

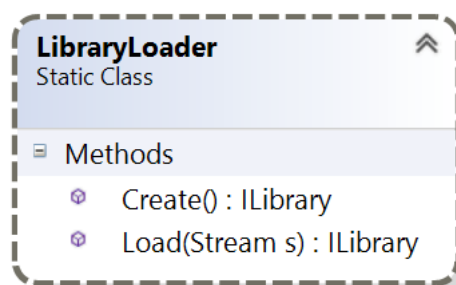
Exercice : La « DocLib »

1 Enoncé

L'objectif de cet exercice est d'implémenter un modèle de gestion d'une librairie de documents spécifié sous la forme de tests unitaires. Autrement dit : l'objectif de l'exercice est de faire passer en vert un jeu de tests.

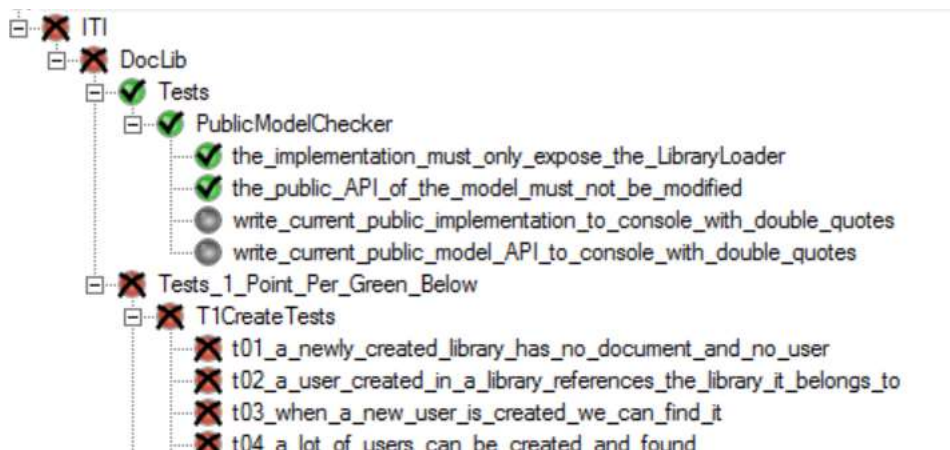
Vous disposez pour cela d'une solution comprenant 3 projets :

- ITI.DocLib.Tests, contient les tests unitaires
- ITI.DocLib.Model, contient 5 interfaces et 3 classes utilitaires (une exception et 2 objets d'évènements).
- ITI.DocLib.Impl contient une classe statique, le LibraryLoader qui est le seul point d'entrée pour manipuler le modèle. C'est cet assembly qui devra contenir toute l'implémentation.



Pour faire tourner les tests unitaires il vous suffit de configurer le projet ITI.DocLib.Tests en tant que projet de démarrage puis d'exécuter la solution.

Comme vous pouvez le constater, pour le moment, tous les tests sont rouges, à l'exception de 2 d'entre eux :

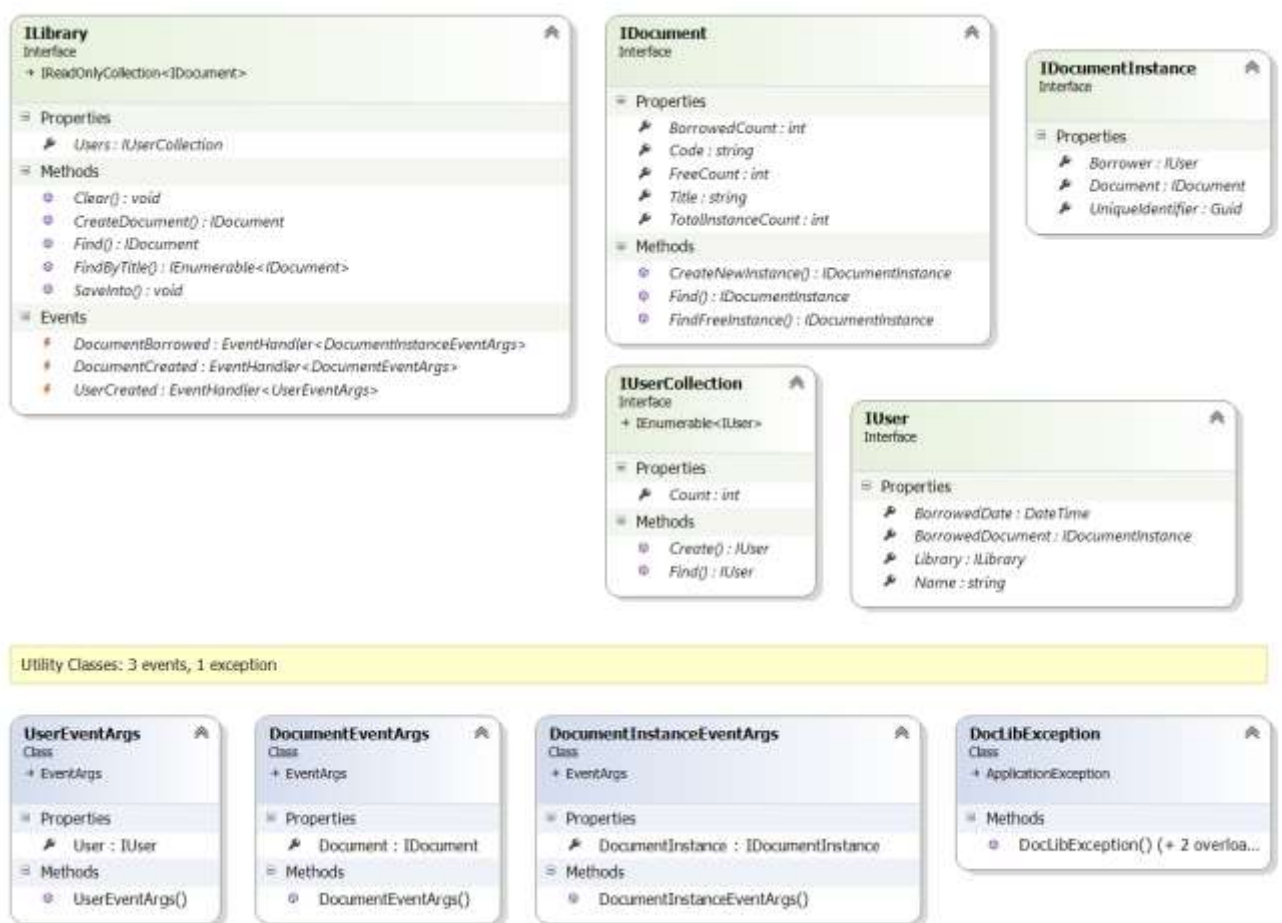


Ces deux tests analysent les objets implémentés dans ITI.DocLib.Model.dll et ITI.DocLib.Impl.dll et en extraient l'API public sous forme XML et compare cette API public avec celle d'origine qui est mémorisée : si d'aventure, vous altérez l'API public des objets (en ajoutant une méthode, une propriété ou un champ public), il passe au rouge : vous devez vous assurer qu'il reste au vert, votre implémentation ne doit modifier ni l'API publique (ITI.DocLib.Model), ni l'implémentation (ITI.DocLib.Impl ne doit avoir en public que la LibraryLoader).

Pour les autres tests, à vous de faire en sorte qu'ils passent en vert. Pour cela, vous avez le droit de faire ce que bon vous semble dans le projet ITI.DocLib.Impl

Les tests unitaires sont là pour spécifier de façon détaillée les fonctionnalités attendues. Cependant voici une courte description de ce qui est attendu :

- Une bibliothèque (*ILibrary*) contient :
 - o Des documents (*IDocument*).
 - o Des utilisateurs (*IUser*).
- A un document est associé un nombre quelconque d'exemplaires (*IDocumentInstance*, identifiés par un GUID) que les utilisateurs peuvent louer.
- Un utilisateur ne peut louer qu'un exemplaire à la fois.
- On peut sauver et restaurer une bibliothèque.



Il y a 29 tests à passer au vert. Attention, les 3 derniers tests ne sont pas simples à implémenter (c'est pourtant une fonctionnalité « classique » et bien utile), et le tout dernier est VRAIMENT subtile...



Au boulot !