

Section	Informations
Rôles et Responsabilités	Développement de la fonction : Valentin Test unitaire avec fixture : Maxime Test integration avec fixture : Maxime Test unitaire avec mock : Valentin
Stratégie de test	3 tests automatisé 1 unitaire avec un mock 1 unitaire avec une fixture 1 integration avec une fixture
Environnement de test	Application déployée sur un serveur physique : Pc portable Les tests ont été faire sur Visual Studio Code et le rapport est un fichier pdf
Cas de test	<p>Test BookListUnitTestFixture</p> <p>Description : Le test vérifie que que la réponse http soit 200 Prérequis : Le logiciel est installé et fonctionnel Etapes du test : 1. Création de 2 livres cls.book1 et cls.book2 2. Création d'un client http pour simuler les requêtes 3. Envoie d'une requête HTTP GET pour tester la réponse 4. Vérification de la réponse http Données : Données valides d'un livre Résultat attendu : La liste des livres s'affiche correctement et donc renvoie un code http 200 Conditions : C'est un test automatisé, l'interaction utilisateur doit être simulée</p> <p>Test BookListIntegrationTestFixture</p> <p>Description : Le test valide l'intégration entre la vue de la liste des livres et le modèle de données. Code http 200 Prérequis : Le logiciel est installé et fonctionnel Etapes du test : 1. Création de 2 livres self.book1 et self.book2 2. Création d'un client http pour simuler les requêtes 3. Envoie d'une requête HTTP GET pour obtenir l'URL associé à la vue book_list 4. Vérifie que la réponse soit un code http 200 / Vérifie que le text « Book 1 » et « Book 2 » soient dans la réponse html Données : Données valides d'un livre Résultat attendu : La liste des livres s'affiche correctement et donc renvoie un code http 200 Conditions : C'est un test automatisé, l'interaction utilisateur doit être simulée</p>

Test BookListUnitTestMock

Description : Le test utilise un mock pour simuler des livres dans la liste et obtenir un code http 200

Prérequis : Le logiciel est installé et fonctionnel

Etapes du test :

1. Le mock est défini comme une liste de deux objet 'Book'
2. Création d'un client http pour simuler les requêtes
3. Envoie d'une requête HTTP GET pour obtenir l'URL associé à la vue book_list
4. Vérifie que la réponse soit bien un code 200. / Vérifie que le template 'book_list.html' est utilisé pour générer la réponse et que la liste contient bien les deux livres créer avant « Mock Book 1 » et « Mock Book 2 »

Données : Données valides d'un livre

Résultat attendu : La liste des livres s'affiche correctement et donc renvoie un code http 200

Conditions : C'est un test automatisé, l'interaction utilisateur doit être simulée