

Compte Rendu Bibliothèque

1)Contexte :

Dans le cadre d'une bibliothèque de plus en plus grandissante, il a été souhaité de pouvoir répertorier l'ensemble des livres et leur attributs dans une base de données, afin de simplifier la gestion et recherche de ces derniers.

Dans cette base de données, se nommant « bookshelf », les livres sont stockés dans la table « books », possédant ainsi six colonnes :

- Titre : le titre du livre.
- Auteur : l'auteur du livre.
- Catégorie : si le livre est une BD, un comics, un roman, etc...
- type : Si le livre est de l'horreur, du comics, etc.
- Édition : l'éditeur du livre.
- Pages : Son nombre de pages au total.

Cette base de données étant stockée sur un serveur en local, il est nécessaire de posséder une application simple d'utilisation afin de pouvoir manipuler et effectuer des recherches dans la base de données de livres, et pouvoir rajouter au besoin une future acquisition de la bibliothèque.

Le choix de la technologie utilisée pour la BD est du Postgresql.

2)Objectifs :

L'objectif principal de l'application est de permettre à son utilisateur de pouvoir rapidement trouver un livre par rapport à son titre, auteur ou édition, et prendre connaissance des détails de celui-ci.

Avec ceci, l'utilisateur doit pouvoir manipuler les attributs du livre pour pouvoir modifier au besoin et d'en rajouter si la bibliothèque viendrait à en obtenir de nouveau.

3)Architecture :

Le choix de l'architecture client serveur c'est porté sur le RMI.

Cette méthode permet le déploiement d'un ensemble d'application « clientes » peu gourmande en ressources, car le traitement de données et des actions sont effectués du côté du serveur, allégeant la charge côté client.

4)Détails des fonctionnalités :

Dans l'application, nous pouvons trouver deux catégories de fonctionnalités :

-affichage de la liste des attributs des livres :

Cette catégorie permet d'obtenir la liste des livres, la listes des auteurs, la liste des catégoire,la listes des types de livres et la liste des éditions.

-La manipulation d'attributs spécifiques :

Cette catégorie permet de rajouter un nouveau livre, obtenir un attribut d'un livre spécifique, ou de modifier cette attribut.

On retrouve donc dans la première catégorie :

-getListTitre :

affiche la liste des livres contenu dans la base de données via un « SELECT Titre FROM public.Books ».

-getListAuthor:

Affiche la liste des auteurs contenu dans la base de données via un « SELECT Auteur FROM public.Books ».

-getListCat:

Affiche la liste des catégories contenu dans la base de données via un « SELECT Catégorie FROM public.Books ».

-getListeType:

Affiche la liste des types contenu dans la base de données via un « SELECT Type FROM public.Books ».

-getListEdition:

Affiche la liste des éditions contenu dans la base de données via un « SELECT Edition FROM public.Books ».

On retrouve dans la seconde catégorie :

-SetBook :

Rajoute un nouveau livre dans la base de données via un « INSERT INTO public.Books VALUES (**le titre , l'auteur , la catégorie ,le type , l'édition , le nombre de pages**) ».

-GetPages :

On récupère le nombre de pages d'un livre que l'on obtient en paramètre de la méthode via « SELECT Titre,Pages FROM public.Books WHERE Titre = **le titre du livre** »

-SetPages :

On défini le nombre de pages d'un livre que on obtient en paramètres de la méthode via « UPDATE public.Books SET Pages = **nombre de pages** WHERE Titre = **le nom du livre** ».

-GetAuthor :

On récupère l'auteur d'un livre que l'on obtient en paramètre de la méthode via « SELECT Titre,Auteur FROM public.Books WHERE Titre = **le titre du livre** »

-SetAuthor :

On définit l'auteur d'un livre que l'on obtient en paramètres de la méthode via « UPDATE public.Books SET Auteur = **nouveau auteur** WHERE Titre = **le nom du livre** ».

-SetTitre :

On modifie le titre d'un livre que l'on obtient en paramètres de la méthode via « UPDATE public.Books SET Titre = **nouveau titre** WHERE Titre = **l'ancien nom du livre** ».

-GetCat :

On récupère la catégorie d'un livre que l'on obtient en paramètre de la méthode via « SELECT Titre,Catégorie FROM public.Books WHERE Titre = **le titre du livre** »

-SetCat :

On définit le nombre de pages d'un livre que l'on obtient en paramètres de la méthode via « UPDATE public.Books SET Catégorie = **nouvelle catégorie** WHERE Titre = **le nom du livre.** »

-GetType :

On récupère le type d'un livre que l'on obtient en paramètre de la méthode via « SELECT Titre,Type FROM public.Books WHERE Titre = **le titre du livre** »

-SetType :

On définit le nombre de pages d'un livre que l'on obtient en paramètres de la méthode via « UPDATE public.Books SET Type= **nouveau type** WHERE Titre = **le nom du livre.** »

-GetEdition :

On récupère le nombre de pages d'un livre que l'on obtient en paramètre de la méthode via « SELECT Titre,Edition FROM public.Books WHERE Titre = **le titre du livre** »

-SetEdition :

On définit le nombre de pages d'un livre que l'on obtient en paramètres de la méthode via « UPDATE public.Books SET Edition = **nouvelle édition** WHERE Titre = **le nom du livre.** »