

2012



ANTOINE Sabri,  
BELLUOT Vincent,  
BENYIAT Mohamed,  
GOURRAUD Nicolas

## [GENIE LOGICIEL]

Il s'agit de refaire la gestion de la bibliothèque de notre école pour que les professeurs et les élèves puissent emprunter des documents de l'EISTI

## Table des matières

Résumé du précédent livrable.....	3
Requêtes SQL.....	5
Conclusion.....	8

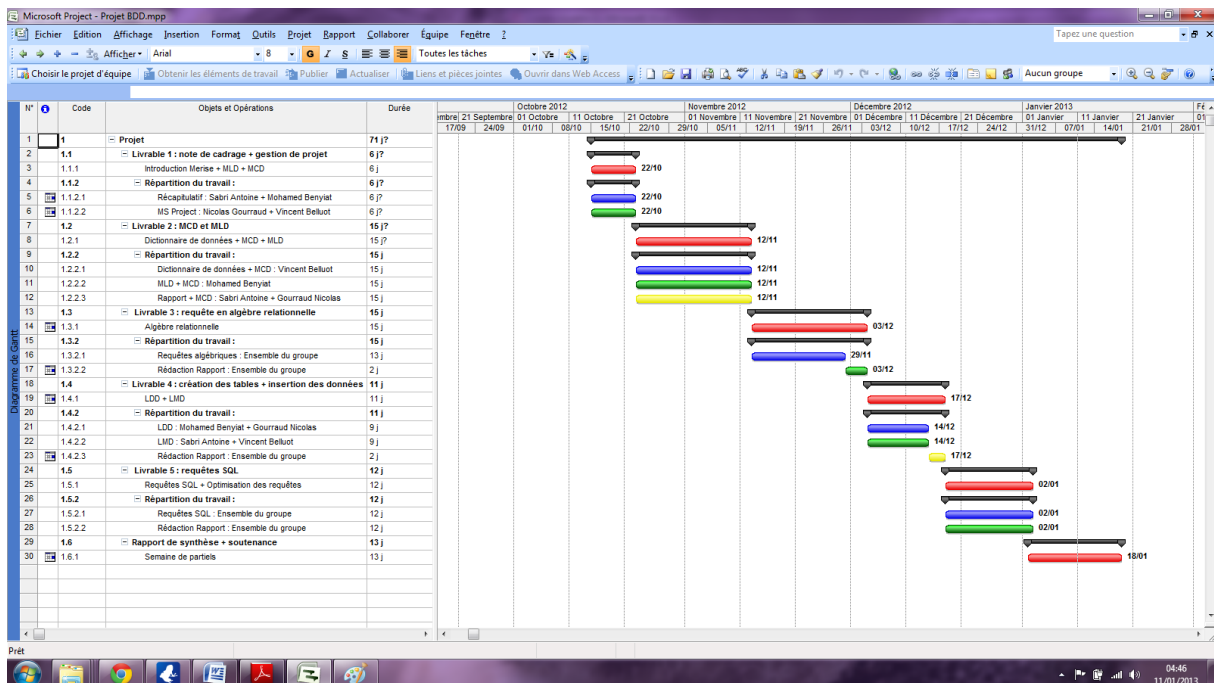
## 1) Résumé du précédent livrable

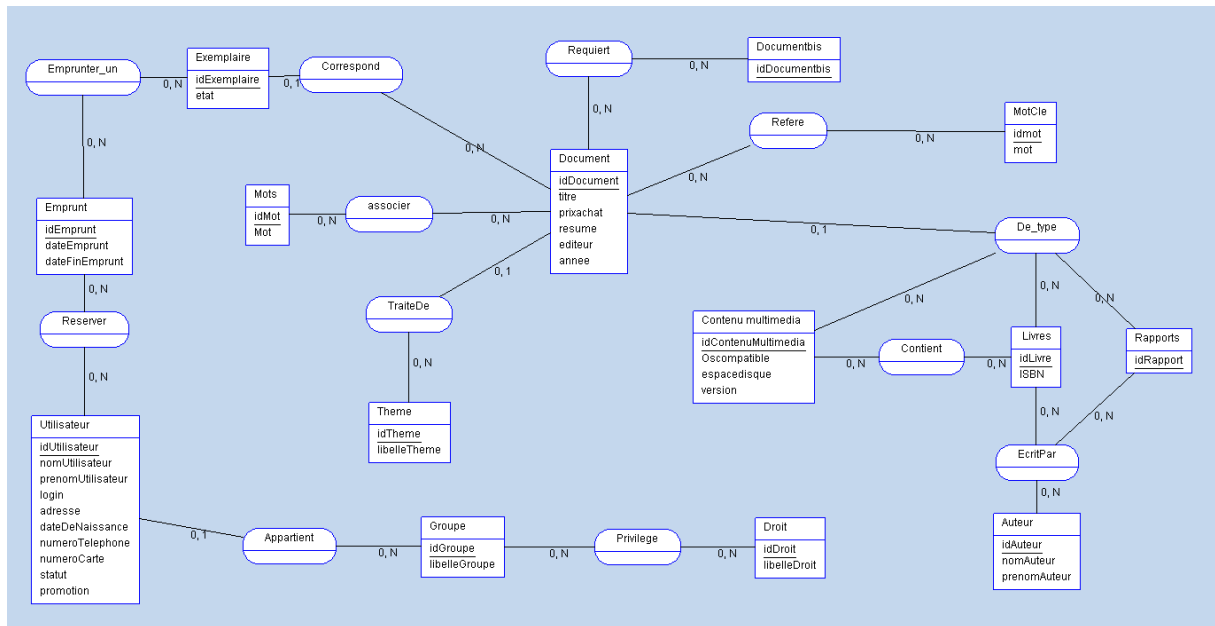
L'objectif de ce projet est la création d'une base de données permettant une gestion efficace de la bibliothèque de l'EISTI. Dans ce projet nous appliquerons nos connaissances acquises pendant le semestre. La portée du projet part du début de sa conception jusqu'à la réalisation de la base de données ainsi que des requêtes SQL permettant son traitement ainsi que son analyse.

Pour le premier livrable nous avons expliqué les fonctionnalités du programme ainsi que le planning en utilisant la méthode des SIXO. Dans le deuxième livrable, nous devions faire le Modèle Conceptuel des Données (MCD), puis le Modèle de Logique des Données (MLD) ainsi qu'une bibliothèque de données. Tout ceci afin de représenter la structure du système d'information, du point de vue des données et aussi pour définir les dépendances ou relations entre ces différentes données. Ensuite dans un troisième livrable, nous avons travaillé avec l'algèbre relationnelle afin de représenter des requêtes sur la base de données dont le résultat s'exprime sous la forme d'une relation. Enfin dans le 4<sup>ème</sup>, nous avons travaillé sur le langage de définition des données (LDD) et le langage de manipulation des données (LMD).

Pour ce 5<sup>ème</sup> livrable, nous devons transcrire les requêtes que nous avons faites avec l'algèbre relationnelle dans le 3<sup>ème</sup> livrable, en SQL. Nous avons donc commencé à coder notre projet ce qui nous a permis de voir les choses plus concrètement.

Voici notre MCD et notre MS Project mis à jour :





## 2) Requêtes SQL

Nous allons travailler ici sur le langage SQL afin de traduire les requêtes en algèbre relationnelle que nous avons faites lors du livrable 3. C'est un langage complet de gestion de bases de données relationnelles (nous avons commencé à l'étudier lors du livrable 4) et c'est à la fois :

- Un langage d'interrogation de la base (**SELECT**)
- Un langage de manipulation des données (**UPDATE, INSERT, DELETE**)
- Un langage de définition des données (**CREATE, ALTER, DROP**)

Voici nos différentes requêtes SQL :

**Themdoc** liste les documents en fonction du thème, c'est-à-dire si c'est un policier ou un roman

```
SELECT titre, libelleTheme
FROM Document d, Theme t
WHERE d.idTheme_c5 = t.idTheme
ORDER BY libelleTheme asc;
```

**UtiliEmprunt** donne tous les utilisateurs avec les documents qu'ils ont empruntés

```
SELECT nomUtilisateur, prenomUtilisateur, titre
FROM Utilisateur u, Reserver r, Emprunt t, Emprunter_un m, Exempleire e,
Document d
WHERE u.idUtilisateur = r.idUtilisateur_c1
AND r.idEmprunt_c1 = t.idEmprunt
AND t.idEmprunt = m.idEmprunt_c2
AND m.idExempleire_c1 = e.idExempleire
AND e.idDocument = d.idDocument
GROUP BY nomUtilisateur;
```

**UtiliGroupe** donne tous les groupes et utilisateurs qui sont dedans

```
SELECT nomUtilisateur, prenomUtilisateur, libelleGroupe
FROM Groupe g, Utilisateur u
WHERE g.idGroupe = u.idGroupe_c1
GROUP BY libelleGroupe;
```

**ListeDocEmprunt** donne la liste de tous les documents empruntés. De plus, il met cette liste par ordre alphabétique.

```
SELECT Titre
FROM Document d, Exempleire e, Emprunt m, Emprunter_un p
WHERE d.idDocument = e.idDocument_c5
AND e.idExempleire = p.idExempleire_c1
AND p.idEmprunt_c2 = m.idEmprunt
ORDER BY titre asc;
```

**PrivGroupe liste tous les privilèges de chaque groupe. Pour cela, on est obligé de savoir quel groupe à quoi comme droit.**

```
SELECT libelleGroupe, libelleDroit
FROM Groupe g, Droit d, Privilege p
WHERE g.idGroupe = p.idGroupe_c1
AND p.idDroit_c1 = d.idDroit
GROUP BY libelleGroupe;
```

**InfoDoc donne chaque information sur un document recherché. Il donne le titre, le prix, donne un résumé, l'éditeur, l'année.**

```
SELECT Titre, PrixAchat, Resume, Editeur, Annee
FROM Document
WHERE titre = 'titreDocumentRchercher';
```

**Requête qui permet de compter le nombre d'exemplaire disponible pour 1 document (savoir s'il il est disponible). On peut retirer le COUNT s'il on veut avoir précisément les id des exemplaires.**

```
SELECT COUNT(*) FROM Exempleire
WHERE etat=0
AND idDocument_c5='2'
```

**Permet de déterminer tous les emprunts effectués par l'utilisateur. Pour les classés dans le temps on pourra y appliquer un filtre en fonction de la date actuelle.**

```
SELECT idEmprunt_c1 FROM RESERVER
WHERE idUtilisateur_c1='4'
```

**Permet de savoir à qui correspond un emprunt, à quel utilisateur (son id)**

```
SELECT idUtilisateur_c1 FROM Reserver
WHERE idUtilisateur_c1='2'
```

**Requête qui permet de déterminer le type de document. Si un des 3 id retourné vaut 0, alors on peut le considéré comme faux. Un document peut ainsi être de plusieurs types à la fois (s'il possède un livre et un dvd par exemple).**

```
SELECT IDCONTENUMULTIMEDIA_C1, IDLIVRE_C1, IDRAPPORT_C1 FROM De_Type
WHERE iddocument_c1='5'
```

**Ce Select nous permet de sélectionner tous les documents correspondant à un thème**

```
SELECT titre FROM Document d,Theme t
WHERE d.idtheme_c5=t.idtheme
AND t.libelletheme='divers'
```

**Cherche un document selon un mot clé.**

```
SELECT d.idDocument FROM Document d,MotCle m,Refere r
WHERE m.mot LIKE 'maths'
AND m.idMotCle=r.idmot_c2
AND r.idDocument_c3=d.idDocument
```

**Filtre un document en fonction de l'année**

```
SELECT * FROM Document WHERE annee=1996
```

**Filtre un document en fonction de l'éditeur**

```
SELECT * FROM Document WHERE editeur='EDITION FRANCE'
```

**Filtre un document en fonction du titre**

```
SELECT * FROM Document WHERE titre='Le Jeu'
```

**Requête qui détermine les dépendances entre les documents**

```
SELECT * FROM Document
WHERE idDocument IN (SELECT idDocumentpere FROM Requier WHERE
idDocumentfils = 13)
```

Pour utiliser ces requêtes, il faut ouvrir le dossier scriptLIV4 qui accompagne ce livrable. Puis il faut exécuter le script *creationbase*, puis *insertion\_1*. Nous pouvons ainsi tester nos différentes requêtes.

### 3) Conclusion

Ce livrable nous a permis de bien avancer dans la construction de notre projet et notamment au niveau du codage des requêtes SQL. Nous pouvons donc maintenant manipuler et retrouver des données enregistrées dans notre base de données. Nous allons consacrer le temps qu'il nous reste à optimiser ses requêtes et pourquoi pas étoffer notre base de données.