

Projet Jeux Olympiques 2024

Rapport de tests

Créateur: Hugo ROGER

Test unitaires d'entités

Pour chaque DTO:

- UserDTO (test de création correcte de l'objet)
- OfferDTO (test de création correcte de l'objet)
- TicketDTO (test de création correcte de l'objet)
- PurchaseDTO (test de création correcte de l'objet)

Pour chaque entité de la base de données:

- User (7/7 méthodes testées)
- Offer (tests de création correcte de l'objet)
- Ticket (tests de création correcte de l'objet) (billets)
- Purchase (tests de création correcte de l'objet)

Pour chaque value objects des entités (qui sont ici surtout des tests de validation des données):

- OfferName (4/4 cas testés)
- OfferDescription (3/3 cas testés)
- EmailTest (4/4 cas testés)
- FirstName (3/3 cas testés)
- LastName (3/3 cas testés)
- Password (1/1 cas testés)
- UserKey (1/1 cas testés)
- PurchaseKey (1/1 cas testés)
- Price (3/3 cas testés) (prix/montant)

Tests des services

Pour chaque service associés à une entité:

- OfferService (7/7 méthodes publiques testées)
- UserService (4/4 méthodes publiques testées)

- TicketService (4/6 méthodes publiques testées (une des méthodes est un job et l'autre est non utilisée)
- PurchaseService (2/2 méthodes publiques testées)

Pour les services associés:

- QRCodeService (2/2 méthodes testées) (associé aux billets)
- StripeService (1/1 méthodes testées) (associé aux achats)

Pas de fichier de test n'a été implémenté pour TicketPDFService car sa méthode est déjà testée dans TicketServiceTest.

Test de routes

Trouvables dans doc/http. Testing des 8/8 les routes données (non page web) dans les contrôleurs:

Offres (utiles pour le dashboard admin):

- Lister les offres
- Obtenir une offre
- Créer une offre
- Mettre à jour une offre
- Supprimer une offre

Utilisateurs:

- Création de compte

Billet:

- Génération du PDF contenant les e-billets

Achats

- Paiement par Stripe (qui sauvegarde également de nouveaux billets et un nouvel achat)