

# LIBS PROJECT

*Thierry Koulbanis & Liliana Kolmakova*

## Cahier des charges

### 1. Objectif du projet

**libsproject.ch** a pour but de créer une application web simple et pratique qui permet à chacun de gérer sa propre bibliothèque en ligne.

L'idée est d'offrir un espace personnel où l'utilisateur peut ajouter ses livres, suivre ses lectures, laisser des notes ou des commentaires, et organiser sa collection selon ses envies.

L'application sera intuitive, accessible depuis Internet, et développée avec PHP et MySQL.

### 2. Description générale

Le site proposera une interface claire avec deux types d'accès :

- un espace public (pour la présentation du projet, l'inscription et la connexion)
- un espace privé (réservé aux utilisateurs connectés)

Deux rôles d'utilisateur seront prévus :

- **Utilisateur standard**, qui gère uniquement sa propre bibliothèque
- **Administrateur**, qui peut en plus gérer les comptes utilisateurs et modérer le contenu

### 3. Fonctionnalités principales

#### Gestion des utilisateurs

- Création d'un compte via un formulaire d'inscription
- Envoi automatique d'un e-mail de confirmation après la création du compte
- Connexion et déconnexion avec gestion de session
- Sécurité assurée avec des mots de passe hashés
- Gestion de deux rôles : utilisateur et administrateur

#### Gestion des livres

- Ajouter un livre avec ses informations : titre, auteur, genre, année, ISBN (optionnel)
- Modifier ou supprimer un livre existant
- Afficher la liste de ses livres
- Donner une note de 1 à 10

### 4. Fonctionnalités optionnelles (si le temps le permet)

- Ajouter une image de couverture personnalisée
- Possibilité de scanner le ISBN

## 5. Structure des pages

Type	Page	Description
Publique	Accueil	Présente le projet et ses objectifs, avec accès à l'inscription et à la connexion
Publique	Inscription	Formulaire de création de compte
Publique	Connexion	Formulaire d'accès à son compte
Privée	Tableau de bord	Vue d'ensemble de la bibliothèque personnelle
Privée	Mes livres	Liste complète avec recherche et tri
Privée	Ajouter un livre	Formulaire d'ajout de nouveau livre
Privée	Détails du livre	Informations complètes, note et commentaire, options de modification
Privée	Administration	Espace réservé à l'administrateur pour gérer les utilisateurs et les livres

## 6. Modèle de données

Table : utilisateurs

Champ	Description
id	Identifiant unique
nom	Nom de l'utilisateur
email	Adresse e-mail
mot_de_passe	Mot de passe hashé
role	"user" ou "admin"
date_creation	Date d'inscription

Table : livres

Champ	Description
id	Identifiant unique
titre	Titre du livre
auteur	Auteur
genre	Genre littéraire
annee	Année de publication
isbn	Numéro ISBN (facultatif)
couverture_url	Image de couverture (facultative)
utilisateur_id	Lien vers le propriétaire du livre

## 7. Aspects techniques

- Langages : PHP, HTML, CSS
- Base de données : MySQL/MariaDB
- Architecture : code organisé en PHP orienté objet (modèle MVC simplifié)
- Sécurité :
  - Hash des mots de passe (password\_hash())
  - Requêtes SQL sécurisées (PDO + requêtes préparées)
  - Validation et nettoyage des données utilisateur
- Déploiement : hébergement sur un serveur public
- Configuration : infos de connexion à la base dans un fichier séparé (config.php)

## 8. Répartition du travail

Phase	Liliana	Thierry	Contributions communes
1. Conception / planification	<ul style="list-style-type: none"><li>- Brainstorming sur les fonctionnalités</li><li>- Élaboration du cahier des charges</li></ul>	<ul style="list-style-type: none"><li>- Organisation du projet dans GitHub</li><li>- Planification des tables et relations de la base</li></ul>	<ul style="list-style-type: none"><li>- Validation des choix ensemble</li><li>- Décision des rôles et responsabilités</li></ul>
2. Base de données	<ul style="list-style-type: none"><li>- Participation à la définition des champs et relations</li></ul>	<ul style="list-style-type: none"><li>- Création de la base MySQL et des tables</li><li>- Préparation de scripts d'insertion de données de test</li></ul>	<ul style="list-style-type: none"><li>- Vérification de l'intégrité des données</li><li>- Tests de cohérence</li></ul>
3. Pages HTML / Formulaires	<ul style="list-style-type: none"><li>- Création des pages publiques simples : accueil, inscription, connexion</li><li>- Formulaires d'ajout de livre</li></ul>	<ul style="list-style-type: none"><li>- Création des pages privées simples : tableau de bord, liste des livres, détails livre, administration</li></ul>	<ul style="list-style-type: none"><li>- Validation que toutes les pages sont fonctionnelles</li><li>- Vérification des liens entre pages</li></ul>
4. PHP / Back-end	<ul style="list-style-type: none"><li>- Participation à la création des classes PHP (Utilisateur, Livre, Lecture)</li><li>- Gestion de la logique métier pour ajouter/modifier/supprimer des livres</li></ul>	<ul style="list-style-type: none"><li>- Gestion de l'authentification et des sessions</li><li>- Gestion des rôles et permissions</li><li>- Fonctionnalités administrateur</li></ul>	<ul style="list-style-type: none"><li>- Test des interactions entre front-end et back-end</li><li>- Vérification que toutes les fonctionnalités demandées fonctionnent</li></ul>
5. Sécurité	<ul style="list-style-type: none"><li>- Validation côté serveur des formulaires</li><li>- Vérification des entrées utilisateurs</li></ul>	<ul style="list-style-type: none"><li>- Implémentation de PDO et requêtes préparées</li><li>- Hash des mots de passe</li><li>- Gestion sécurisée des sessions</li></ul>	<ul style="list-style-type: none"><li>- Tests globaux de sécurité</li><li>- Contrôle des droits d'accès</li></ul>
6. Tests et débogage	<ul style="list-style-type: none"><li>- Tests des pages publiques et privées</li></ul>	<ul style="list-style-type: none"><li>- Tests des fonctionnalités critiques (authentification, rôle)</li></ul>	<ul style="list-style-type: none"><li>- Correction des bugs ensemble</li><li>- Vérification que tout est fonctionnel selon le cahier des charges</li></ul>
7. Déploiement	<ul style="list-style-type: none"><li>- Préparation des fichiers pour le serveur</li></ul>	<ul style="list-style-type: none"><li>- Upload sur serveur public et configuration de la base</li></ul>	<ul style="list-style-type: none"><li>- Vérification finale que le site fonctionne en ligne</li></ul>
8. Documentation / Rapport final	<ul style="list-style-type: none"><li>- Rédaction du cahier des charges final</li></ul>	<ul style="list-style-type: none"><li>- Rédaction de la partie technique (structure base, code, architecture)</li></ul>	<ul style="list-style-type: none"><li>- Relecture et validation finale</li><li>- Préparation du mail de soumission</li></ul>

## 9. Conclusion

**libsproject.ch** est une application web qui aide à gérer sa bibliothèque personnelle.

Elle propose une interface simple : authentification, gestion des rôles, sécurité, sessions et base de données. Le but est de créer un site fonctionnel et facile à utiliser, tout en mettant en pratique ce qu'on a appris en cours. Ce projet permet de voir concrètement comment fonctionne une application web complète, depuis le développement jusqu'au déploiement.

### Notes fin de projet

Certaines fonctionnalités n'ont pas pu être mises en place

- Suivi de lecture
- API pour le référencement des livres

### Retour d'expérience

Nous avons finalement décidé de créer un compte "admin" plutôt qu'un compte "auteur" afin de pouvoir éventuellement exploiter les données récupérées (notes, genre, nombre de livres, ...). Dans un cadre réel un rôle auteur aurait nécessité un contrôle approfondi de l'identité de l'auteur.

Globalement le projet s'est bien déroulé, nous avons pu nous répartir les tâches selon les compétences et préférences de chacun.e. L'IA (Chat GPT et Claude) ont été utilisés parfois à des fins de contrôle, débbuging et compréhension.

### Difficultés rencontrées

La plus grande difficulté a été la mise en place de l'envoi d'e-mail. Nous avons été bloqués pendant un moment à cause du mot de passe généré par Infomaniak (le symbole \$ posait problème dans les mdp générés).

Nous avons pu également constater des difficultés à collaborer sur GitHub (conflits lors des push, ...). On s'est également un peu mélangé les pinceaux entre les essais en local et le distant.

Au final, tout va bien !