

LE PROJET

Projet OLibrary

**Module de référence : Merise/SQL – UML/POO – HTML & CSS – Ergonomie – PHP – Javascript
– Linux**

1. Mission

Création d'un portail de gestion de fonds documentaires et son catalogue public.

Vous devez réaliser un SIGB (Système Intégré de Gestion de Bibliothèque) et son OPAC (Open Public Access Catalog – Catalogue Public).

2. Enjeux Objectifs

- Mettre en application les connaissances de différentes matières
- Etudier un cahier des charges pour en faire ressortir un produit fini
- Respecter des contraintes fonctionnelles, techniques et temporelles

3. Règles du jeu

Cahier des charges global

Les fonctions à intégrer au système sont libres mais celles listées ci-après sont obligatoires :

SIGB

- Espace sécurisé par un couple utilisateur/mot de passe
- Les utilisateurs doivent être administrables depuis l'interface
- Gestion de notices et des exemplaires associés
- Gestion des autorités : auteurs, éditeurs, fournisseurs, collections, ...
- Gestion des emprunteurs
- Gestion de la circulation prêts, retours, retards, rappels, historique, réservation, ...
- Création d'un échantillon suffisant de données pour les tests (cf. Ressources en ligne)

OPAC

- Consultation du fonds documentaire
- Recherche
- Bonus : espace « adhérents » (prêts en cours, réservation, profil, ...)

LE PROJET

Le portail de gestion de fonds documentaires et son catalogue public devront être hébergés sur un serveur virtualisé Linux avec les outils nécessaires au développement de l'application et à la gestion de données.

Le travail devra s'effectuer par **groupes de 4 à 5 étudiants**. Les groupes définis en début de projet resteront inchangés jusqu'au rendu final.

Le projet sera divisé en **trois phases** :

- **Réponse Appel d'Offre : Devis, cahier des charges**
 - o **Lancement de la phase** : lundi 3 octobre par Sébastien VITA
 - o **Fin de la phase** : lundi 14 novembre
 - o **Objectifs** :
 - Appréhender le projet et les différentes matières attenantes
 - Comprendre le besoin pour en cadrer les fonctionnalités
 - Comprendre les différents aspects d'un projet
 - o **Ce qu'il faudra faire** :
 - Rédaction d'une réponse à un l'appel d'offre (RAO)
 - Chiffrage du projet et réalisation d'un devis
- **Analyse et conception**
 - o **Lancement de la phase** : lundi 21 novembre par Sébastien VITA
 - o **Fin de la phase** : vendredi 20 janvier
 - o **Objectifs** :
 - Identifier les entités
 - Identifier les relations entre les entités
 - Gérer les rôles
 - Modéliser les classes et la base de données
 - o **Ce qu'il faudra faire** :
 - Réaliser la modélisation Merise du projet
 - Rédiger le MLD, MCD et MPD
 - Créer les différents objets de la base de données
 - Planifier les phases de développement
- **Développement et déploiement**
 - o **Lancement de la phase** : vendredi 20 janvier par Bertrand HUBERT
 - o **Fin de la phase** : lundi 15 mai
 - o **Objectifs** :
 - Mettre en pratique les langages Web
 - Administrer un serveur
 - Migrer d'une solution locale vers une solution à distance
 - o **Ce qu'il faudra faire** :
 - Développer le système en utilisant les langages HTML, CSS et PHP
 - Utiliser au moins un des langages suivants : jQuery, JavaScript, Ajax, fonctionnalités particulières de HTML5.
 - Utiliser un framework CSS

- Monter un serveur Web
- Basculer son système local sur l'environnement virtualisé

4. Livrables

L'intégralité des développements et des documentations seront à centraliser dans un projet Git, de manière régulière. Chaque étudiant du groupe devra participer à la mise en ligne des ressources (chaque action est nominative et horodatée).

Avant le début du projet, vous devrez :

- Créer un compte sur Github (<https://github.com/>)
- Activer une offre « Student Pack » (<https://education.github.com/>)

A la fin de chaque phase, un livrable sera attendu :

- **Réponse Appel d'Offre :**
 - **Date de rendu :** Lundi 14 novembre, 23 heures 59
 - **Une section Wiki** dans votre projet Git contenant :
 - Le devis du projet
 - L'analyse fonctionnelle
- **Analyse et conception :**
 - **Date de rendu :** Lundi 20 janvier, 23 heures 59
 - **Une section Wiki** contenant :
 - La modélisation Merise MLD
 - La modélisation Merise MCD
 - La modélisation Merise MPD
 - Ensemble des ordres SQL (page wiki ou fichier versionné sur Git)
 - Bonus : ordres SQL adaptés pour la création d'objets équivalents sur une base de données Oracle
 - La modélisation des classes
 - Le rétro-planning de la phase 3
- **Développement et déploiement :**
 - **Date de rendu :** Lundi 15 mai, 23 heures 59
 - **Un ensemble de ressources** comprenant :
 - L'ensemble des rendus intermédiaires
 - Une documentation d'installation du serveur Web (cf. Annexe cahier des charges et les travaux à réaliser)
 - Le code source de votre projet
 - Les informations d'accès au serveur hébergeant votre solution

5. Évaluation

Chaque livrable fera l'objet d'une évaluation.

Une **soutenance orale** de fin de projet sera organisée.

- **Durée de l'oral** : 20 minutes
- **Durée des questions-réponse** : 5 minutes
- **Débriefing** : 5 minutes

Votre oral devra être appuyé par le support visuel de type PowerPoint.

Durant la soutenance orale, voici les critères qui seront évalués :

Critère	Barème
Support de présentation	/5
<i>Qualité du support (ppt), structure, plan, illustration, gestion du temps</i>	
Aisance orale	/5
<i>Expression orale, présentation et attitude, mise en scène, originalité, répartition des temps de parole</i>	
Transmission du message	/5
<i>Organisation du projet, capacité à vulgariser la technique, pertinence des technologies, démonstrations, schémas</i>	
Questions-réponses	/5
<i>Pertinence, apport des réponses, gestion du stress</i>	
NOTE	/20

6. Ressources en lignes pour aller plus loin !

- Pour se familiariser avec **SIGB** et **OPAC**
 - o PMB : <http://www.sigb.net>
 - o Koha : <http://www.koha-fr.org>
- Création d'un **jeu de données pour les tests** :
 - o <http://catalogue.bnf.fr>
 - o <http://data.bnf.fr>

7. Annexes

7.1 Cahier des charges du projet

Réponse à l'appel d'offre

Vous commencerez par rédiger une réponse à l'appel d'offre (RAO) avec les diverses fonctionnalités que vous prévoyez. Ce document devra être validé par le maître d'œuvre (MOA) avant de commencer l'analyse et le développement.

Modélisation

Vous devez réaliser la modélisation Merise de votre projet en rédigeant le MLD, le MCD, puis le MPD. Vous devrez atteindre au minimum la troisième forme normale, voire la forme de Boyce-Codd. Les formes normales suivantes n'ont pas d'intérêt pour ce projet.

Une fois votre modélisation terminée, vous pourrez créer les différents objets de la base de données. Pour cela, vous n'utiliserez pas l'interface graphique phpMyAdmin, mais des scripts SQL. L'ensemble des ordres SQL (création de(s) l'utilisateur(s), base(s) de données, table(s), vue(s), ...) devra être consigné dans un fichier SQL. L'exécution de ce seul fichier devra créer l'ensemble des objets sur le serveur MySQL.

Bonus :

Un second fichier SQL adapté pour la création des objets équivalents sur une base de données Oracle.

Installation d'un serveur Web

Vous allez utiliser **une machine virtuelle** (réalisée avec VirtualBox ou VMWare) sur laquelle vous installerez un système d'exploitation, le serveur Web Apache, l'interpréteur PHP, le système de base de données MySQL, ainsi que divers outils utiles voire indispensables sur un serveur Web.

N.B. : Les logiciels tout-en-un de type "LAMP" ne sont pas autorisés.

Vous choisirez un **système d'exploitation libre et gratuit** de type Unix (tel qu'une distribution Linux comme Debian, CentOS, Mandriva, openSUSE, ...).

Travail à réaliser : Expliquez de manière technique votre choix de distribution en réalisant un tableau comparatif des critères que vous aurez choisi.

Lors de l'installation, vous choisirez un partitionnement manuel du disque afin de séparer les points de montage "/" et "/home" ainsi que l'espace d'échange ("swap").

Travail à réaliser : Faites une représentation graphique du partitionnement de votre disque en expliquant vos choix (taille des partitions, système de fichiers, ...).

N.B. : Certaines distributions proposent d'installer des ensembles de paquets préconfigurés (comme par exemple "Serveur Web"). Vous n'utiliserez pas cette fonctionnalité. Vous n'installerez pas l'environnement graphique.

LE PROJET

Vous allez ensuite **installer tous les logiciels nécessaires à votre serveur Web**.

Travail à réaliser : Décrivez toutes les étapes que vous avez suivies pour réussir votre installation, ainsi que les tests que vous avez effectués.

Liste non exhaustive des applications à installer et configurer :

- Serveur Web : Apache2
- Interpréteur : PHP5
- Base de données : MySQL
- Serveur FTP (exemple : proftpd)
- Serveur mail (exemple : postfix)

À chaque étape, vous créerez les utilisateurs supplémentaires nécessaires. Afin de commencer à sécuriser votre serveur, **vous installerez et configurerez sudo** avant de **désactiver les connexions à SSH avec l'utilisateur "root"**. Vous créerez également un utilisateur "olibrary" avec un espace personnel dans /home.

Apache sera configuré avec des **VirtualHosts**. L'un d'eux pointera sur le dossier de votre application OLibrary (/home/olibrary/www).

Travail à réaliser : Vous rédigerez une documentation expliquant la mise en place des VirtualHosts et leur intérêt, ainsi que les différents principaux paramètres ; une seconde documentation portera sur la gestion des utilisateurs Unix, les groupes et les droits.

Créez également un utilisateur "phpmyadmin" afin de télécharger (en ligne de commande uniquement, sans gestionnaire de paquets) phpMyAdmin dans le dossier de votre utilisateur. Vous installerez et configurerez l'application dans le dossier /home/phpmyadmin/www. Créez le VirtualHost Apache adapté.

Votre client FTP doit être couplé à MySQL afin de pouvoir gérer les utilisateurs et dossiers FTP directement grâce à la base de données. Vous vérifierez que vos utilisateurs FTP sont limités aux dossiers et sous-dossiers choisis (ils ne doivent pas pouvoir remonter la structure des dossiers).

À cette étape, vous devriez avoir deux bases de données MySQL (phpmyadmin et proftpd).

Travail à réaliser : Comparez les fonctionnalités de MySQL et Oracle en termes d'utilisateurs, schéma, bases de données, privilèges.

Réalisez ensuite les schémas de chaque base de données (MPD) et remontez jusqu'au MLD.

Bonus :

Afin de sécuriser votre serveur, cherchez des bonnes pratiques, ou programmes (tels que Fail2ban) qui pourraient être utiles et mettez-les en place.

Travail à réaliser : Chaque action doit être commentée dans la documentation.

Développement de l'application

Vous devez vous imprégner des différents termes utilisés (notice, document, exemplaire, périodique, catalogage, paniers, circulation, prêt, réservation, auteur, éditeur, collection...). Les fonctions à intégrer sont libres, mais celles listées ci-après sont obligatoires.

SIGB

- Espace sécurisé par un couple utilisateur/mot de passe
- Les utilisateurs doivent être administrables depuis l'interface
- Gestion* de notices* et des exemplaires* associés
- Gestion* des autorités : auteurs*, éditeurs*, fournisseurs*, collections*, ...
- Gestion des emprunteurs*
- Gestion de la circulation (prêts, retours, retards, rappels, historique, réservation, ...)
- Création d'un échantillon suffisant de données pour les tests (cf. Ressources en ligne)

OPAC

- Consultation du fond documentaire
- Recherche
- Bonus : espace « adhérents » (prêts en cours, réservation, profil, ...)

Le SIGB et l'OPAC devront être développés en PHP. Vous utiliserez également les langages HTML, CSS et au moins un des langages suivants : jQuery, javascript, AJAX, fonctionnalités particulières de HTML5. L'utilisation d'un framework CSS comme bootstrap est recommandée.

L'ensemble du projet devra être déposé, au fur et à mesure de l'avancer, dans un projet Git.

LE PROJET

7.2 Annexe 2 : Planning du projet

ingésup Bachelor 1 - Projet Transversal & Suivi

