TEST PLAN FOR MYFITNESS PAL

ChangeLog

Version	Change Date	By	Description
version number	Date of Change	Name of person who made changes	Description of the changes made

1 INTRODUCTION 2
1.1 SCOPE 2
1.2 WORK PLAN 2
1.3 ROLES AND RESPONSIBILITIES 3
2 TEST PLAN AND STRATEGY 3
2.1 FUNTINCAL TESTING 4
2.2 TEST CRITERIA 4
2.2.1 ENTRY CRITERIA 4
2.2.2 EXIT CRITERIA 4
2.3 TEST PROCEDURE 3
2.3.1 TESTING TYPES 3
2.3.2 TESTING LEVELS 3
2.4 BUG REPORTS 3
3 TEST DELIVERABLES 6
3.1 BEFORE TESTING PHASE 4
3.2 DURING THE TESTING 4
3.3 AFTER THE TESTING CYCLES IS OVER 4
4 RESOURCES 7
4.1 TESTING TOOLS 7
4.2 TEST ENVIRONMENT 7
5 TESTING PROCESS RISK 5
6 TESTING EFFORT ESTIMATION 6

1 Introduction

The Test Plan is designed to prescribe the scope, approach, resources, and schedule of all testing activities of the MyFitness Pal application.

The plan identify the items to be tested, the features to be tested, the types of testing to be performed, the personnel responsible for testing, the resources and schedule required to complete testing, and the risks associated with the plan.

1.1 Scope

Testing of mobile application is in the scope of this test plan. The following components and functions would be tested:

1. Create account using email.

2. Create account using Social media accounts using (Apple ID - Facebook account).

3. Log out.

4.Join a plan.

5.Add Exercise.

6. Create meals.

7. Create Recipe

8.Create Food

9. Find Recipes.

10.Use the barcode scanner.

11.Community

12. Profile

13. Calorie Counter

14. Settings

1.2 Work Plan

The parties are agreed to follow the next work plan:

1. Test plan preparation
2. Test plan approval
3. Functional testing and bugs reporting
4. Daily reports preparation
5. Final report preparation

1.3 Roles and Responsibilities

Roles	Responsibilities
Project Manager	● Managing the whole testing process. ● Providing all the needed resources for the testing activities.
QA Lead	● Collecting and learning the requirements. ● Validation of the documentation. ● Planning the testing works. ● Monitoring the testing activities, making sure that the works are performed according to the plan. ● Reporting about the progress, number and severity of the found errors.
Test Engineer	● QA process / logging found errors into the approved bug tracking system.

2 Test Plan and strategy

2.1 Functional testing

The objective of functional testing is to make sure that the whole software product works according to the requirements, and no significant errors appear in the application. Functional testing is the most substantial part of software testing. It involves checking of different aspects of the system. A software product must pass all the planned tests. Only in this case its quality can be assured.

2.2 Test Criteria

1- Entry Criteria

- The application construction is completed.
- The test engineers are dedicated.
- Availability of complete or partially testable code
- Access to sufficient and desired test data
- The readiness of test cases
- Setting up of test environment with all the necessary resources like tools and devices
- Spot checks to ensure all the preconditions are met,and eliminate any defects or tasks that are delaying the process timelines.

2- Exit Criteria

- All the planned tests are performed.
- There are no show-stopping errors.
- All the errors of high priority and severity are fixed.
- The test results are evaluated, discussed and approved.

2.3 Test Procedure

Test procedure assumes the next points: Reporting of found software bugs.

1- Testing Types.

The main testing type that would be executed:

- Smoke Testing
- Functional Testing
- Usability Testing
- Regression testing
- Retesting (during the second round if needed)
- API Testing.
- Automation Regression Testing

2- Testing Levels.

- Unit testing: validate that each unit of the software code performs as expected.
- Integration Testing (Individual software modules are combined and tested as a group)
- System Testing: Conducted on a complete, integrated system to evaluate the system's compliance with its specified requirements.

2.4 Bug Reports

Defects found during the Testing will be categorized according to the bug-reporting tool "JIRA" and the categories are:

- Critical (blocker) defects are the failure of the complete software system or of a critical subsystem, and no work or testing can be carried out after the occurrence of the defect. It also applies to data loss failures and with processes that leave inconsistent data stored on the database.

- Major defects (and crashes) are those which also causes failure of entire or part of the system, but there are some processing alternatives which allows further

operation of the system. It also applies to the system crashing, or aborting, during normal operation of a non-critical flow.

- Minor defects do not result in failure but causes the system to show incorrect, incomplete, or inconsistent results.

- Low defects are small errors that do not affect the functionality: typos, grammar mistakes, wrong terminology, etc.

3 Test Deliverables

Test deliverables are provided as below

3.1 Before testing phase

- Test plans document.
- Test cases documents
- Test Design specifications.
- Test Scripts.

3.2 During the testing

- Tools (Mobile devices) OR (Simulators/Emulators)
- Test Data
- Test Traceability Matrix
- Error logs and execution logs.
- Daily Status Report

3.3 After the testing cycles is over

- Test Results/reports
- Defect Report
- Installation/ Test procedures guidelines
- Release notes

4 Resources

4.1 Testing Tools

The following tools will be used for this project:

Name of Process	Tool
Defect Tracking	JIRA
Test Cases	JIRA
Screenshot/video recording	For screenshot : Press the Side button and volume Up button together. For Video recording : Using Screen Recording.
Automation Tools	JDK - IDE – SDK – Simulators/Emulators

4.2 Test Environment

The list of the devices:

Name of device	OS
iPhone devices (8,X,11,12)	IOS version (14.3,13.7,11.3)
Android devices (Samsung S8+, Samsung S10,Samsung S9)	Android versions (8,9,10)
Huawei nova 5i	OS 10

5 Testing Process Risks

The next issues may influence testing works:

- Changes and modifications of the software product that were not planned and discussed with the test team beforehand.
- Changes in the software requirements that were not discussed with the test team beforehand.
- Delays in correcting/fixing errors.
- Delays in delivering new builds to the test team.

6 Testing Effort Estimation

Lists out all the activities that have to be performed by the QA team and estimates how many man-hours each activity is going to take.

- Divide the whole project task into subtasks
- Allocate each task to team member
- Effort Estimation for Tasks