

Plan de tests

Tests unitaires sur les classes du package model

But du test	Action de contrôle	Résultat attendu	Bilan
Contrôler la méthode ToString() de la classe Categorie pour voir si elle retourne le libelle.	Test unitaire lancé après avoir créé un objet de type Genre avec libelle contenant : "Roman"	"Roman"	OK
Contrôler la méthode ToString() de la classe Etat pour voir si elle retourne le libelle.	Test unitaire lancé après avoir créé un objet de type Etat avec libelle contenant : "usagé"	"usagé"	OK
Contrôler la méthode ToString() de la classe Suivi pour voir si elle retourne le stade.	Test unitaire lancé après avoir créé un objet de type Suivi avec stade contenant : "En cours"	"En cours"	OK
Contrôler la méthode ParutionDansAbonnement(DateTime dateParution) de la classe Abonnement pour voir si elle retourne bien vrai ou faux selon la date passée	Test unitaire lancé après avoir créé un objet de type Abonnement avec une date de commande de 2025-03-01 et une date de fin d'abonnement de 2025-04-01 et essai avec des dates comprises et non comprises dans l'abonnement	2025-02-25 -> False 2025-03-01 -> True 2025-03-15 -> True 2025-04-01 -> False 2025-04-15 -> False	OK
Contrôler que la désérialisation d'un objet de type Abonnement s'effectue avec succès	Test unitaire lancé en désérialisant une chaine JSON venant de l'API REST représentant un objet de type Abonnement et validation des propriétés valorisés	L'objet Abonnement n'est pas null et les propriétés sont toutes valorisés	OK
Contrôler que la classe mère des types Public, Rayon et Genre est bien Categorie	Création d'objets de types Public, Rayon et Genre et vérification que le type de l'objet est une sous classe de Categorie	Les objets de type Public, Rayon et Genre sont une sous classe de Categorie	OK
Contrôler que la désérialisation d'un objet de type CommandeLivreDvd s'effectue avec succès	Test unitaire lancé en désérialisant une chaine JSON venant de l'API REST représentant un objet de type CommandeLivreDvd et validation des propriétés valorisés	L'objet CommandeLivreDvd n'est pas null et les propriétés sont toutes valorisés	OK
Contrôler que la classe mère des types Abonnement et CommandeLivreDvd est bien Commande	Création d'objets de types Abonnement et CommandeLivreDvd et vérification que le type de l'objet est une sous classe de Commande	Les objets de type Abonnement et CommandeLivreDvd sont une sous classe de Commande	OK
Contrôler que la classe mère des Livre, Dvd et Revue est bien Document	Création d'objets de types Livre, Dvd et Revue et vérification que le type de l'objet est une sous classe de Document	Les objets de type Livre, Dvd et Revue sont une sous classe de Document	OK
Contrôler que la désérialisation d'un objet de type Dvd	Test unitaire lancé en désérialisant une chaine JSON	L'objet Dvd n'est pas null et les propriétés	OK

s'effectue avec succès	venant de l'API REST représentant un objet de type Dvd et validation des propriétés valorisés	sont toutes valorisés	
Contrôler que la désérialisation d'un objet de type Etat s'effectue avec succès	Test unitaire lancé en désérialisant une chaine JSON venant de l'API REST représentant un objet de type Etat et validation des propriétés valorisés	L'objet Etat n'est pas null et les propriétés sont toutes valorisés	OK
Contrôler que la désérialisation d'un objet de type Exempleaire s'effectue avec succès	Test unitaire lancé en désérialisant une chaine JSON venant de l'API REST représentant un objet de type Exempleaire et validation des propriétés valorisés	L'objet Exempleaire n'est pas null et les propriétés sont toutes valorisés	OK
Contrôler la méthode ToRestApiObject() de la classe Exempleaire pour voir si l'objet retourné est bien celui attendu.	Test unitaire lancé après avoir créé un objet de type Exempleaire avec les propriétés valorisés	L'objet retourné possède seulement les propriétés attendues par le point d'entrée de l'API	OK
Contrôler que la désérialisation d'un objet de type Genre s'effectue avec succès	Test unitaire lancé en désérialisant une chaine JSON venant de l'API REST représentant un objet de type Genre et validation des propriétés valorisés	L'objet Genre n'est pas null et les propriétés sont toutes valorisés	OK
Contrôler que la classe mère des Livre et Dvd est bien LivreDvd	Création d'objets de types Livre et Dvd et vérification que le type de l'objet est une sous classe de LivreDvd	Les objets de type Livre et Dvd sont une sous classe de LivreDvd	OK
Contrôler que la désérialisation d'un objet de type Livre s'effectue avec succès	Test unitaire lancé en désérialisant une chaine JSON venant de l'API REST représentant un objet de type Livre et validation des propriétés valorisés	L'objet Livre n'est pas null et les propriétés sont toutes valorisés	OK
Contrôler que la désérialisation d'un objet de type Public s'effectue avec succès	Test unitaire lancé en désérialisant une chaine JSON venant de l'API REST représentant un objet de type Public et validation des propriétés valorisés	L'objet Public n'est pas null et les propriétés sont toutes valorisés	OK
Contrôler que la désérialisation d'un objet de type Rayon s'effectue avec succès	Test unitaire lancé en désérialisant une chaine JSON venant de l'API REST représentant un objet de type Rayon et validation des propriétés valorisés	L'objet Rayon n'est pas null et les propriétés sont toutes valorisés	OK
Contrôler que la désérialisation d'un objet de type RevueAbonnementAExpiration s'effectue avec succès	Test unitaire lancé en désérialisant une chaine JSON venant de l'API REST représentant un objet de type RevueAbonnementAExpiration et validation des propriétés valorisés	L'objet RevueAbonnementAExpiration n'est pas null et les propriétés sont toutes valorisés	OK
Contrôler que la désérialisation d'un objet de type Revue s'effectue avec succès	Test unitaire lancé en désérialisant une chaine JSON venant de l'API REST représentant un objet de type Revue et validation des propriétés valorisés	L'objet Revue n'est pas null et les propriétés sont toutes valorisés	OK
Contrôler que la désérialisation d'un objet de type Suivi s'effectue avec succès	Test unitaire lancé en désérialisant une chaine JSON venant de l'API REST représentant un objet de type Suivi et validation des propriétés valorisés	L'objet Suivi n'est pas null et les propriétés sont toutes valorisés	OK
Contrôler que la désérialisation d'un objet de type Utilisateur s'effectue avec succès	Test unitaire lancé en désérialisant une chaine JSON venant de l'API REST représentant un objet de type Utilisateur et validation des propriétés valorisés	L'objet Utilisateur n'est pas null et les propriétés sont toutes valorisés	OK

Tests fonctionnels de l'interface utilisateur de l'application

But du test	Action de contrôle	Résultat attendu	Bilan
Contrôler que la recherche d'un livre selon le genre fonctionne de manière attendue	Après sélection de l'onglet livres, sélection du genre "Bande dessinée" et vérification du nombre de livres	5	OK

Tests fonctionnels dans Postman (fonctionnalités de l'API)

But du test	Action de contrôle	Résultat attendu	Bilan
Contrôler que l'accès à une URL invalide retourne un code d'erreur 404	Requête GET lancée sur http://url-api.com/	Code de la réponse : 404 Message de la réponse : ce point d'entrée de l'API n'existe pas	OK
Contrôler que la lecture des livres fonctionne	Requête GET lancée sur http://url-api.com/livre	Code de la réponse : 200 Le premier livre correspond	OK
Contrôler que la lecture des DVDs fonctionne	Requête GET lancée sur http://url-api.com/dvd	Code de la réponse : 200 Le premier DVD correspond	OK
Contrôler que la lecture des revues fonctionne	Requête GET lancée sur http://url-api.com/revue	Code de la réponse : 200 La première revue correspond	OK
Contrôler que la lecture des abonnements de revue arrivant à expiration fonctionne	Requête GET lancée sur http://url-api.com/revue_abonnement_expiration	Code de la réponse : 200 La première revue arrivant à expiration correspond	OK
Contrôler que la lecture des exemplaires d'un document fonctionne	Requête GET lancée sur http://url-api.com/exemplaire/{ "id": "10001" }	Code de la réponse : 200 Le premier exemplaire du document correspond	OK
Contrôler que la lecture des commandes d'un livre / DVD fonctionne	Requête GET lancée sur http://url-api.com/commandedocument/{ "id": "00010" }	Code de la réponse : 200 La première commande du livre / DVD correspond	OK
Contrôler que la lecture des abonnements d'une revue fonctionne	Requête GET lancée sur http://url-api.com/abonnement/{ "id": "10001" }	Code de la réponse : 200 Le premier abonnement de la revue correspond	OK
Contrôler que la lecture des genres fonctionne	Requête GET lancée sur http://url-api.com/genre	Code de la réponse : 200 Le premier genre correspond	OK
Contrôler que la lecture des publics fonctionne	Requête GET lancée sur http://url-api.com/public	Code de la réponse : 200 Le premier public correspond	OK
Contrôler que la lecture des rayons fonctionne	Requête GET lancée sur http://url-api.com/rayon	Code de la réponse : 200	OK

		Le premier rayon correspond	
Contrôler que la lecture des etats possibles d'une commande fonctionne	Requête GET lancée sur http://url-api.com/etat	Code de la réponse : 200 Le premier etat correspond	OK
Contrôler que la lecture des suivis possibles d'une commande fonctionne	Requête GET lancée sur http://url-api.com/suivi	Code de la réponse : 200 Le premier suivi correspond	OK
Contrôler que les authentifications fonctionnent	Requête PUT lancée sur http://url-api.com/authentification Champs : Login et Password valides	Code de la réponse : 200 L'utilisateur authentifié est retourné	OK
Contrôler que l'insertion d'un livre fonctionne	Requête POST lancée sur http://url-api.com/livre Champs : les champs pour ajouter un livre valide	Code de la réponse : 200 L'objet créé est retourné	OK
Contrôler que la modification d'un livre fonctionne	Requête PUT lancée sur http://url-api.com/livre Champs : les champs pour modifier un livre valide	Code de la réponse : 200 L'objet modifié est retourné	OK
Contrôler que la suppression d'un livre fonctionne	Requête DELETE lancée sur http://url-api.com/livre Champs : Id du livre à supprimer	Code de la réponse : 200 Le nombre de lignes supprimées est retourné	OK
Contrôler que l'insertion d'un DVD fonctionne	Requête POST lancée sur http://url-api.com/dvd Champs : les champs pour ajouter un DVD valide	Code de la réponse : 200 L'objet créé est retourné	OK
Contrôler que la modification d'un DVD fonctionne	Requête PUT lancée sur http://url-api.com/dvd Champs : les champs pour modifier un DVD valide	Code de la réponse : 200 L'objet modifié est retourné	OK
Contrôler que la suppression d'un DVD fonctionne	Requête DELETE lancée sur http://url-api.com/dvd Champs : Id du DVD à supprimer	Code de la réponse : 200 Le nombre de lignes supprimées est retourné	OK
Contrôler que l'insertion d'une revue fonctionne	Requête POST lancée sur http://url-api.com/revue Champs : les champs pour ajouter une revue valide	Code de la réponse : 200 L'objet créé est retourné	OK
Contrôler que la modification d'une revue fonctionne	Requête PUT lancée sur http://url-api.com/revue Champs : les champs pour modifier une revue valide	Code de la réponse : 200 L'objet modifié est retourné	OK
Contrôler que la suppression d'une revue fonctionne	Requête DELETE lancée sur http://url-api.com/revue Champs : Id de la revue à supprimer	Code de la réponse : 200 Le nombre de lignes supprimées est retourné	OK