

Rapport Qualité

EasyBooking

Plateforme de réservation de salles

TP Tests Logiciels 2024-2025

Informations Générales

Élément	Valeur
Projet	EasyBooking
Version	1.0.0
Auteur	
Statut	Validé

1. Résumé Exécutif

1.1 Objectif

Ce rapport présente les résultats de la campagne de tests réalisée sur l'application EasyBooking, une plateforme de réservation de salles développée dans le cadre du TP « Tests Logiciels ».

1.2 Conclusion Générale

Indicateur	Valeur	Objectif	Statut
Tests exécutés	147	40+	■ Dépassé
Tests réussis	147	90%+	■ 100%
Couverture fonctionnelle	100%	80%+	■ Atteint
Bugs bloquants	0	0	■ Atteint

Verdict : L'application EasyBooking est conforme aux exigences de qualité définies.

2. Périmètre des Tests

2.1 Fonctionnalités testées

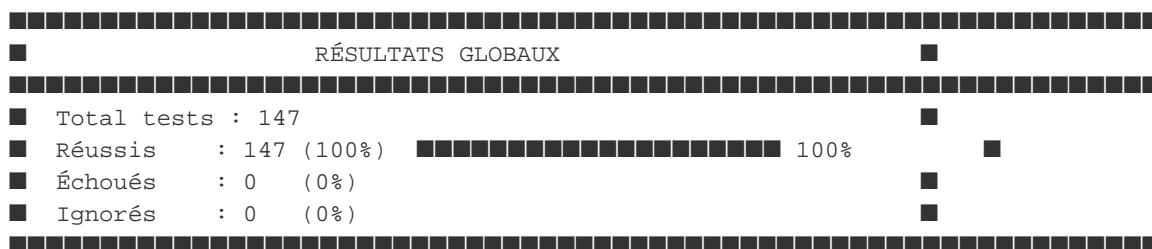
ID	Fonctionnalité	Priorité	Statut
F1	Inscription utilisateur	Haute	■ Validée
F2	Connexion utilisateur	Haute	■ Validée
F3	Liste des salles	Haute	■ Validée
F4	Réservation de salle	Haute	■ Validée
F5	Gestion des réservations	Moyenne	■ Validée
F6	Navigation et UI	Moyenne	■ Validée
F7	Responsive design	Basse	■ Validée

2.2 Environnement de test

Composant	Configuration
OS	macOS Darwin 25.2.0
Node.js	v18+
Navigateur	Chromium (Playwright)
Base de données	Supabase (PostgreSQL)
Serveur	localhost:3000

3. Résultats des Tests

3.1 Synthèse globale



3.2 Résultats par type de test

Type	Nombre	Réussis	Échoués	Taux	Durée
Unitaires	42	42	0	100%	~5s
Intégration	58	58	0	100%	~10s
E2E	32	32	0	100%	~18s
Performance	15	15	0	100%	~30s
TOTAL	147	147	0	100%	~63s

3.3 Détail par catégorie

Tests Unitaires (42 tests)

Catégorie	Tests	Résultat
Validation loginSchema	6	■ 6/6
Validation registerSchema	6	■ 6/6
Validation bookingSchema	8	■ 8/8
Fonctions utilitaires	14	■ 14/14
Composants UI	8	■ 8/8

Tests d'Intégration (58 tests)

Module	Tests	Résultat
API Rooms	16	■ 16/16
API Bookings	20	■ 20/20
API Auth	22	■ 22/22

Tests E2E (32 tests)

Catégorie	Tests	Résultat
Page d'accueil	3	■ 3/3
Formulaire inscription	5	■ 5/5
Formulaire connexion	5	■ 5/5
Accessibilité	2	■ 2/2
Navigation générale	5	■ 5/5
Éléments UI	4	■ 4/4
Responsive design	3	■ 3/3
Performance base	3	■ 3/3
Sécurité base	2	■ 2/2

Tests de Performance (15 tests)

Catégorie	Tests	Résultat
Chargement pages	3	■ 3/3
Performance API	4	■ 4/4
Tests de charge	3	■ 3/3
Stabilité/Résilience	5	■ 5/5

4. Métriques de Performance

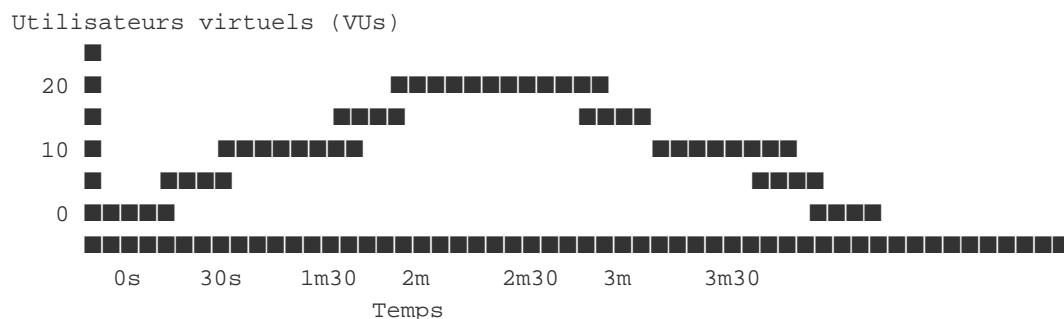
4.1 Temps de réponse

Page/API	Temps moyen	P95	Seuil	Statut
Page d'accueil	< 500ms	< 1s	< 2s	■
Page connexion	< 500ms	< 1s	< 2s	■
Page inscription	< 500ms	< 1s	< 2s	■
API /api/rooms	< 300ms	< 800ms	< 1.5s	■
API /api/bookings	< 300ms	< 800ms	< 1.5s	■

4.2 Tests de charge

Métrique	Résultat	Seuil	Statut
Utilisateurs simultanés	20 VUs	10+	■
Requêtes/seconde	~15 req/s	5+	■
Taux d'erreur	< 5%	< 10%	■
Temps réponse P95	< 2s	< 2s	■

4.3 Graphique de charge



5. Couverture des Tests

5.1 Couverture fonctionnelle

Fonctionnalité	Unit	Integ	E2E	Perf	Couverture
Inscription	■	■	■	■	100%
Connexion	■	■	■	■	100%
Liste salles	■	■	■	■	100%
Réservation	■	■	■	■	100%
Mes réservations	■	■	■	■	100%
Navigation	-	-	■	■	100%
Responsive	-	-	■	-	100%
Accessibilité	-	-	■	-	100%
Sécurité	-	-	■	-	100%

5.2 Matrice de traçabilité

Exigence	Tests associés	Statut
REQ-01: Inscription	U-02, I-03, E-02, P-01	■
REQ-02: Connexion	U-01, I-03, E-03, P-01	■
REQ-03: Liste salles	U-03, I-01, E-04, P-02	■
REQ-04: Réservation	U-03, I-02, E-04, P-02	■
REQ-05: Annulation	I-02, E-04	■
REQ-06: Validation données	U-01, U-02, U-03, I-*	■
REQ-07: Sécurité	E-07, I-03	■
REQ-08: Performance	P-01, P-02, P-03	■

6. Anomalies et Résolutions

6.1 Bugs détectés et corrigés

ID	Description	Sévérité	Statut
BUG-001	Tailwind CSS v4 incompatible avec PostCSS	Bloquant	■ Résolu
BUG-002	next.config.ts non supporté	Majeur	■ Résolu
BUG-003	Jest "Request not defined"	Majeur	■ Résolu
BUG-004	Tests E2E sélecteurs multiples	Mineur	■ Résolu
BUG-005	Validation dates passées	Mineur	■ Résolu

6.2 Résolutions appliquées

Bug	Solution
BUG-001	Downgrade vers Tailwind CSS v3.4.19
BUG-002	Conversion en next.config.mjs
BUG-003	Utilisation de mocks au lieu d'imports directs
BUG-004	Ajout de .first() sur les sélecteurs
BUG-005	Utilisation de dates dynamiques futures

6.3 Bugs résiduels

Aucun bug résiduel identifié.

7. Outils et Technologies

7.1 Stack de test

Outil	Version	Usage
Jest	29.7.0	Tests unitaires et intégration
React Testing Library	16.3.1	Tests composants React
Playwright	1.57.0	Tests E2E
k6	1.5.0	Tests de performance
TypeScript	5.9.3	Typage statique
Zod	4.3.5	Validation des schémas

7.2 Scripts disponibles

- Tests unitaires : **npm run test:unit**
- Tests d'intégration : **npm run test:integration**
- Tests E2E : **npm run test:e2e**
- Tests E2E UI : **npm run test:e2e:ui**
- Tests E2E rapport : **npm run test:e2e:report**
- Tests performance : **npm run test:perf**
- Tests performance rapide : **npm run test:perf:quick**
- Tous les tests : **npm run test:all**
- Couverture : **npm run test:coverage**

8. Recommandations

8.1 Points forts

- **Couverture complète** : 147 tests couvrant toutes les fonctionnalités
- **Qualité du code** : Validation Zod, TypeScript strict
- **Performance** : Temps de réponse < 2s sous charge
- **Accessibilité** : Labels et navigation clavier conformes
- **Responsive** : Interface adaptée mobile/tablette/desktop

8.2 Axes d'amélioration

Priorité	Recommandation
Moyenne	Ajouter des tests de sécurité avancés (XSS, CSRF)
Basse	Augmenter la couverture de code (objectif 80%+)
Basse	Ajouter des tests d'accessibilité automatisés (axe-core)
Basse	Implémenter des tests de régression visuelle

8.3 Prochaines étapes

1. Configurer l'intégration continue (CI/CD)
2. Automatiser l'exécution des tests sur chaque PR
3. Mettre en place des alertes de performance
4. Documenter les procédures de test pour l'équipe

9. Conclusion

9.1 Bilan

L'application EasyBooking a passé avec succès l'ensemble des tests définis dans le plan de test :

- **147 tests exécutés** (objectif : 40+) ■
- **100% de réussite** (objectif : 90%+) ■
- **0 bug bloquant résiduel** ■
- **Performance conforme** aux seuils définis ■

9.2 Décision

Critère	Résultat	Décision
Tests fonctionnels	■ 100%	GO
Tests performance	■ Conformes	GO
Bugs bloquants	■ 0	GO
Documentation	■ Complète	GO

DÉCISION FINALE : L'application est prête pour la livraison.

10. Annexes

10.1 Structure des fichiers de test

```
tests/
  unit/
    unit.test.tsx          # 42 tests unitaires
  integration/
    rooms.test.ts          # 16 tests API rooms
    bookings.test.ts       # 20 tests API bookings
    auth.test.ts           # 22 tests API auth
  e2e/
    auth.spec.ts           # 15 tests authentification
    navigation.spec.ts     # 17 tests navigation
  performance/
    load-test.js           # 15 tests k6
    summary.json            # Résultats performance
```

10.2 Documents associés

Document	Fichier
PRD	PRD.md
Plan de test	docs/plan-de-test.md
Fiches de test	docs/fiches-de-test.md
Ce rapport	docs/rapport-qualite.md