

# DOSSIER DE CONCEPTION & DÉPLOIEMENT

## Projet : Réseau de Partage de Livres (UTC504 / NFP107)

Étudiant : BONNETON Hugo & NEWTON Atheo Année : 2025-2026

---

### 1. Introduction

Ce document présente la conception technique et fonctionnelle de l'application web de **Réseau de Partage de Livres**. Ce projet a pour objectif de permettre à des cercles d'amis de référencer leurs bibliothèques personnelles, de gérer la localisation de leurs ouvrages et de faciliter les emprunts au sein du groupe.

L'application a été conçue en respectant les principes du **"Minimal Design"**, avec une interface immersive ("Dark Academia") et une architecture robuste reposant sur une base de données **Oracle**.

**Oracle Database** est un [système de gestion de base de données relationnelle](#) (SGBDR) qui depuis l'introduction du support du [modèle objet](#) dans sa version 8 peut être aussi qualifié de [système de gestion de base de données relationnel-objet](#) (SGBDRO). Fourni par [Oracle Corporation](#), il a été développé par [Larry Ellison](#), accompagné entre autres, de [Bob Miner](#) et [Ed Oates](#).

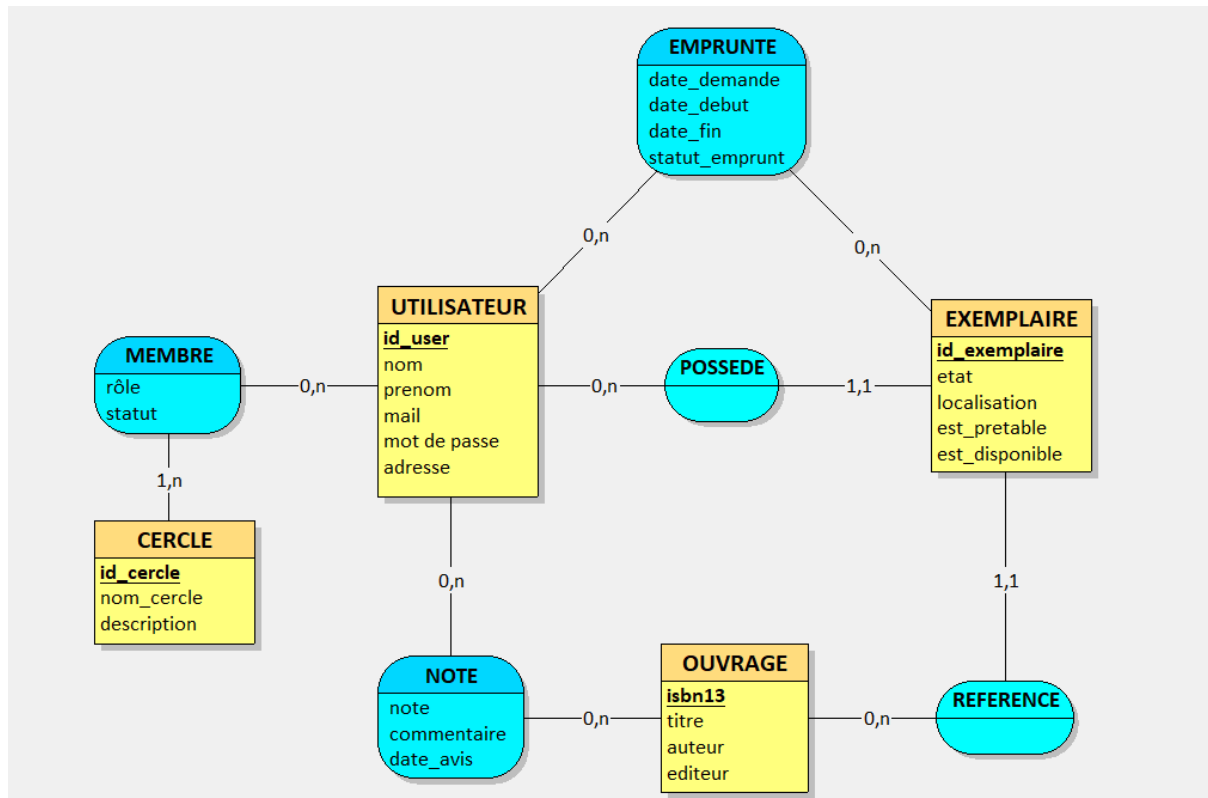
---

### 2. Conception des Données (Base de Données)

La structure des données a été pensée pour distinguer clairement l'œuvre abstraite (Livre) de l'objet physique (Exemplaire).

## 2.1. Modèle Conceptuel de Données (MCD)

Le schéma ci-dessous illustre les relations entre les entités principales : Utilisateurs, Cercles, Ouvrages et Exemplaires.



## 2.2. Modèle Logique de Données (MLD)

La traduction du MCD en modèle relationnel est la suivante:

**UTILISATEUR** (#id\_user, nom, prenom, mail, mot\_de\_passe, adresse)

**CERCLE** (#id\_cercle, nom\_cercle, description)

**OUVRAGE** (<ins>isbn13</ins>, titre, auteur, editeur)

**EXEMPLAIRE** (#id\_exemplaire, etat, localisation, est\_pretable, est\_disponible, #id\_user, #isbn13)

- #id\_user référençant UTILISATEUR(id\_user)
- #isbn13 référençant OUVRAGE(isbn13)

**MEMBRE** (#id\_user, #id\_cercle, rôle, statut)

- #id\_user référençant UTILISATEUR(id\_user)
- #id\_cercle référençant CERCLE(id\_cercle)

**EMPRUNT** (#id\_user, #id\_exemplaire, date\_demande, date\_debut, date\_fin, statut\_emprunt)

- #id\_user référençant *UTILISATEUR*(id\_user)
- #id\_exemplaire référençant *EXEMPLAIRE*(id\_exemplaire)

**AVIS** (#id\_user, #isbn13, note, commentaire, date\_avis)

- #id\_user référençant *UTILISATEUR*(id\_user)
- #isbn13 référençant *OUVRAGE*(isbn13)

## 2.3. Logique Métier (PL/SQL)

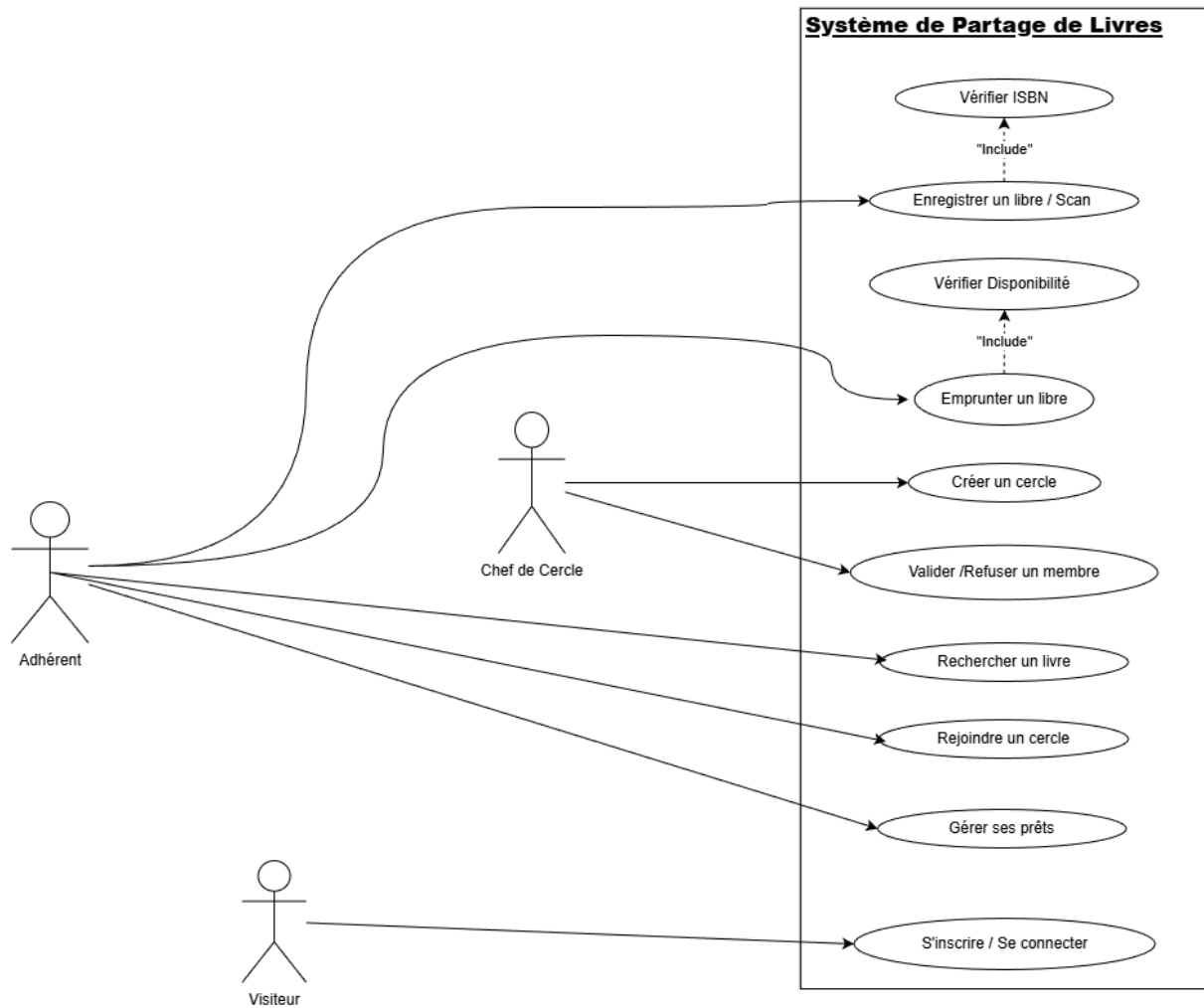
Afin d'assurer l'intégrité et la performance des données directement dans le SGBD, les éléments suivants ont été implémentés :

- **Trigger TRG\_MAJ\_DISP0** : Automatise la gestion des stocks. Dès qu'un emprunt est validé (date de début renseignée), l'exemplaire concerné passe automatiquement en statut "Indisponible" (*est\_disponible* = 0) dans la table *T\_EXEMPLAIRE*.
  - **Vue V\_STATISTIQUES** : Centralise les indicateurs clés (Nombre d'utilisateurs, Livres uniques, Prêts actifs) pour alimenter le tableau de bord en temps réel.
  - **Procédure AJOUTER\_USER** : Permet l'insertion sécurisée et standardisée des nouveaux inscrits.
- 

## 3. Conception Fonctionnelle

### 3.1. Diagramme des Cas d'Utilisation (Use Case)

Le système gère trois niveaux d'acteurs : le Visiteur, l'Adhérent et le Chef de Cercle.



### 3.2. Priorisation des fonctionnalités

Conformément au cahier des charges, les fonctionnalités ont été développées dans cet ordre de priorité :

1. **Gestion de l'inventaire (Priorité Haute)** : Ajout de livre par scan ISBN, localisation physique (Grenier, Salon...).
2. **Gestion des Cercles (Priorité Moyenne)** : Création de groupes, adhésion.
3. **Gestion des Prêts (Priorité Moyenne)** : Demande et validation d'emprunt.

## 4. Interface & Expérience Utilisateur

L'interface graphique adopte un style "Bibliothèque Classique" (tons sombres, bois, doré) pour une expérience utilisateur immersive.

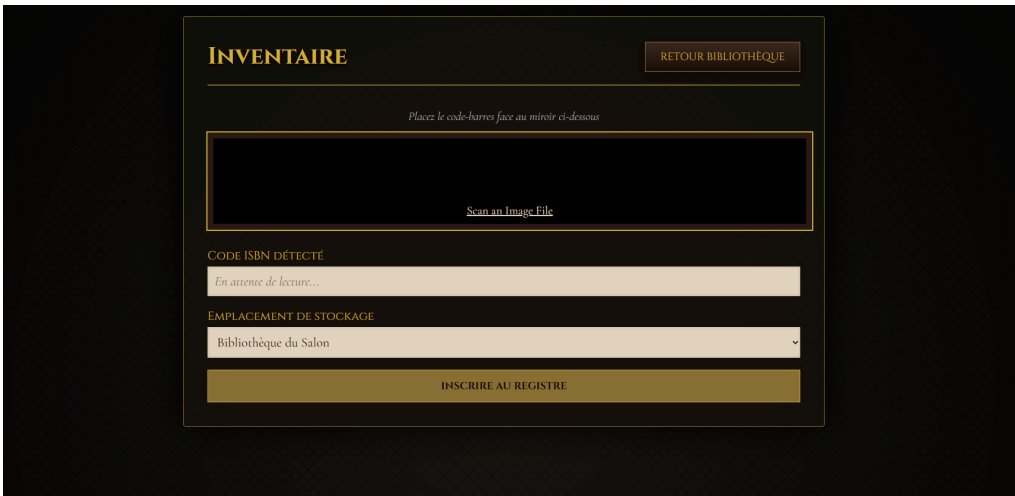
### 4.1. Tableau de Bord (Accueil)

La page d'accueil présente les statistiques globales via la Vue SQL et la liste des livres sous forme de cartes ("Grid Layout"). Des badges visuels (Vert/Rouge) indiquent instantanément la disponibilité.



### 4.2. Module de Scan (Inventaire)

L'ajout de livre utilise la caméra de l'appareil (PC ou Smartphone) pour scanner le code-barres. L'utilisateur précise ensuite la localisation physique de l'ouvrage via un menu déroulant dynamique.



## 5. Procédure de Déploiement

Cette section détaille la mise en place de l'application sur le serveur de l'école.

### Prérequis

- Accès au serveur Oracle de l'établissement.
- Accès FTP au serveur Web (Apache/Nginx/PHP).
- Client SQL (SQL Developer) et Client FTP (FileZilla).

### Étape 1 : Initialisation de la Base de Données

1. Ouvrir le fichier `db_projet.sql`.
2. Se connecter à la base Oracle via SQL Developer.
3. Exécuter le script complet (F5).
  - *Résultat attendu* : Création des tables `T_...`, du trigger `TRG_MAJ_DISPO` et de la vue `V_STATISTIQUES`.

### Étape 2 : Configuration de l'Application Web

1. Ouvrir le fichier `db.php` avec un éditeur de texte.

Modifier les variables de connexion avec les identifiants fournis par les enseignants :

PHP

```
$host = 'serveur-etudiant.univ.fr';  
$user = 'VOTRE_LOGIN_ORACLE';  
$pass = 'VOTRE_MOT_DE_PASSE';
```

- 2.

### Étape 3 : Mise en ligne

1. Transférer les fichiers suivants dans le dossier `public_html` (ou `www`) du serveur via FTP :
  - `index.php` (Page principale)
  - `scan.php` (Module d'inventaire)
  - `style.css` (Feuille de style)
  - `db.php` (Fichier de configuration)
2. L'application est désormais accessible à l'adresse :  
`http://100.95.181.113/index.php`

---

## 6. Conclusion

Ce projet a permis de mettre en œuvre une architecture complète n-tiers. La séparation stricte entre la logique de données (Oracle), la logique applicative (PHP) et l'interface (CSS/JS) garantit la maintenabilité et l'évolutivité de la solution, répondant ainsi aux exigences du cahier des charges.