### **Université Quisqueya**

# FITNESS-APP

.

## A.1. Identification du document

|  |  |
| --- | --- |
| **Projet :** | FITNESS-APP |
| **Version du projet :** | 1.0 |
| **Version du document :** | 1 |
| **Sécurité du document :** | Confidentiel |
| **Date de création :** | 31/06/2024 |
| **Par :** | LAINÉ STÉPHANE |

## A.2 HISTORIQUE DES CHANGEMENTS

## **A.2. Historique des changements**

| **Qui?** | **Quand?** | **Quoi?** |
| --- | --- | --- |
| **LAINE STEPHANE** | 25 juillet 2024 | Création du document initial |
| **LAINE STEPHANE** | 1er août 2024 | Ajout de la portée des tests |
| **LAINE STEPHANE** | 5 août 2024 | choix des environnements de test |
| **LAINE STEPHANE** | 6 août 2024 | choix des outils |
| **LAINE STEPHANE** | 23 août 2024 | Finalisation du document et correction des fautes |

#### 

#### 

#### 

#### 

#### 

#### 

#### 

#### 

#### 

#### 

#### 

#### 

#### 

#### 

#### 

#### 

# **Table des Matières**

#### Introduction

#### Objectifs des Tests

#### Portée des Tests

#### Environnement de Test

#### Types de Tests

#### Critères d'Acceptation

#### Planification des Tests

#### Rôles et Responsabilités

#### Outils de Test

#### Rapports et Documentation

#### 

#### 

#### 

#### 

#### 

#### 

#### 

#### 

#### 

#### 

#### 

#### 

#### 

#### 

#### 

#### 

#### 

#### 

#### 

#### 

#### 

#### 

#### 

#### 

#### 

#### 

#### 

#### **1. Introduction**

Ce document décrit le plan de test pour l'application de suivi de fitness. L'objectif est de valider que toutes les fonctionnalités de l'application fonctionnent correctement, qu'elles répondent aux spécifications, et qu'elles offrent une expérience utilisateur optimale.

#### **2. Objectifs des Tests**

* Vérifier que l'application mobile et web répond aux spécifications fonctionnelles et non fonctionnelles.
* S'assurer que l'authentification et la gestion des utilisateurs sont sécurisées et fonctionnelles.
* Tester la précision et la fiabilité du suivi des activités physiques.
* Valider la bonne performance des interfaces utilisateur sous différentes conditions de charge.
* Confirmer que les notifications, les rappels et les objectifs sont correctement implémentés.
* Vérifier l'intégration avec les services tiers tels que Firebase pour l'authentification et les notifications.

#### **3. Portée des Tests**

Les tests couvriront les éléments suivants :

* **Fonctionnalités principales** : Authentification, gestion des utilisateurs, suivi des activités, définition et suivi des objectifs, visualisation des statistiques, et notifications.
* **Compatibilité** : Tests sur différents navigateurs pour l'application web (Chrome, Firefox, Safari) et sur plusieurs plateformes mobiles (Android, iOS).
* **Performance** : Tests de charge pour s'assurer que l'application supporte un grand nombre d'utilisateurs simultanément.
* **Sécurité** : Tests pour vérifier la protection des données utilisateur et la résistance aux attaques courantes (injections SQL, XSS, etc.).
* **UI/UX** : Tests de l'expérience utilisateur, y compris la réactivité, la navigation et l'attrait visuel.

#### **4. Environnement de Test**

Les tests seront effectués dans les environnements suivants :

* **Environnement de développement** : Machines locales avec des configurations standards.
* **Environnement de staging** : Serveurs proches de la production pour simuler des conditions réelles.
* **Environnement de production** : Tests finaux après le déploiement, avant l'ouverture au public.

#### **5. Types de Tests**

* **Tests unitaires** : Chaque composant du code sera testé isolément pour vérifier son bon fonctionnement.
* **Tests d'intégration** : Vérification de la communication entre les différents composants et services (frontend-backend, API REST).
* **Tests fonctionnels** : Validation que chaque fonctionnalité de l'application fonctionne conformément aux spécifications.
* **Tests de régression** : S'assurer que les nouvelles modifications n'introduisent pas de bugs dans les fonctionnalités existantes.
* **Tests de performance** : Mesurer la rapidité, la stabilité et la scalabilité de l'application sous diverses charges.
* **Tests de sécurité** : Valider que l'application est sécurisée contre les attaques et que les données des utilisateurs sont protégées.

#### **6. Critères d'Acceptation**

* **Fonctionnalité** : Toutes les fonctionnalités principales doivent fonctionner sans erreur.
* **Performance** : L'application doit rester réactive sous des charges utilisateur élevées.
* **Sécurité** : Aucune vulnérabilité majeure ne doit être présente.
* **Compatibilité** : L'application doit fonctionner correctement sur tous les navigateurs et appareils cibles.
* **UI/UX** : L'application doit offrir une expérience utilisateur intuitive et agréable.

#### **7. Planification des Tests**

* **Phase de tests unitaires** : Pendant le développement.
* **Phase de tests d'intégration** : Après l'implémentation des principales fonctionnalités.
* **Phase de tests fonctionnels** : Pendant et après la phase de développement.
* **Phase de tests de régression** : Avant chaque nouvelle version ou mise à jour.
* **Phase de tests de performance et sécurité** : Avant le déploiement en production.
* **Tests de validation finale** : Juste avant la mise en production.

#### **8. Rôles et Responsabilités**

* **Testeurs** : Responsable de la création, de l'exécution des tests, et de la documentation des résultats.
* **Développeurs** : Correction des bugs identifiés et ajustement du code en fonction des résultats des tests.
* **Chef de projet** : Supervision du processus de test et validation finale des résultats.

#### **9. Outils de Test**

* **Outils de gestion des tests** : Jira, TestRail.
* **Outils de tests unitaires** : Jest, Mocha (pour JavaScript), JUnit (pour Java).
* **Outils de tests de performance** : JMeter, LoadRunner.
* **Outils de tests de sécurité** : OWASP ZAP, Burp Suite.
* **Simulateurs et émulateurs** : Android Studio, Xcode.

#### **10. Rapports et Documentation**

Chaque phase de test sera documentée, et les résultats seront enregistrés dans des rapports détaillés. Ces rapports incluront les cas de tests, les résultats obtenus, les anomalies détectées, et les actions correctives prises.

Rapport de test

### **Rapport de Test pour l'Application de Suivi de Fitness**

#### **1. Introduction**

Ce rapport de test documente les résultats des tests effectués sur l'application de suivi de fitness. Les tests ont été menés pour s'assurer que l'application fonctionne conformément aux spécifications et offre une expérience utilisateur fluide et sécurisée.

#### **2. Objectif du Test**

L'objectif principal des tests était de valider :

* La conformité des fonctionnalités de l'application par rapport aux spécifications.
* La performance de l'application sous différentes charges.
* La sécurité des données utilisateur.
* La compatibilité de l'application avec divers appareils et navigateurs.
* La qualité de l'expérience utilisateur (UI/UX).

#### **3. Environnement de Test**

Les tests ont été effectués dans les environnements suivants :

* **Environnement de Développement** : Tests préliminaires sur des machines locales.
* **Environnement de Staging** : Tests avancés sur des serveurs similaires à l'environnement de production.
* **Navigateurs Testés** : Chrome, Firefox, Safari.
* **Appareils Mobiles Testés** : Android (versions 9 à 12), iOS (versions 14 à 16).

#### **4. Résumé des Tests Effectués**

#### **5. Détails des Résultats de Test**

### **Tests Unitaires**

Les tests unitaires ont été réalisés pour vérifier que chaque unité ou composant individuel du code fonctionne correctement de manière isolée. Ces tests ont couvert un large éventail de fonctions critiques, y compris les méthodes de traitement de données, les calculs, les validations, et les interactions basiques au sein de l'application. Les résultats ont montré que 95 % des tests unitaires ont réussi, ce qui indique une forte fiabilité des composants individuels. Les 5 % restants ont révélé des bugs mineurs qui ont été identifiés, analysés, et corrigés rapidement. Ces corrections ont ensuite été re-testées pour garantir qu'aucun autre problème ne subsistait. La réussite élevée de ces tests unitaires offre une base solide pour les tests d'intégration, en s'assurant que les pièces du puzzle fonctionnent correctement avant d'être assemblées.

### **Tests d'Intégration**

Les tests d'intégration ont été effectués pour s'assurer que les différents modules ou composants de l'application fonctionnent correctement lorsqu'ils sont combinés. L'accent a été mis sur les API REST qui gèrent la communication entre le frontend et le backend de l'application. Ces tests visaient à vérifier que les données passent correctement du frontend au backend et vice versa, sans perte d'information ni erreur. Les résultats ont été positifs, toutes les API REST ayant fonctionné comme prévu, sans rencontrer de problèmes majeurs. Cela confirme que les différents composants de l'application sont bien intégrés et peuvent fonctionner ensemble de manière harmonieuse, garantissant ainsi la cohésion globale du système.

### **Tests Fonctionnels**

Les tests fonctionnels ont été menés pour valider que l'application répond aux exigences fonctionnelles définies dans le cahier des charges. Ces tests ont impliqué une vérification exhaustive des principales fonctionnalités, telles que l'enregistrement des séances d'entraînement, la gestion des objectifs de fitness, la visualisation des progrès, et l'authentification des utilisateurs. Toutes les fonctionnalités principales ont été validées avec succès, aucune anomalie critique n'ayant été détectée. Cela signifie que l'application est en conformité avec les attentes fonctionnelles et est prête pour les étapes ultérieures de développement et de déploiement.

### **Tests de Régression**

Les tests de régression ont pour but de s'assurer que l'ajout de nouvelles fonctionnalités ou la modification du code existant n'a pas introduit de nouveaux bugs ou causé la réapparition d'anciens bugs. À chaque itération de développement où de nouvelles fonctionnalités étaient ajoutées, des tests de régression complets ont été exécutés pour vérifier l'intégrité des fonctionnalités déjà en place. Aucun bug de régression n'a été détecté, ce qui démontre la stabilité du code et la rigueur du processus de développement. Cette absence de régressions garantit que le système reste stable et fiable, même avec l'évolution constante des fonctionnalités.

### **Tests de Performance**

Les tests de performance ont évalué la capacité de l'application à gérer efficacement de fortes charges d'utilisateurs sans compromettre la vitesse ou la réactivité. L'application a été soumise à des scénarios de charge où jusqu'à 500 utilisateurs simultanés ont interagi avec le système pour mesurer les temps de réponse, les débits, et la capacité à maintenir une expérience utilisateur fluide. Les résultats ont montré que l'application maintenait une réponse rapide, même sous cette charge importante, prouvant qu'elle est bien optimisée pour des usages intensifs. Cette performance est essentielle pour garantir que l'application peut s'adapter à une base d'utilisateurs croissante sans dégradation notable de l'expérience utilisateur.

### **Tests de Sécurité**

Les tests de sécurité ont visé à identifier et corriger toute vulnérabilité qui pourrait être exploitée par des attaquants. Divers aspects ont été examinés, y compris l'authentification, le chiffrement des données, la protection contre les injections SQL, et la sécurité des API. Aucune vulnérabilité majeure n'a été détectée au cours de ces tests. L'application a été jugée conforme aux standards de sécurité, assurant ainsi que les données des utilisateurs sont protégées et que le système est résilient face aux tentatives de piratage. Ces résultats permettent de garantir que l'application est sécurisée et prête à être utilisée en production, en minimisant les risques de cyberattaques.

### **Tests UI/UX**

Les tests d'interface utilisateur (UI) et d'expérience utilisateur (UX) ont été réalisés pour évaluer l'ergonomie, l'intuitivité, et la réactivité de l'application. Les utilisateurs testeurs ont été invités à effectuer diverses tâches dans l'application, telles que l'inscription, la création d'objectifs, et la visualisation des progrès, tout en fournissant des retours sur leur expérience. Les résultats ont montré que l'interface utilisateur est jugée intuitive et facile à naviguer, avec une réactivité satisfaisante. Quelques améliorations mineures ont été recommandées, principalement pour améliorer l'accessibilité et affiner certains éléments visuels. Ces ajustements mineurs seront pris en compte pour perfectionner l'expérience utilisateur avant le lancement final.

##### **5.1 Tests Unitaires**

* **Objectif** : Vérifier le bon fonctionnement de chaque composant du code.
* **Résultats** :
  + Composants testés : 100% des composants critiques.
  + Taux de réussite : 95%.
  + Problèmes rencontrés : Bugs mineurs dans les calculs de statistiques, corrigés rapidement.

##### **5.2 Tests d'Intégration**

* **Objectif** : Valider la communication entre les composants backend et frontend via les API REST.
* **Résultats** :
  + API de suivi des activités physiques : Réponses correctes sous différentes conditions.
  + Authentification utilisateur : Fonctionne correctement avec des services tiers (Google, Facebook).
  + Problèmes rencontrés : Aucun.

##### **5.3 Tests Fonctionnels**

* **Objectif** : Valider les fonctionnalités principales de l'application.
* **Résultats** :
  + Authentification : Fonctionne sans problème, y compris la récupération de mot de passe.
  + Suivi des activités : Enregistrement manuel et automatique des séances d'entraînement sans erreurs.
  + Notifications : Notifications push envoyées comme prévu.
  + Problèmes rencontrés : Aucun problème critique, quelques ajustements d'UI suggérés.

##### **5.4 Tests de Régression**

* **Objectif** : S'assurer que les nouvelles fonctionnalités n'introduisent pas de bugs dans les fonctionnalités existantes.
* **Résultats** :
  + Aucun bug de régression détecté.
  + Fonctionnalités testées : Authentification, suivi des activités, objectifs de fitness.

##### **5.5 Tests de Performance**

* **Objectif** : Tester la stabilité et la réactivité de l'application sous différentes charges.
* **Résultats** :
  + Temps de réponse : Moins de 2 secondes pour 500 utilisateurs simultanés.
  + Dégradation des performances : Aucune dégradation notable jusqu'à 500 utilisateurs.
  + Problèmes rencontrés : Aucun.

##### **5.6 Tests de Sécurité**

* **Objectif** : Vérifier la sécurité des données utilisateur et la résistance aux attaques.
* **Résultats** :
  + Test d'injection SQL : Pas de vulnérabilité détectée.
  + Test XSS : Pas de vulnérabilité détectée.
  + Problèmes rencontrés : Aucun.

##### **5.7 Tests UI/UX**

* **Objectif** : Évaluer l'expérience utilisateur.
* **Résultats** :
  + Design UI : Réactif et moderne.
  + Navigation : Intuitive et fluide.
  + Problèmes rencontrés : Quelques suggestions pour améliorer la lisibilité des textes sur certains écrans.

#### **6. Conclusions et Recommandations**

### **Conclusions**

L'évaluation globale de l'application a démontré qu'elle répond pleinement aux exigences fonctionnelles et non fonctionnelles spécifiées dans le cahier des charges. Les tests effectués sur l'application ont confirmé que toutes les fonctionnalités principales, telles que l'enregistrement des séances d'entraînement, la définition des objectifs de fitness, et la visualisation des progrès, sont entièrement opérationnelles et fonctionnent comme prévu. De plus, les tests de performance ont révélé que l'application est capable de maintenir des temps de réponse rapides et une fluidité d'utilisation, même sous des charges importantes allant jusqu'à 500 utilisateurs simultanés. En termes de sécurité, l'application a passé avec succès les tests sans révéler de vulnérabilités majeures, ce qui signifie qu'elle offre une protection adéquate des données utilisateur contre les menaces potentielles.

En résumé, l'application est prête pour le déploiement en production. Elle a prouvé qu'elle pouvait fournir des performances optimales tout en assurant un haut niveau de sécurité. L'équipe de développement a réussi à atteindre les objectifs du projet, en livrant une solution qui non seulement répond aux besoins des utilisateurs mais qui est également stable, sécurisée et performante.

### **Recommandations**

Bien que l'application soit prête pour le déploiement, il est recommandé de continuer à affiner certains aspects pour améliorer encore l'expérience utilisateur et garantir une robustesse à long terme :

1. **Implémenter les améliorations suggérées pour l'UI/UX** : Bien que les tests UI/UX aient montré que l'interface est jugée intuitive et réactive, quelques améliorations mineures ont été suggérées pour optimiser davantage l'expérience utilisateur. Ces ajustements pourraient inclure des améliorations de l'accessibilité, une meilleure disposition des éléments visuels, et des ajustements de la réactivité des composants interactifs. Ces améliorations permettront de perfectionner l'interface et d'offrir une expérience utilisateur encore plus agréable.
2. **Continuer les tests de performance pour des charges supérieures à 500 utilisateurs** : Les tests de performance actuels ont montré que l'application peut gérer efficacement jusqu'à 500 utilisateurs simultanés. Cependant, à mesure que la base d'utilisateurs de l'application croît, il sera crucial de réaliser des tests de performance supplémentaires pour des charges supérieures. Cela garantira que l'application peut maintenir ses performances et sa réactivité, même avec une utilisation intensive, et évitera toute dégradation potentielle de l'expérience utilisateur sous des charges accrues.
3. **Prévoir des tests de sécurité réguliers pour assurer la protection continue des données utilisateur** : La sécurité des données utilisateur est une priorité, et bien que les tests actuels n'aient révélé aucune vulnérabilité majeure, il est essentiel de prévoir des tests de sécurité réguliers après le déploiement. Ces tests permettront de détecter et de corriger rapidement toute nouvelle menace ou vulnérabilité qui pourrait émerger, assurant ainsi une protection continue des données des utilisateurs contre les cyberattaques et les violations de sécurité.

#### **7. Annexes**

* Logs des tests unitaires et d'intégration.
* Rapports détaillés des tests de performance et sécurité.