**TEST PLAN**

**Fitness Tracking Application**

**Created By**

**Vignesher Gandhi**

## 

**Document Revision History**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **DATE** | **VERSION** | **DESCRIPTION** | **AUTHOR** | **REVIEWER** | **APPROVER** |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |

[1. Introduction 2](#_Toc133624402)

[1.1 Purpose 2](#_Toc133624403)

[1.2 Scope 2](#_Toc133624404)

[1.3 Objectives 2](#_Toc133624405)

[2. Test Environment 2](#_Toc133624406)

[2.1 Hardware 2](#_Toc133624407)

[3. Test Strategy 2](#_Toc133624408)

[3.1 Functional Testing 2](#_Toc133624409)

[3.2 Usability Testing 3](#_Toc133624410)

[3.3 Performance Testing 3](#_Toc133624411)

[4. Test Cases 3](#_Toc133624412)

[4.1 Functional Test Cases 3](#_Toc133624413)

[4.2 Usability Test Cases 3](#_Toc133624414)

[5. Test Execution 3](#_Toc133624415)

[5.1 Test Environment 3](#_Toc133624416)

[5.2 Hardware Requirements 4](#_Toc133624417)

[5.2 Test Execution Strategy 4](#_Toc133624418)

[5.2.1 Manual Testing 5](#_Toc133624419)

[5.2.2 Automated Testing 5](#_Toc133624420)

[5.3 Test Reporting 4](#_Toc133624421)

[6. Features to Be Tested 4](#_Toc133624422)

[7. Entry and Exit Criteria 4](#_Toc133624423)

[7.1 Entry Criteria: 4](#_Toc133624424)

[7.2 Exit Criteria: 5](#_Toc133624425)

[8. Risk Assumptions 5](#_Toc133624426)

[6. Test Summary Report 5](#_Toc133624427)

[6.1 Defect Tracking and Management 6](#_Toc133624428)

[6.2 Test Closure 6](#_Toc133624429)

[7. Conclusion 6](#_Toc133624430)

# 1. Introduction

## 1.1 Purpose

The purpose of this test plan is to ensure that the fitness tracking mobile app meets its requirements, functions properly, is user-friendly and performs well under the expected load.

## 1.2 Scope

This test plan covers the functional, usability, and performance testing of the fitness-tracking mobile app.

## 1.3 Objectives

The objectives of this test plan are to:

* Identify defects and ensure the functionality of the app
* Ensure the app is user-friendly and easy to use
* Ensure the app performs well under the expected load
* Provide a comprehensive report of the testing results
* Collaborate with the development team to prioritize and resolve defects

# 2. Test Environment

## 2.1 Hardware

The fitness-tracking mobile app will be tested on the following hardware:

* Android: version 9.0 or later
* iOS: version 11.0 or later 2.2 Software The fitness tracking mobile app will be tested on the following software:
* Android Studio: version 4.0 or later
* Xcode: version 12.0 or later
* Appium: version 1.20.2 or later
* Java: version 8 or later

# 3. Test Strategy

## 3.1 Functional Testing

Functional testing will ensure that the app's features work as intended. The following functional testing will be conducted:

* Logging workouts with exercises, sets, and reps
* Tracking progress over time and setting goals for fitness and nutrition
* Connecting with friends and sharing progress and achievements
* Accessing a library of exercises and workout plans
* Integrating with other fitness apps and devices

## 3.2 Usability Testing

Usability testing will ensure that the app is user-friendly and easy to use. The following usability testing will be conducted:

* Navigation and ease of use for all features
* Clarity of instructions and labels
* User satisfaction with the overall experience

## 3.3 Performance Testing

Performance testing will ensure that the app performs well under the expected load. The following performance testing will be conducted:

* Load testing to simulate high user traffic
* Stress testing to simulate heavy user loads
* Response time testing to ensure timely responses
* Battery and memory usage testing to ensure efficient use of resources

# 4. 5.Test Cases

## 4.1 Functional Test Cases

* Verify that the app allows users to log workouts with exercises, sets, and reps
* Verify that the app allows users to track progress over time and set goals for fitness and nutrition
* Verify that the app allows users to connect with friends and share progress and achievements
* Verify that the app allows users to access a library of exercises and workout plans
* Verify that the app allows users to integrate with other fitness apps and devices

## 4.2 Usability Test Cases

* Verify that the app is easy to navigate and use for all features
* Verify that the instructions and labels are clear and concise
* Verify that users are satisfied with the overall experience 4.3 Performance Test Cases
* Verify that the app performs well under high user traffic load
* Verify that the app performs well under heavy user load stress
* Verify that the app has fast response times
* Verify that the app uses resources efficiently (battery and memory)

# 5. Test Execution

## 5.1 Test Environment

The fitness tracking mobile app will be tested in the following environment:

* Operating System: Android 9.0
* Devices: Google Pixel 3, Nexus S
* Appium Version: 1.20.2
* Java Development Kit: 1.8
* Integrated Development Environment: Eclipse

## 5.2 Hardware Requirements

* Mobile devices (smartphones and tablets) with Android Version 9.0
* A desktop or laptop computer running the appropriate development environment, such as Xcode for iOS or Android Studio for Android
* USB cables for connecting mobile devices to the computer for debugging and testing purposes
* Adequate storage space for storing test data and results

## 5.2 Test Execution Strategy

The test cases will be executed using both manual and automated testing methods.

### **Manual Testing**

* The manual testing will involve the following steps:
* Testers will execute the test cases manually on the mobile devices listed above.
* Testers will ensure that the app is functioning as expected and that all requirements are met
* Testers will document any issues or defects found during testing.

### **Automated Testing**

* Automated testing will be performed using Appium and the Java programming language.
* Test scripts will be developed using Appium and Java to automate the test cases.
* The automated tests will be executed on the devices listed in the Test Environment section.
* Automated tests will be run using a continuous integration tool, such as Jenkins, to ensure that they are executed on a regular basis.

## Test Reporting

Test results will be documented in a test report, which will include the following:

* Summary of test execution results, including number of tests executed, passed, failed, and blocked.
* Detailed description of any issues or defects found during testing, including steps to reproduce and severity.
* Suggestions for improvement and recommendations for addressing any defects or issues found during testing.
* Any additional feedback or comments related to the testing process.
* The test report will be shared with the development team on a regular basis to ensure t that they are aware of the progress of testing and any issues that need to be addressed.

# 6. Features to Be Tested

* The ability for users to log workouts, including exercises, sets, and reps.
* The ability for users to track progress over time and set goals for fitness and nutrition.
* The ability for users to connect with friends and share progress and achievements.
* The ability for users to access a library of exercises and workout plans.
* The ability for users to integrate with other fitness apps and devices.

# 7. Entry and Exit Criteria

## 7.1 Entry Criteria:

* The software requirements specification (SRS) document is available and reviewed.
* The fitness tracking app is installed on the test environment.
* The test environment is set up and configured properly.
* The necessary test data and inputs are available.
* The test team is trained on the usage of the fitness tracking app and the testing methodologies.

## 7.2 Exit Criteria:

* All the planned test cases have been executed and reported.
* The test results are analysed and all high and medium severity defects are resolved.
* The performance and usability benchmarks are met as per the defined requirements.
* The test report is generated and shared with the stakeholders.
* The fitness tracking app is ready for release.

# 8. Risk Assumptions

* Technical Issues: There may be technical issues with the mobile app, such as compatibility issues with certain devices, that could impact the testing process.
* Time Constraints: There may be time constraints that limit the amount of testing that can be performed, leading to the possibility of undiscovered bugs.
* Data Security: There may be risks associated with storing and handling user data, such as the possibility of data breaches or unauthorized access.
* Integration Issues: There may be issues with integrating the app with other fitness devices or apps, which could lead to unexpected behaviour and errors.
* User Behaviour: Users may interact with the app in unexpected ways, leading to scenarios that were not anticipated during testing.
* Performance Issues: There may be issues with the app's performance, such as slow load times or crashes, that could impact the user experience.

It's important to keep in mind that risk assumptions may vary depending on the specific project and context, so it's important to consider all potential risks and how they could impact the testing process.

# 6. Test Summary Report

At the end of the testing phase, a Test Summary Report will be created by the QA

Engineer to provide an overview of the testing process and results. The report will include the following details:

* Summary of the testing activities and their outcomes
* Metrics on the number of test cases executed, passed, failed, and blocked
* Defects identified during testing along with their severity, priority, and status
* Recommendations for improving the quality and user experience of the fitness-tracking app
* Overall assessment of the fitness tracking app's readiness for releases

The Test Summary Report will be shared with the development team, project stakeholders, and management for their review and feedback.

# 7. Defect Tracking and Management

During the testing phase, defects and bugs will be identified by the QA Engineer and logged in the defect tracking system. The development team will be responsible for resolving the defects and verifying the fixes. The status of each defect will be tracked in the defect tracking system and communicated to the project stakeholders.

# 8. Test Closure

Once the testing phase is complete, the QA Engineer will review the testing process and results to identify areas for improvement. A Test Closure Report will be created to summarize the testing activities and outcomes, and provide recommendations for future testing efforts.

The Test Closure Report will be shared with the development team and project stakeholders for their feedback and review.

# 9. Conclusion

In conclusion, this test plan outlines the approach, scope, and objectives of testing the fitness tracking mobile app. The plan covers functional testing, usability testing, and performance testing, and includes a comprehensive set of test cases to ensure the app meets the requirements and provides a positive user experience. The test plan also includes a framework for test automation, defect tracking and management, test reporting, and test closure. The successful execution of this test plan will help to identify and resolve defects and ensure the fitness tracking app is of high quality and ready for release.