

Dossier documentaire

1. Existant.....	1
Présentation.....	1
Les différents onglets.....	2
La base de données	5
L'API REST.....	6
2. Missions à réaliser.....	7
Remarques générales	7
Mission 1 : gérer les documents (8h).....	7
Mission 2 : gérer les commandes (12h).....	7
Mission 3 : gérer le suivi de l'état des exemplaires (5h)	8
Mission 4 : mettre en place des authentifications (4h).....	8
Mission 5 : assurer la sécurité, la qualité et intégrer des logs (8h).....	8
Mission 6 : tester et documenter (8h).....	9
Mission 7 : déployer et gérer les sauvegardes des données (4h)	9

1. Existant

Un premier développeur s'est occupé de la construction de la base de données et du développement de certaines fonctionnalités de l'application. C'est une application de bureau, prévue d'être installée sur plusieurs postes dans la médiathèque et accédant à la même base de données.

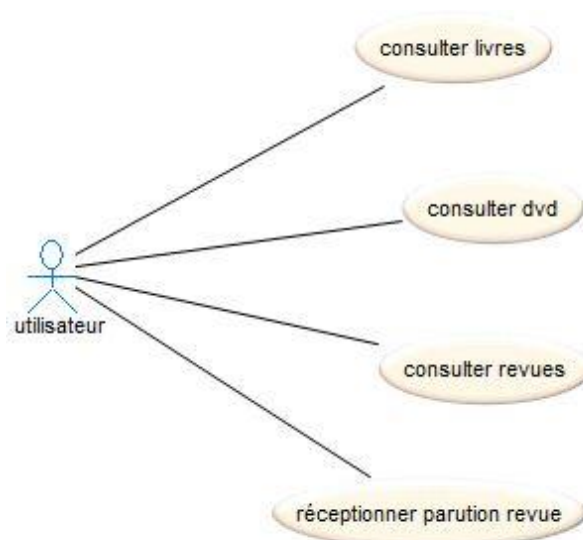
Le code actuel de l'application peut être récupéré ici :

<https://github.com/CNED-SLAM/MediaTekDocuments>

L'application exploite une API REST pour accéder à la BDD MySQL. Des explications sont données plus loin, ainsi que le lien de récupération.

Présentation

Le premier développeur qui s'est occupé de l'application a codé les fonctionnalités suivantes : recherches et affichage d'informations sur les documents de la médiathèque (livres, DVD, revues), réception de nouveaux numéros de revues.



L'application ne comporte qu'une seule fenêtre divisée en plusieurs onglets.

Les différents onglets

Onglet 1 : Livres

Cet onglet présente la liste des livres, triée par défaut sur le titre.

La liste comporte les informations suivantes : titre, auteur, collection, genre, public, rayon.

Gestion Médiathèque

Livres

DVD

Revue

Parutions des revues

Recherches

Saisir le titre ou la partie d'un titre :

Ou sélectionner le genre : X

Saisir un numéro de document : Rechercher

Ou sélectionner le public : Tous publics X

Ou sélectionner le rayon : X

Id	Titre	Auteur	Collection	Genre	Public	Rayon
00019	Guide Vert - Iles Canaries		Guide Vert	Voyages	Tous publics	Voyages
00020	Guide Vert - Irlande		Guide Vert	Voyages	Tous publics	Voyages
00011	L'île des oubliés	Victoria Hislop		Roman	Tous publics	Littérature française
00006	La marque jaune	Edgar P. Jacobs	Blake et Mortimer	Bande dessinée	Tous publics	BD Adultes
00026	La planète des singes	Pierre Boulle	Julliard	Science Fiction	Tous publics	Littérature française
00012	La souris bleue	Kate Atkinson		Roman	Tous publics	Littérature française
00018	Le Routard - Maroc		Guide du Routard	Voyages	Tous publics	Voyages
00014	Mauvaise étoile	RJ Ellory		Policier	Tous publics	Policiers français étrangers

Informations détaillées

Numéro de document : 00026

Code ISBN : 2266283022

Titre :

La planète des singes

Auteur(e) :

Pierre Boulle

Collection :

Julliard

Genre :

Science Fiction

Public :


Tous publics

Rayon :

Littérature française

Chemin de l'image :

C:\Images\Livres\La_planète_des_singes.jpg



Recherches

Par le titre : Il est possible de rechercher un ou plusieurs livres par le titre. La saisie dans la zone de recherche se fait en autocomplétions sans tenir compte de la casse. Seuls les livres concernés apparaissent dans la liste.

Par le numéro : il est possible de saisir un numéro et, en cliquant sur "Rechercher", seul le livre concerné apparaît dans la liste (ou un message d'erreur si le livre n'est pas trouvé, avec la liste remplie à nouveau).

Filtres

Il est possible d'appliquer un filtre (un seul à la fois) sur une de ces 3 catégories : genre, public, rayon.

Un combo par catégorie permet de sélectionner un item. Seuls les livres correspondant à l'item sélectionné, apparaissent dans la liste (par exemple, en choisissant le genre "Policier", seuls les livres de genre "Policier" apparaissent).

Le fait de sélectionner un autre filtre ou de faire une recherche, annule le filtre actuel.

Il est possible aussi d'annuler le filtre en cliquant sur une des croix.

Tris

Le fait de cliquer sur le titre d'une des colonnes de la liste des livres, permet de trier la liste par rapport à la colonne choisie.

Affichage des informations détaillées

Si la liste des livres contient des éléments, par défaut il y en a toujours un de sélectionné. Il est aussi possible de sélectionner une ligne (donc un livre) en cliquant n'importe où sur la ligne.

La partie basse de la fenêtre affiche les informations détaillées du livre sélectionné (numéro de document, code ISBN, titre, auteur(e), collection, genre, public, rayon, chemin de l'image) ainsi que l'image.

Onglet 2 : DVD

Cet onglet présente la liste des DVD, triée par titre.

La liste comporte les informations suivantes : titre, durée, réalisateur, genre, public, rayon.

Le fonctionnement est identique à l'onglet des livres.

La seule différence réside dans certaines informations détaillées, spécifiques aux DVD : durée (à la place de ISBN), réalisateur (à la place de l'auteur), synopsis (à la place de collection).

Onglet 3 : Revues

Cet onglet présente la liste des revues, triées par titre.

La liste comporte les informations suivantes : titre, périodicité, délai mise à dispo, genre, public, rayon.

Le fonctionnement est identique à l'onglet des livres.

La seule différence réside dans certaines informations détaillées, spécifiques aux revues : périodicité (à la place de l'auteur), délai mise à dispo (à la place de collection).

Onglet 4 : Parutions des revues

Cet onglet permet d'enregistrer la réception de nouvelles parutions d'une revue.

Il se décompose en 2 parties (groupbox).

Partie "Recherche revue"

Cette partie permet, à partir de la saisie d'un numéro de revue (puis en cliquant sur le bouton "Rechercher"), d'afficher toutes les informations de la revue (comme dans l'onglet précédent), ainsi que son image principale en petit, avec en plus la liste des parutions déjà reçues (numéro, date achat, chemin photo). Sur la sélection d'une ligne dans la liste des parutions, la photo de la parution correspondante s'affiche à droite.

Dès qu'un numéro de revue est reconnu et ses informations affichées, la seconde partie ("Nouvelle parution réceptionnée pour cette revue") devient accessible.

Si une modification est apportée au numéro de la revue, toutes les zones sont réinitialisées et la seconde partie est rendue inaccessible, tant que le bouton "Rechercher" n'est pas utilisé.

Partie "Nouvelle parution réceptionnée pour cette revue"

Cette partie n'est accessible que si une revue a bien été trouvée dans la première partie.

Il est possible alors de réceptionner une nouvelle parution en saisissant son numéro, en sélectionnant une date (date du jour proposée par défaut) et en cherchant l'image correspondante (optionnel) qui doit alors s'afficher à droite.

Le clic sur "Valider la réception" va permettre d'ajouter un tuple dans la table Exemple de la BDD. La parution correspondante apparaîtra alors automatiquement dans la liste des parutions et les zones de la partie "Nouvelle parution réceptionnée pour cette revue" seront réinitialisées.

Si le numéro de la parution existe déjà, il n'est pas ajouté et un message est affiché.

Gestion des documents de la médiathèque

Livres

DVD

Revue

Parutions des revues

Recherche revue

Numéro revue :

10011

Rechercher

Titre :

Geo

Périodicité :

MS

Délai mise à dispo :

52

Genre :

Presse Culturelle

Public :

Tous publics

Rayon :

Magazines

Chemin de l'image :

Parutions :

Numero	DateAchat	Photo
505	16/10/2022	
514	01/12/2021	
513	01/11/2021	
512	06/10/2021	
511	01/09/2021	

Image revue :

Image exemplaire :

Nouvelle parution réceptionnée pour cette revue

Numéro réceptionné :

Date de parution :

19/10/2022

Emplacement image :

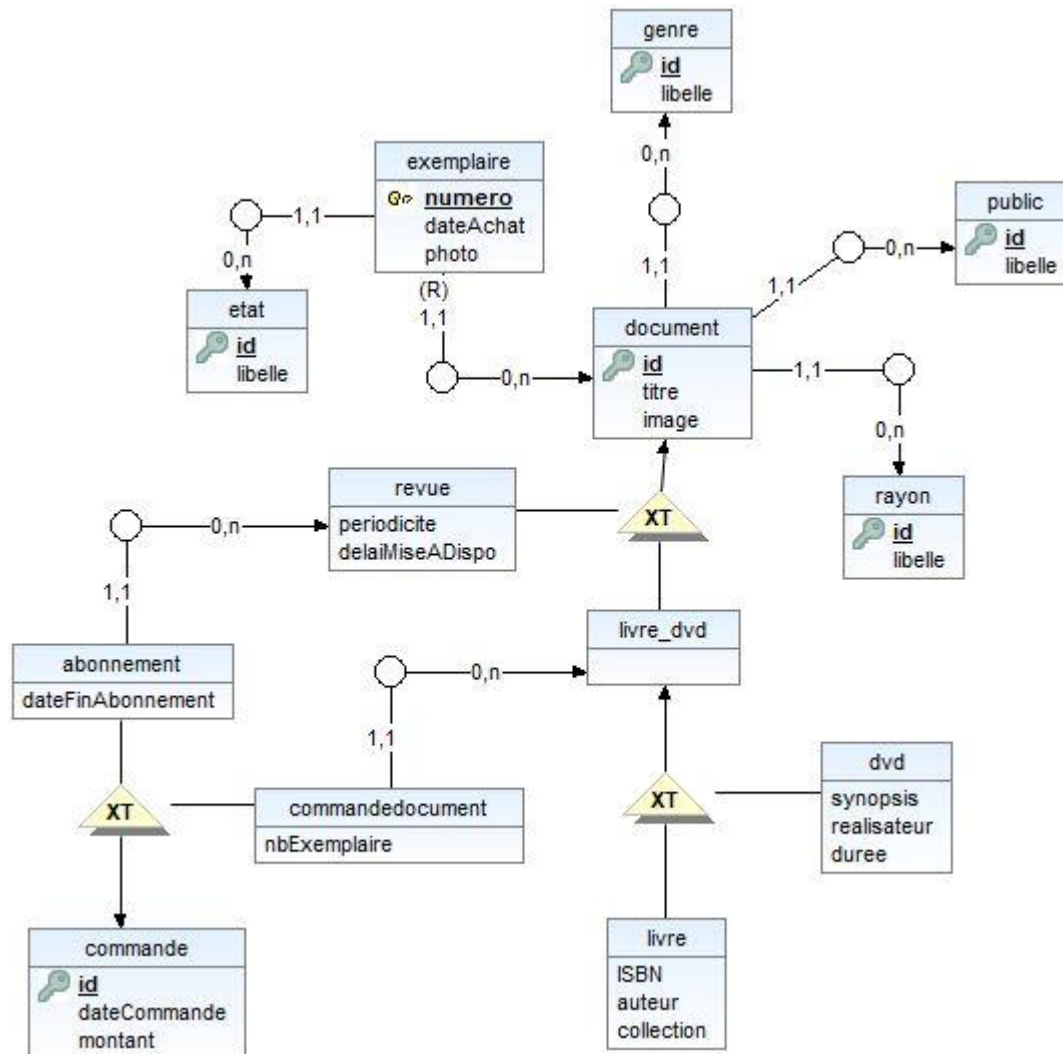
Rechercher

Valider la réception

Image exemplaire :

La base de données

La base de données 'mediatek86' est au format MySQL.
Voici sa structure :



On distingue les documents "génériques" (ce sont les entités Document, Revue, Livres-DVD, Livre et DVD) des documents "physiques" qui sont les exemplaires de livres ou de DVD, ou bien les numéros d'une revue ou d'un journal.

Chaque exemplaire est numéroté à l'intérieur du document correspondant, et a donc un identifiant relatif. Cet identifiant est réel : ce n'est pas un numéro automatique.

Un exemplaire est caractérisé par :

- un état d'usure, les différents états étant mémorisés dans la table Etat ;
- sa date d'achat ou de parution dans le cas d'une revue ;
- un lien vers le fichier contenant sa photo de couverture de l'exemplaire, renseigné uniquement pour les exemplaires des revues, donc les parutions (chemin complet) ;

Un document a un titre (titre de livre, titre de DVD ou titre de la revue), concerne une catégorie de public, possède un genre et est entreposé dans un rayon défini. Les genres, les catégories de public et les rayons sont gérés dans la base de données. Un document possède aussi une image dont le chemin complet est mémorisé. Même les revues peuvent avoir une image générique, en plus des photos liées à chaque exemplaire (parution).

Une revue est un document, d'où le lien de spécialisation entre les 2 entités. Une revue est donc identifiée par son numéro de document. Elle a une périodicité (quotidien, hebdomadaire, etc..) et un délai de mise à disposition (temps pendant lequel chaque exemplaire est laissé en consultation). Chaque parution (exemplaire) d'une revue n'est disponible qu'en un seul "exemplaire".

Un livre a aussi pour identifiant son numéro de document, possède un code ISBN, un auteur et peut faire partie d'une collection. Les auteurs et les collections ne sont pas gérés dans des tables séparées (ce sont de simples champs textes dans la table Livre).

De même, un DVD est aussi identifié par son numéro de document, et possède un synopsis, un réalisateur et une durée. Les réalisateurs ne sont pas gérés dans une table séparée (c'est un simple champ texte dans la table DVD). Enfin, 3 tables permettent de mémoriser les données concernant les commandes de livres ou DVD et les abonnements. Une commande est effectuée à une date pour un certain montant. Un abonnement est une commande qui a pour propriété complémentaire la date de fin de l'abonnement : il concerne une revue. Une commande de livre ou DVD a comme caractéristique le nombre d'exemplaires commandé et concerne donc un livre ou un DVD.

Le développeur qui a construit la base de données, l'a remplie de quelques exemples pour pouvoir tester son application. Dans les champs image (de Document) et photo (de Exemplaire) doit normalement se trouver le chemin complet vers l'image correspondante. Pour les tests, vous devrez créer un dossier, le remplir de quelques images et mettre directement les chemins dans certains tuples de la base de données qui, pour le moment, ne contient aucune image.

Lorsque l'application sera opérationnelle, c'est le personnel de la médiathèque qui sera en charge de saisir les informations des documents.

L'API REST

L'accès à la BDD se fait à travers une API REST protégée par une authentification basique.

Cette API contient les fichiers suivants :

- **.htaccess** : contient les règles de réécriture pour l'accès à mediatekdocuments.php avec les bons paramètres.
- **Mediatekdocuments.php** : point d'entrée de l'api, contrôle l'accès sécurisé, récupère les paramètres puis, suivant le verbe utilisé (GET, POST, PUT, DELETE), appelle la méthode concernée dans la classe Controle.
- **Controle.php** : suivant les demandes, appelle les méthodes concernées dans la classe AccesBDD puis retourne le résultat au format json.
- **AccesBDD.php** : construit les requêtes SQL avec les collections de paramètres, demande à la classe ConnexionPDO de les exécuter et retourne le résultat.
- **ConnexionPDO.php** : se connecte à la BDD, construit les requêtes en intégrant les paramètres, les exécute et retourne le résultat.

Le code de l'API se trouve ici :

https://github.com/CNED-SLAM/rest_mEDIATEKDOCUMENTS

avec toutes les explications pour l'utiliser (dans le README.md).

2. Missions à réaliser

Le projet revient à votre équipe de développeurs, gérée par un responsable qui a déterminé les différentes missions à effectuer.

Remarques générales

Votre responsable retient votre attention sur plusieurs aspects :

- Le code ajouté doit respecter la structure en couches et la logique de code de l'application actuelle.
- Les manipulations des utilisateurs doivent être sécurisées, comme c'est le cas dans l'application actuelle (par exemple, en mettant certains objets graphiques en lecture seule).
- Les requêtes SQL doivent éviter les actes malveillants.
- Les données correspondant à des valeurs prédéfinies ne doivent pas être saisies mais sélectionnées dans une liste.

Mission 1 : gérer les documents (8h)

L'application doit permettre d'ajouter, modifier ou supprimer des documents, sous certaines conditions. Les documents concernés sont les livres, les DVD et les revues.

L'ajout d'un document ne suppose pas forcément une commande, même s'il intervient classiquement au moment d'une commande. Dans cette mission, on ne s'occupe pas encore des commandes.

La modification d'un document ne peut pas porter sur son numéro.

La suppression d'un document n'est possible que s'il ne possède aucun exemplaire ni commande. Cela permet d'éviter les accidents de suppression.

Mission 2 : gérer les commandes (12h)

Tâche 1 : gérer les commandes de livres ou de DVD (8h)

Il doit être possible de commander un ou plusieurs exemplaires d'un livre ou d'un DVD et de suivre l'évolution d'une commande.

Une commande passe par différents stades :

- au moment de son enregistrement, elle est "en cours" ;
- au moment de sa réception, elle est "livrée" ;
- une fois livrée, le paiement est effectué, elle est alors "réglée" ;
- dans le cas où la livraison tarde, la commande est "relancée" (ce qui provoque un envoi de mail au fournisseur : cet aspect n'est pas à gérer) ;

La gestion des stades n'a pas encore été modélisée dans la base de données actuelle : vous devez donc vous en charger.

L'application doit permettre de voir, pour les livres ou les DVD, la liste des commandes et gérer le suivi.

Lorsqu'une commande de livre ou de DVD est "livrée", il faut que les exemplaires concernés soient automatiquement générés dans la BDD, avec un numéro séquentiel par rapport au document concerné.

Une commande doit pouvoir être supprimée si elle n'est pas encore livrée.

Tâche 2 : gérer les commandes de revues (4h)

Il doit être possible de commander une revue. Une commande de revue revient à réaliser un abonnement (ou un renouvellement d'abonnement, ce qui revient au même).

L'application doit permettre de voir la liste des commandes, même pour les abonnements expirés. Une commande ne peut être supprimée que si aucun exemplaire (parution) lié à cette commande n'est enregistré.

Au démarrage de l'application, une petite fenêtre doit s'ouvrir automatiquement pour afficher les revues dont l'abonnement se termine dans moins de 30 jours.

Mission 3 : gérer le suivi de l'état des exemplaires (5h)

L'application doit permettre de gérer l'état des documents physiques, donc des exemplaires.

Aussi bien pour les exemplaires de livres, de DVD et de revues (donc les parutions), leur création les met dans l'état "neuf". L'application doit permettre de changer l'état ("neuf", "usagé", "détérioré", "inutilisable").

Il doit être possible de supprimer un exemplaire.

Mission 4 : mettre en place des authentifications (4h)

Il existe plusieurs services dont les employés ne doivent pas avoir les mêmes droits d'accès aux fonctionnalités de l'application.

Les employés du service administratif gèrent les commandes. Ils ont aussi accès à la consultation et la mise à jour du catalogue (livres, DVD, revues). Donc ils ont accès à toutes les fonctionnalités de l'application.

Les employés du service Prêts s'occupent des prêts de documents, des retours, de l'étiquetage et du rangement dans les rayons. Ils ont accès en consultation seulement au catalogue (livres, DVD, revues) disponible.

Les employés du service Culture organisent divers événements (conférences, expositions, lecture aux enfants, etc..) et gèrent la venue d'intervenants pour ces événements. Ils ne doivent donc pas avoir accès aux fonctionnalités de cette application.

L'administrateur a accès à toutes les fonctionnalités de toutes les applications.

Dans le cadre d'une première version du système d'authentification, le but est de gérer les employés (avec login/pwd) et les services directement dans la base de données.

L'application doit donc démarrer sur une authentification qui permet de déterminer si l'employé est reconnu et son service d'affectation.

Suivant le service d'appartenance de l'employé, certaines fonctionnalités ne seront pas accessibles. Dans le cas des employés du service Culture, un message doit les informer que cette application n'est pas accessible pour eux.

Mission 5 : assurer la sécurité, la qualité et intégrer des logs (8h)

Tâche 1 : corriger des problèmes de sécurité (3h)

Pour chaque problème, créer une "issue" dans le dépôt GitHub correspondant, affecter l'issue à un des développeurs (si vous êtes 2, vous ou votre collègue, sinon vous-même), créer une branche pour proposer une correction de code en expliquant la correction, faire un pull request, accepter le pull request si le résultat correspond bien aux attentes.

Problème n°1 :

Actuellement, l'accès à l'API se fait en authentification basique, avec le couple "login:pwd" en dur dans le code de l'application (dans le constructeur de la classe Access). Le but est de sécuriser cette information.

Problème n°2 :

Si, pour accéder à l'API directement dans un navigateur, on donne juste l'adresse de l'API sans mettre de paramètres :

http://localhost/rest_mediatekdocuments/

on obtient la liste des fichiers contenus dans le dossier de l'API. Le but est d'avoir un retour d'erreur.

Tâche 2 : contrôler la qualité (3h)

Mettre en place, en local, un serveur SonarQube et le lien avec l'application C# avec une intégration continue entre C# et SonarQube, avec Jenkins.

Le code d'origine a été nettoyé en suivant les conseils de Sonarlint et des avertissements de Visual Studio. Le seul point qui n'a pas été corrigé est le nom des méthodes événementielles, qui commencent par une minuscule car les objets graphiques commencent par une minuscule.

Excepté ce point, les règles d'écriture de code sont les règles standards.

Le but est donc de nettoyer le code ajouté en suivant la même logique.

Contrôler la qualité avec SonaQube.

Tâche 3 : intégrer des logs (2h)

Dans la classe Access, tous les affichages console doivent être aussi enregistrés dans un fichier de logs.

Mission 6 : tester et documenter (8h)

Tâche 1 : gérer les tests (5h)

Tests unitaires :

Écrire les tests unitaires sur les classes du package Model.

Tests fonctionnels :

Le responsable désire voir le fonctionnement des tests avec Specflow : dans l'application C#, écrire les tests fonctionnels sur les recherches d'un onglet (par exemple les livres).

Construire une collection de tests dans Postman pour contrôler les fonctionnalités de l'API d'accès à la BDD.

Tâche 2 : créer les documentations techniques (1h)

Générer les documentations techniques correspondant à l'application C# et à l'API REST.

Tâche 3 : créer la documentation utilisateur en vidéo (2h)

Créer une vidéo de 10mn maximum qui présente l'ensemble des fonctionnalités de l'application C#.

Mission 7 : déployer et gérer les sauvegardes des données (4h)

Tâche 1 : déployer le projet (3h)

Mettre en ligne l'API et la base de données (chez un hébergeur).

Créer un installeur de l'application C# pour qu'elle puisse s'installer facilement.

Tâche 2 : gérer les sauvegardes des données (1h)

Automatiser une tâche de sauvegarde quotidienne de la BDD.