



Jeux Olympiques

Rapport de couverture des tests automatisés

Rapport de couverture de tests

Contexte

Les tests automatisés du projet `olympique_tickets` ont été réorganisés et étendus afin de vérifier les fonctionnalités clés autour des utilisateurs, des permissions, des formulaires et des utilitaires. Un script dédié (`tools/run_trace_coverage.py`) permet désormais de mesurer facilement la couverture à l'aide du module standard `trace`.

Procédure d'exécution

Tests unitaires

`bash python manage.py test` Résultat actuel : 18 tests exécutés avec succès.

Rapport de couverture

`bash python tools/run_trace_coverage.py` Cette commande :

3. exécute la suite de tests,
4. calcule la couverture (en se concentrant sur `tickets_bah` et `appAdmin`),
5. produit un résumé dans la console,

6. enregistre un rapport détaillé dans `trace_coverage.json`.

Résultats de couverture (trace)

Module / Fichier	Lignes	Exécutées	Manquantes	Couverture
tickets_bah/core/permissions.py	43	36	7	83,7 %
tickets_bah/views.py	260	58	202	22,3 %
appAdminNews.py	237	38	199	16,0 %
olympique_tickets_bah/settings.py	78	69	9	88,5 %
olympique_tickets_bah/test_runner.py	8	8	0	100,0 %
tickets_bah/tests/test_models.py	69	69	0	100,0 %
tickets_bah/tests/test_utils.py	35	35	0	100,0 %
tickets_bah/tests/test_permissions.py	51	51	0	100,0 %
tickets_bah/tests/test_decorators.py	27	27	0	100,0 %

tickets_bah/tests/test_forms.py	24	24	0	100,0 %
Total projet (tickets_bah + appAdmin)	810	136	674	16,8 %

Le fichier `trace_coverage.json` contient l'ensemble des modules analysés avec la liste des lignes manquantes pour un travail de couverture plus fin.

Recommandations

1. **Modèles** `tickets_bah/models.py` : ajouter des tests pour les managers personnalisés, les méthodes liées aux réservations et au panier, ainsi que la génération de QR-code.
2. **Utilitaires** `tickets_bah/utils.py` : compléter les tests pour couvrir tous les cas (y compris la journalisation des erreurs lors d'un échec d'envoi).
3. **Application** `appAdmin` : écrire des tests fonctionnels simples sur les vues principales ou, en attendant, exclure explicitement l'app de la couverture si elle n'est pas prioritaire.
4. **Outil de couverture plus riche** : envisager `coverage.py` (HTML, branches) dès que l'installation de dépendances supplémentaires est possible.

Fichiers générés

- `tools/run_trace_coverage.py` : script de génération du rapport.
- `trace_coverage.json` : rapport détaillé de la dernière exécution.

Ce document peut être partagé tel quel ou exporté en PDF (via un éditeur Markdown) pour diffusion.