**Test Plan for Personal Health & Fitness Tracker**

**Table of Contents**

1. **Test Plan Identifier** **and Test Cases**......................................................... 2
2. **References** ................................................................. 2
3. **Introduction** ............................................................... 2
4. **Test Items** ................................................................. 2
5. **Software Risk Issues** .................................................. 2
6. **Features to be Tested** ............................................... 2
7. **Features not to be Tested** ......................................... 3
8. **Approach** ................................................................. 3
9. **Item Pass/Fail Criteria** ............................................ 3
10. **Suspension Criteria and Resumption Requirements** ..... 3
11. **Test Deliverables** ...................................................... 3
12. **Remaining Test Tasks** .............................................. 3
13. **Environmental Needs** .............................................. 4
14. **Staffing and Training Needs** .................................. 4
15. **Responsibilities** ....................................................... 4
16. **Schedule** ................................................................. 4
17. **Planning Risks and Contingencies** ....................... 4
18. **Approvals** .............................................................. 4
19. **Glossary** .............................................................. 5

**1. Test Plan Identifier** This document is the official test plan for the **Personal Health and Fitness Tracker** project, aimed at ensuring the quality, functionality, and reliability of the system.

**2. References**

* IEEE 829 Standard for Software Test Documentation
* Project Requirements Document
* System Design Document
* User Stories and Backlog

**3. Introduction** The **Personal Health and Fitness Tracker** is a web-based application that helps users monitor their fitness activities, log workouts, track calorie intake, and follow diet plans. This test plan outlines the strategy, scope, and approach for testing the system to ensure it meets user expectations and functional requirements.

**4. Test Items** The test will cover the following components of the system:

* **User Authentication**: Login and registration functionality.
* **Workout Logging**: Adding and tracking workout routines.
* **Diet & Calorie Tracking**: Logging meals and monitoring calorie intake.
* **Subscription & Payment**: Membership plans and payment processing.
* **UI & Navigation**: Overall user experience, responsiveness, and aesthetics.
* **Contact & Support**: Ensuring seamless user queries and support system functionality.

**5**. **Test Cases**:

TC01: Valid login

TC02: Invalid login

TC03: Add new workout, diet list, sleep analysis

TC04: Empty fields validation

TC06: Fetch correct data

TC07: Data format check

TC08: Correct tips shown

TC09: Goal matching logic

TC09: Query returns relevant answer

TC10: Handles unexpected questions gracefully

**5. Software Risk Issues**

* **Security Concerns**: Ensuring safe storage of user data.
* **Scalability Issues**: Handling multiple user requests efficiently.
* **Integration Failures**: Possible issues integrating external APIs for diet plans or workout recommendations.
* **Performance Bottlenecks**: Slow response times affecting user experience.

**6. Features to be Tested**

* **User Registration and Login** functionality.
* **Workout and meal logging** to ensure correct data input and retrieval.
* **Subscription management** for purchasing and renewing memberships.
* **System notifications** including workout reminders.
* **Database interactions** for proper data storage and retrieval.

**7. Features Not to be Tested**

* Third-party API updates that are not under our control.
* Backend database optimization beyond required functionality.
* Performance testing beyond expected user loads at this stage.
* **LLM Integration** for data analysis and responses

**8. Approach** Testing will follow a **black-box testing approach**, focusing on functional validation.

* **Unit Testing**: Developers will test individual components.
* **Integration Testing**: Ensuring different modules work together.
* **System Testing**: End-to-end testing of the complete system.
* **Acceptance Testing**: Verifying the system meets business and user requirements.

**9. Item Pass/Fail Criteria** A test case is considered **passed** if the actual result matches the expected result. If a critical issue is found, the test will be marked as **failed**, and a fix will be required before proceeding.

**10. Suspension Criteria and Resumption Requirements** Testing will be **suspended** if:

* A critical defect blocks further execution.
* An external dependency (e.g., API, database) is unavailable. Testing will **resume** once the blocking issue is resolved.

**11. Test Deliverables**

* **Test Cases** for each functionality.
* **Bug Reports** documenting all defects found.
* **Test Summary Reports** covering test results and findings.

**12. Remaining Test Tasks**

* Finalizing test cases.
* Conducting regression testing after fixing major bugs.
* User acceptance testing with a small group of beta users.

**13. Environmental Needs**

* Web hosting for live testing.
* Testing environments with different browsers and devices.
* Secure database access for testing data storage and retrieval.

**14. Staffing and Training Needs**

* **Developers** for unit testing.
* **QA Testers** for functional and system testing.
* **End Users** for acceptance testing.

**15. Responsibilities**

* **Test Manager**: Oversees overall testing efforts.
* **QA Engineers**: Execute test cases and report defects.
* **Developers**: Fix defects and conduct unit testing.

**16. Schedule**

* **Week 1-2**: Writing and reviewing test cases.
* **Week 3-4**: Execution of unit and integration tests.
* **Week 5**: System and user acceptance testing.

**17. Planning Risks and Contingencies**

* **Delays in fixing critical defects**: Allocate extra development time.
* **Limited test data**: Generate synthetic data for testing.
* **Unavailability of external APIs**: Use mock data for simulation.

**18. Approvals** This test plan will be reviewed and approved by the **Project Manager** and the **Lead Developer**.

**19. Glossary**

* **UAT**: User Acceptance Testing.
* **Regression Testing**: Retesting after bug fixes.
* **Mock Data**: Simulated data for testing purposes.

This test plan ensures the **Personal Health and Fitness Tracker** is tested effectively, ensuring a smooth user experience and reliable system performance.