Numero_ADHERENT),
FOREIGN KEY(COTE)REFERENCES OEUVRE(COTE)

)ENGINE = INNODB;

```

```

CREATE TABLE AUTEUR (
ID_AUTEUR Varchar (25) NOT NULL , NOM_AUTEUR Varchar (25) NOT NULL ,
PRENOM_AUTEUR VARCHAR (25) NOT NULL, PRIMARY KEY(ID_AUTEUR)
)ENGINE = INNODB;

```

```

CREATE TABLE MOT_CLE (
NUM_MOTCLE Varchar (25) NOT NULL , MOT_CLES Varchar (25) NOT NULL , PRIMARY KEY(NUM_MOTCLE)
)ENGINE = INNODB;

```

```

CREATE TABLE REDIGER (
ID_AUTEUR Varchar (25) NOT NULL ,
COTE Varchar (25) NOT NULL ,
PRIMARY KEY(ID_AUTEUR,COTE),
FOREIGN KEY(ID_AUTEUR)REFERENCES AUTEUR(ID_AUTEUR),
FOREIGN KEY(COTE)REFERENCES OEUVRE(COTE)
)ENGINE = INNODB;

```

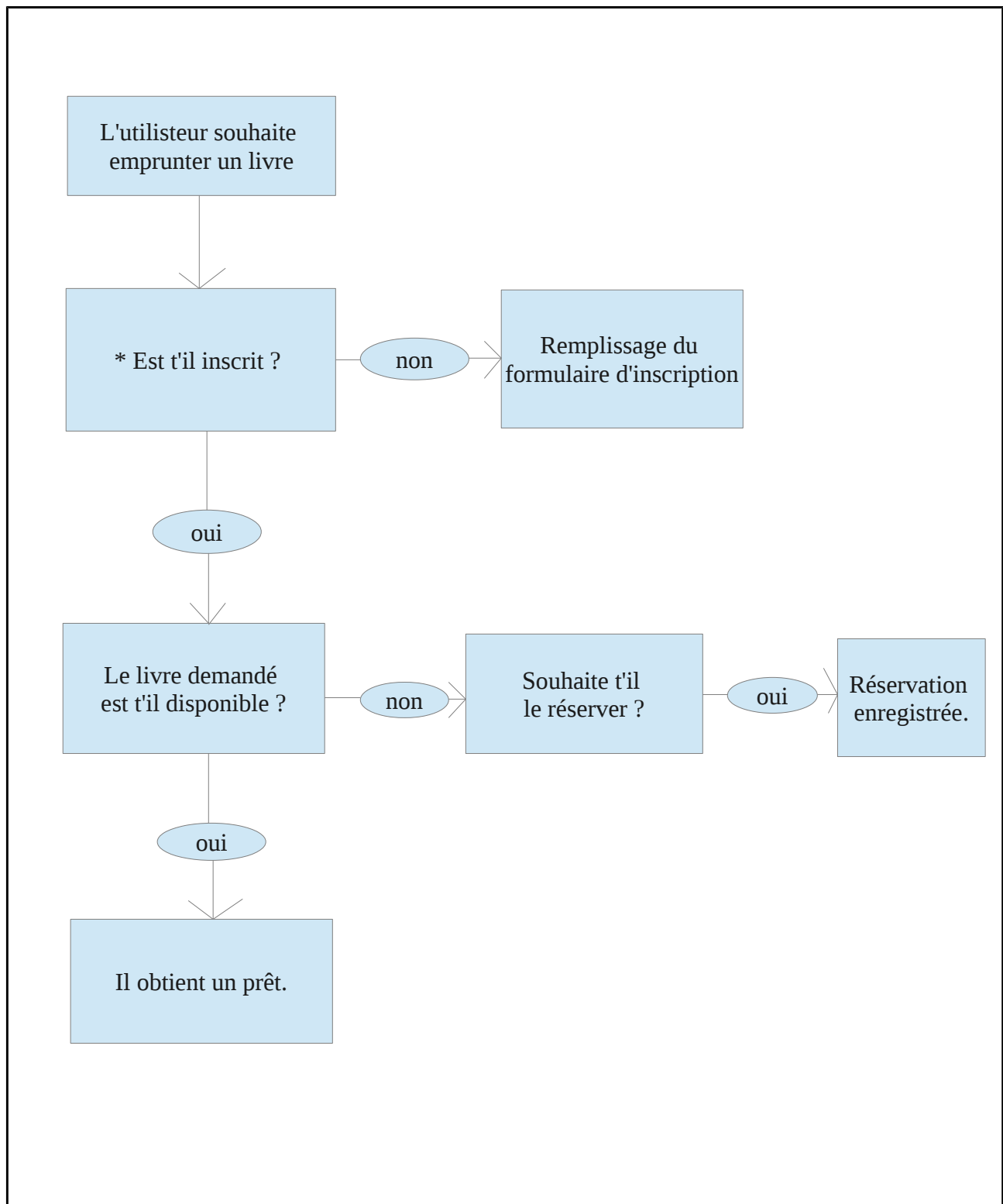
```

CREATE TABLE CARACTERISE (
NUM_MOTCLE Varchar (25) NOT NULL , COTE Varchar (25) NOT NULL ,
PRIMARY KEY(NUM_MOTCLE,COTE),
FOREIGN KEY(NUM_MOTCLE)REFERENCES MOT_CLE(NUM_MOTCLE),
FOREIGN KEY(COTE)REFERENCES OEUVRE(COTE)
)ENGINE = INNODB;

```

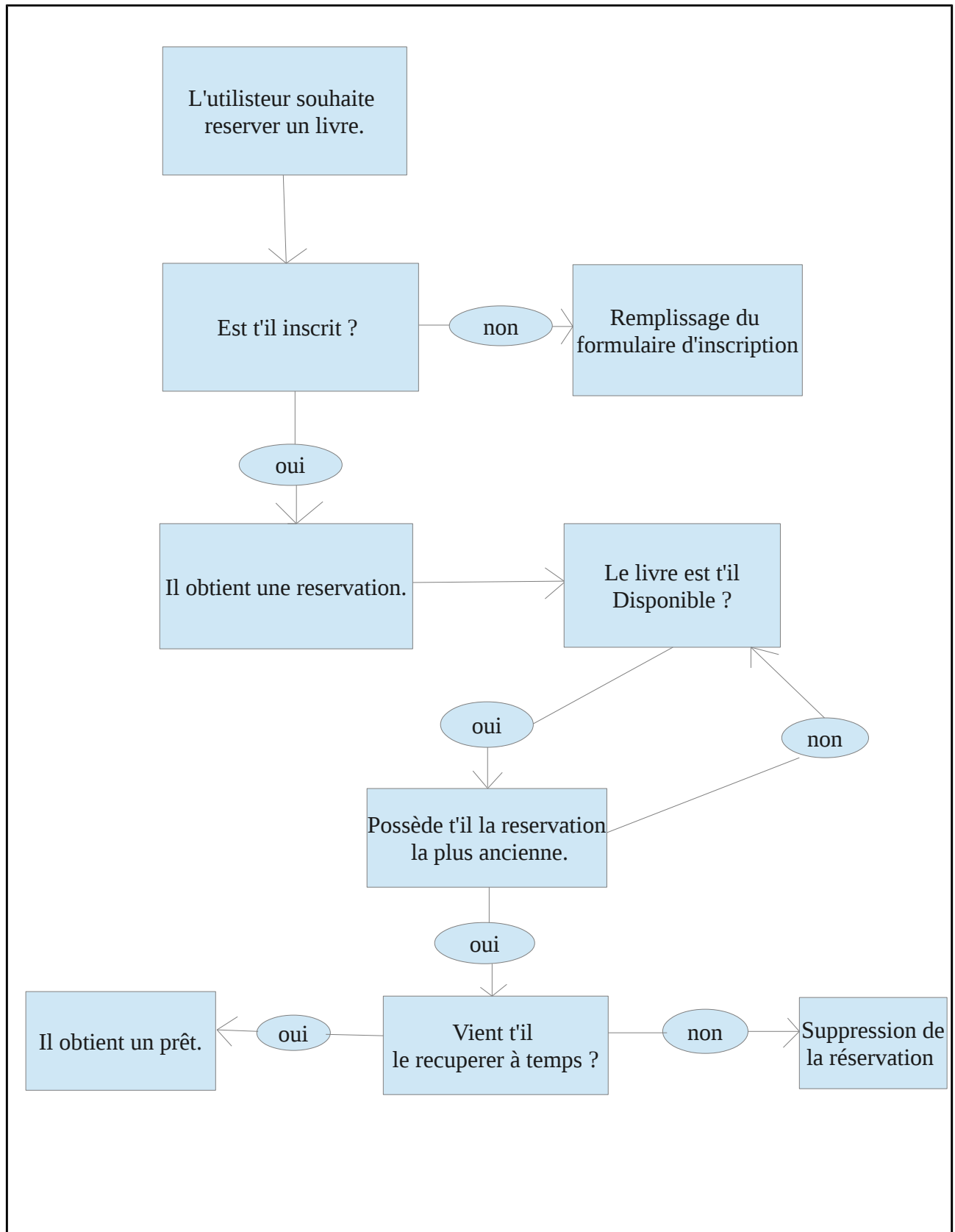
2.7 Analyse fonctionnelle

Prêt:



* Via une page d'administration, le bibliothécaire vérifie si l'utilisateur est inscrit.

Réservation:



Remarque: Les fonctionnalités très simples comme par exemple, l'inscription ou le retour d'un livre ne sont pas décrites .

III - Programmation de l'interface graphique

L'interface graphique proposé est une application web en PHP qui interagira avec une base de données MySQL. La programmation en PHP sera orienté objet pour des raisons pratiques (maintenance, facilité de lecture et de modifications, ...)

3.1 Architecture de l'application

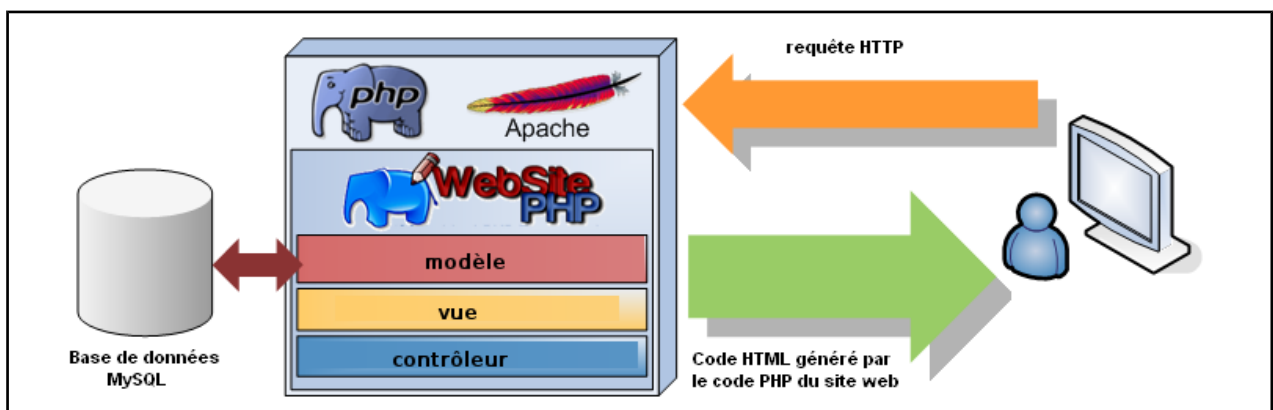
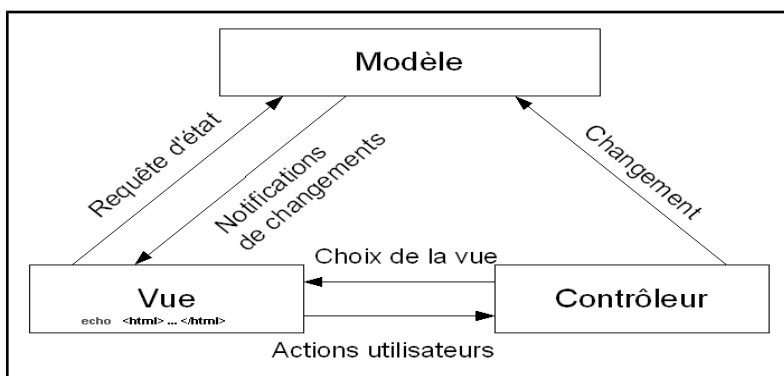
Le code source de l'application est organisé selon l'architecture **MVC** (modèle-vue-contrôleur) qui regroupe les fonctions nécessaires en trois catégories :

modèle : gère les données de l'application web. Son rôle est de récupérer les informations dans la base de données, de les organiser et de les assembler pour qu'elles puissent ensuite être traitées par le contrôleur. On y trouve donc les requêtes SQL.

Contrôleur : gère la logique du code et prend des décisions. Il est l'intermédiaire entre le modèle et la vue . Le contrôleur va demander au modèle, les données, les analyser, prendre des décisions et renvoyer le texte à afficher à la vue. Aussi il détermine si le visiteur a le droit de voir la page ou non (gestion des droits d'accès)

vue: gère l'affichage. Son rôle est de récupérer des variables pour savoir ce qu'elle doit afficher. On y trouve essentiellement du code HTML mais aussi quelques boucles et conditions PHP très simples, pour afficher par exemple la liste des adhérents ou des livres de la bibliothèque.

Voici deux schémas pour mieux comprendre le principe de cette architecture.



3.2 Code Source

Arborescence du site web :

```
bibliothèque
|
| index.php
|
+---img
|   header.jpg
|   schema.jpg
|
+---css
|   style.css
|
| \---img
|     *****.png
|
+---views
|   View.inc.php
|   ManageMemberView.inc.php
|   |
|   | *****View.inc.php
|   |
|   \---templates
|       loginform.inc.php
|       page.inc.php
|       |
|       | ***.inc.php
|       |
|
+---script
|   script.js
|   |
|   \---php
|       SearchISBN.php
|       SearchTitle.php
|       SearchPublisher.php
|
+---model
|   BooksList.inc.php
|   Book.inc.php
|   Database.inc.php
|   MembersList.inc.php
|
| \---actions
|   Action.inc.php
|   AddBookAction.inc.php
|   RegisterMemberAction.inc.php
|   |
|   | *****Action.inc.php
```

Remarque: Tous les fichiers ne sont pas représentés.

Database.inc.php contient des fonctions exécutant des requêtes SQL.

Requêtes SQL

Quelques requêtes SQL utilisé par la classe Database (database.inc.php) :

```
public function loadBookByKeyword($keyword){

    $query = $this->connection->prepare("
        SELECT ISBN, DISPONIBILITE, DATE_ACHAT, LIVRE.COTE, CODE_BARRE, TITRE,
        DATE_FORMAT(DATE_PARUTION , '%d-%m-%Y') DATE_PARUTION , NOM_AUTEUR,
        PRENOM_AUTEUR, NOM_EDIT, DATE_FORMAT(DATE_ACHAT , '%d-%m-%Y') DATE_ACHAT,
        NUMERO_EXEMPLAIRE
        FROM LIVRE, OEUVRE, AUTEUR, REDIGER,EDITEUR,CARACTERISE, MOT_CLE
        WHERE LIVRE.COTE = OEUVRE.COTE
        AND LIVRE.COTE = REDIGER.COTE
        AND AUTEUR.ID_AUTEUR = REDIGER.ID_AUTEUR
        AND LIVRE.ID_EDITEUR = EDITEUR.ID_EDITEUR
        AND CARACTERISE.NUM_MOTCLE=MOT_CLE.NUM_MOTCLE
        AND LIVRE.COTE=CARACTERISE.COTE
        AND MOT_CLES like ?");

    $sql = $query->execute(array('%'.$keyword.'%'));

    [ ... ]

}
```

Cette fonction retourne une liste de livres en fonction d'un mot clé (fonction recherche par mot clé)

```
public function checkCodeBarre($code){

    $command = "SELECT * FROM LIVRE WHERE CODE_BARRE= ?";
    $query = $this->connection->prepare($command);

    $sql = $query->execute(array($code));

    while ($row = $query->fetch(PDO::FETCH_NUM, PDO::FETCH_ORI_NEXT)) {
        return false;
    }

    return true;

}
```

Cette fonction vérifie si un code barre existe déjà dans la base de données (vérification lorsque on ajoute un nouveau livre dans la base de données)

3.3 Interface graphique

Gestion des adhérents :

ID	Nom	Prénom
81858	MDJASSIRI	AMIRDINE
64169	NARRAN	Maxime
95213	RAMBO	JOHN

Recherche

Valider

☒ Identifiant ☐ Nom

Options

Modifier Nouveau

GESTION DES ADHÉRENTS

Cette page permet de gérer tous les adhérents (modifications, création d'un adhérent, ...)

Gestion des livres:

Code	Titre	Auteur
5555	Les Miserables	HUGO VICTOR
7548	Stars Wars	HUGO VICTOR
8124	Stars Wars	HUGO VICTOR
5485	Un Mondes Parfait	HUGO VICTOR
1763	Stars Wars II	Georges Lucas
2755	Lucky Luke contre Pat Pok	De Bever Maurice

Recherche

Valider

☒ Titre ☐ Auteur ☐ Mot-clés

Options

Nouveau Retard

GESTION DES LIVRES

Cette page permet de gérer tous les livres (modifications, ajouter un nouveau livre, ...)

Gestion des prêts :

PRET D'UN LIVRE

ID ADHÉRENT :

N ° LIVRE :

Envoyer

Effacer

Cette page permet de gérer les prêts . Pour le retour des livre on a le même formulaire.

Gestion des réservations :

Reservation

ID ADHÉRENT :

COTE DU LIVRE:

Envoyer

Effacer

Liste des réservations

Numéro Adhérent	Nom	Prénom	Titre	Cote	Date réservation	Date limite
81858	MDJASSIRI	AMIRDINE	Les Miserables	R100	29-04-2014	07-05-2014

Cette page permet de faire et de lire des réservations

IV – Conclusion

Lors de ce projet, nous avons pu répondre aux exigences minimales du cahier des charges. Malheureusement, cette application n'est pas assez robuste. C'est-à-dire que le client doit remplir les champs en évitant de faire des erreurs de saisies (exemple: oublier de remplir un champs), afin de ne pas provoquer des dysfonctionnements.

Ce projet, nous a permis de revoir et approfondir nos connaissances en base de données. Et parfois de découvrir des choses comme par exemple les différences entre Oracle et MySQL, ou l'utilisation des rollbacks qui pouvaient être très utile pour notre application.

La plus grande difficulté était la conception et la programmation de l'interface graphique qui assez long à faire. De plus, il faut prévoir tous les cas possibles.