

Projet - Application BookShop

Nous décrivons ici un cahier des charges de l'application BookShop.

Travail à réaliser

L'objectif est de compléter les pages existantes, et développer les pages manquantes pour l'application Bookshop.

La page `index.php`

La page `index.php` représente la page d'accueil de l'application BookShop. Accessible aux utilisateurs authentifiés ou non, elle affiche en devanture les 4 livres les plus récents (les derniers ajouts au catalogue) et les 4 livres les plus vendus. Ces livres peuvent être ajoutés au panier. Si l'utilisateur est authentifié, il lui est également possible d'ajouter cet article à sa liste de cadeaux (s'il n'existe pas déjà).

Cette page contient un lien vers la page de présentation ([page presentation.html](#)).

La couverture de chaque livre est un lien vers la page affichant les détails du livre ([page details.php](#)).

Les auteurs affichés pour chaque livre renvoient vers la page de recherche par auteur, sur le nom de l'auteur ([page recherche.php](#)).

Il s'agit de la page réalisée lors du TP2 et du TP3, avec deux différences :

- les données manipulées sont cette fois extraites de la base de données,
- cette page offre la possibilité d'ajouter les articles au panier du visiteur, et à la liste de cadeaux, pour les utilisateurs authentifiés.

La page `presentation.html`

La page `presentation.html` présente un descriptif de l'application BookShop.

Il s'agit de la page réalisée lors du TP1.

La page `recherche.html`

La page de recherche permet de chercher un article dans les collections offertes par BookShop.

La recherche peut s'effectuer soit par le nom de l'auteur, soit par un mot contenu dans le titre de l'article.

Chaque résultat affiché peut être ajouté au panier du visiteur. Pour les utilisateurs authentifiés, il est aussi possible d'ajouter ces articles à la liste de cadeaux de l'utilisateur. Les auteurs affichés pour chaque résultat sont des liens qui renvoient au résultat d'une recherche par auteur sur le nom de l'auteur correspondant.

La couverture de chaque livre est un lien vers la page affichant les détails du livre ([page details.php](#)).

Il s'agit de la page réalisée lors du TP4, intégrant l'implémentation des boutons d'ajout au panier et à la liste de cadeaux.

La page `connexion.php`

La page de connexion permet à un utilisateur de fournir ses identifiants (adresse email + mot de passe) afin de s'identifier (bloc de gauche). Elle fournit également un moyen d'accès à la page d'inscription (bloc de droite - [page inscription.php](#)).

Si l'utilisateur remplit les bonnes informations, il est automatiquement redirigé vers la page ayant mené à la page de connexion. Par



exemple : si l'utilisateur était sur la page d'accueil, qu'il demande à s'authentifier, et qu'il réussit cette étape, il revient (cette fois-ci, authentifié) sur la page d'accueil.

En cas d'erreur, un message informera l'utilisateur.

Il s'agit de la page réalisée au TP5.

La page **deconnexion.php**

Cette page déconnecte l'utilisateur actuellement authentifié.

La page **inscription.php**

La page d'inscription présente un formulaire demandant à l'utilisateur

- d'indiquer son email,
- de choisir un mot de passe,
- de confirmer ce mot de passe,
- de donner son nom et son prénom,
- de saisir sa date de naissance.

Si l'utilisateur saisit des données correctes, un nouveau client est ajouté à la base de données, et sa session est ouverte. Comme pour la connexion, il est automatiquement redirigé vers la page ayant commandé l'authentification de l'utilisateur.

Si l'utilisateur saisit des données erronées, les erreurs sont indiquées et le formulaire réaffiche les informations saisies (à l'exception des mots de passe).

Il s'agit de la page réalisée au TP5.

La page **liste.php**

Cette page permet de consulter la liste d'un cadeau d'un utilisateur. L'utilisateur fournit l'adresse email de son ami et la page affiche alors les articles qui n'ont pas encore été offerts dans la liste de cet ami.

La couverture de chaque livre est un lien vers la page affichant les détails du livre ([page details.php](#)).

Dans cette page, la seule possibilité offerte pour les articles affichés, est d'ajouter cet article dans son panier, en vue de l'offrir.

La page **details.php**

La page de détails permet d'avoir le détail de chaque livre, conformément au modèle montré ci-contre.

Cette page permet au visiteur d'ajouter des articles dans son panier. Si le visiteur est authentifié, il peut aussi les ajouter à sa liste de cadeaux.

La page **panier.php**

Cette page permet d'afficher le panier de l'utilisateur. On affichera pour chaque article, sur chaque ligne :

- son titre, qui est un lien renvoyant vers le détail d'un article
- la quantité, qui peut-être modifiée pour être augmentée ou diminuée
- le prix unitaire du livre
- le total de la ligne (prix unitaire * quantité)

Le total est affiché en bas du tableau ; il s'agit de la somme des totaux de chaque ligne.

Il est possible de supprimer des éléments du panier, en cliquant sur la croix rouge en fin de chaque ligne.



Le bas de la page diffère si l'utilisateur est authentifié ou pas. S'il n'est pas connecté, un bouton renvoie vers la [page de connexion](#). S'il est déjà authentifié, on lui rappelle les coordonnées de livraison, et un bouton permet d'enregistrer la commande et ainsi de vider le panier.

La page **compte.php**

Cette page permet de visualiser et modifier les informations liées au compte de l'utilisateur authentifié. Elle n'est pas accessible sans une athentification préalable.

La première partie de la page permet de modifier les informations de connexion :

- le nom et le prénom
- la date de naissance
- l'adresse email de l'utilisateur
- un mot de passe et sa confirmation, qui permettent, s'ils sont renseignés de mettre à jour le mot de passe.

Les conditions de validité de ces champs sont identiques à celles vues lors du TP5.

La seconde partie de la page permet de modifier les informations de livraison :

- l'adresse (numéro et rue), qui peut être vide,
- code postal, obligatoire et composé de 4 ou 5 chiffres,
- la ville, obligatoire,
- le pays, obligatoire.

Une commande ne peut pas être enregistrée si ces informations ne sont pas renseignées.

La page **commandes.php**

Cette page permet de récapituler les commandes qui ont été passées par un utilisateur. Elle n'est pas accessible sans une athentification préalable.

Pour chaque commande, on affiche la date et l'heure à laquelle celle-ci a été passée (i.e. enregistrée), ainsi que le détails, sous la même forme que dans la [page du panier](#).

On affichera ces commandes en commençant par la plus récente.

La page **maliste.php**

Cette page permet à un utilisateur d'afficher la liste des cadeaux qu'il a choisi au fur et à mesure de ses visites. Cette page n'est pas accessible sans une athentification préalable

Pour chaque article, on affiche si celui-ci est déjà offert, et, le cas échéant par qui. Le lien sur le généreux donateur ouvre un mail de remerciement, prêt à être envoyé.

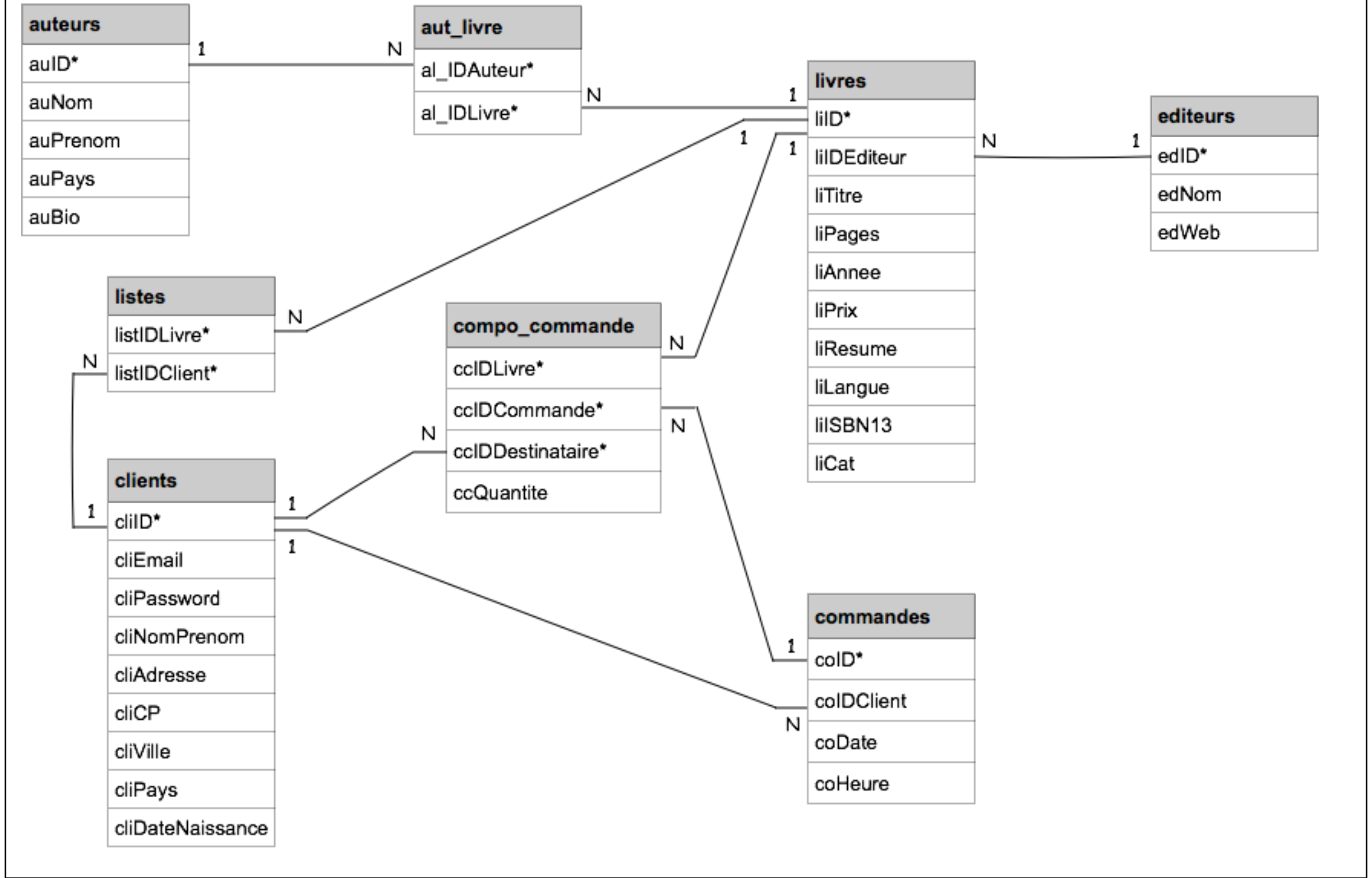
Il est possible de retirer de la liste des articles qui n'ont pas encore été offerts.

On remarquera que, de par la structure de la base de données, il n'est pas possible d'avoir plusieurs fois le même article dans sa liste.



Base de données **bookshop**

La base de données **bookshop** est composée de 8 tables, et permet de gérer les livres, et leurs auteurs, les éditeurs, les clients, les commandes, et les listes de cadeaux.



Nous donnons ici le détail de chacun des champs de la base (typage et description).

La table livres collecte tous les livres présentés dans la boutique en ligne BookShop.		
Champ	Type	Description
liID	int(11)	Clé primaire, numéro automatique, identifiant du livre
liIDEditeur	int(11)	Clé étrangère, référence de l'éditeur (editeurs.edID)
liTitre	char(255)	Titre du livre
liPages	int(4)	Nombre de pages du livre
liAnnee	int(4)	Année de publication du livre
liPrix	decimal(5,2)	Prix du livre.
liResume	text	Résumé du livre.
liLangue	char(2)	Langue du livre ('FR', 'EN')
liISBN13	int(20)	Numéro ISBN du livre
liCat	char(5)	Catégorie du livre ('ROMAN', 'TECH', 'BD')
La table auteurs enregistre les auteurs des livres.		
Champ	Type	Description
auID	int(11)	Clé primaire, numéro automatique, identifiant de l'auteur
auNom	char(30)	Nom de l'auteur
auPrenom	char(30)	Prénom de l'auteur
auPays	char(2)	Pays de l'auteur sur 2 lettres ('FR', 'US', 'GB', etc.)
auBio	texte	Biographie de l'auteur
La table aut_livre permet d'associer les auteurs aux livres qu'ils ont écrits.		
Champ	Type	Description

al_IDLivre	int(11)	Clé primaire avec al_IDAuteur , référence le livre (livres.liID)
al_IDAuteur	int(11)	Clé primaire avec al_IDLivre , référence l'auteur (auteurs.auID)
La table editeurs enregistre les éditeurs de livres.		
Champ	Type	Description
edID	int(11)	Clé primaire, numéro automatique, identifiant de l'éditeur
edNom	char(30)	Nom de l'éditeur
edWeb	char(100)	Site web de l'éditeur
La table clients enregistre les clients/utilisateurs enregistrés auprès de la boutique.		
Champ	Type	Description
cliID	int(11)	Clé primaire, numéro automatique, identifiant du client
cliEmail	char(50)	Clé candidate (doit être unique), adresse email du client
cliPassword	char(50)	Mot de passe du client. On enregistre ici un hash md5 du mot de passe, et pas la valeur en clair.
cliNomPrenom	char(100)	Nom et prénom du client.
cliAdresse	char(100)	Adresse du client (numéro et rue)
cliCP	int(5)	Code postal du client
cliVille	char(50)	Ville du client
cliPays	char(50)	Pays du client
cliDateNaissance	int(8)	Date de naissance du client (format aaaammjj)
La table commandes enregistre les commandes passées par les clients.		
Champ	Type	Description
coID	int(11)	Clé primaire, numéro automatique, identifiant de la commande
coIDClient	int(11)	Clé étrangère, référence au client ayant passé la commande (clients.cliID)
coDate	int(8)	Date à la quelle la commande a été passée, exprimée par 8 chiffres, au format AAAAMMJJ.
coHeure	int(4)	Heure à la quelle la commande a été passée, exprimée par 4 chiffres, au format HHMM.
La table compo_commande enregistre les articles composants la commande, avec leur quantité associée et la personne à qui se destine chaque article.		
Champ	Type	Description
ccIDCommande	int(11)	Clé primaire (avec ccIDLivre et ccDestinataire), référence de la commande (commandes.coID)
ccIDLivre	int(11)	Clé primaire (avec ccIDCommande et ccDestinataire), référence au livre commandé (livres.liID)
ccDestinataire	int(11)	Clé primaire (avec ccIDCommande et ccIDLivre), référence le client à qui se destine l'achat.
ccQuantite	int(3)	Quantité de livre commandé.
La table liste enregistre les listes de cadeaux des différents utilisateurs. Restriction : chaque utilisateur ne peut demander un cadeau qu'une seule fois.		
Champ	Type	Description
lisIDClient	int(11)	Clé primaire (avec lisIDLivre), référence au client souhaitant se faire offrir le livre associé.
lisIDLivre	int(11)	Clé primaire (avec lisIDClient), référence au livre demandé par le client associé.

La base de données **bookshop** est installée sur le même serveur que celui où seront stockés vos scripts PHP (accès depuis les scripts : **localhost**).

Vous disposerez chacun d'un accès personnalisé, avec les paramètres suivants :

- Nom de la base : **nomEtudiant_bookshop** (ex. **dadeau_bookshop**)
- Nom d'utilisateur : **nomEtudiant_user** (ex. **dadeau_user**)
- Mot de passe utilisateur : **nomEtudiant_pass** (ex. **dadeau_pass**)

Si vous souhaitez travailler en local (sur votre machine perso), vous pouvez utiliser ce [script de création de la base](#) qu'il vous suffira d'importer dans PHPMyAdmin. Ce script SQL a pour effet de créer la base **bookshop_db** avec l'utilisateur **bookshop_user** ayant pour mot de passe **bookshop_pass**.

Recommandations

Il vous est demandé de bien suivre les [recommandations](#) pour l'écriture de code PHP/MySQL qui vous ont déjà été présentées.

Répertoires

Vous **DEVEZ** utiliser l'arborescence de répertoires suivante :

```
bookshop
├── html
├── images
├── styles
└── php
```

Pour travailler chez vous, vous trouverez [dans cette archive](#) les images utiles pour ce projet.