

1. Introduction

1.1 Purpose

The purpose of this test plan is to verify the functionality of the Umob mobile application.

1.2 Scope

This test plan covers testing for Umob version [X.X] on both Android and iOS platforms.

1.3 Objectives

The main objectives of this test plan are to:

- Ensure seamless user experience in booking and using various transportation services (like E-moped, Bike, Taxi, Public Transport).
- Validate the accuracy and reliability of location-based features.
- Identify and report any defects/observations.

2. Test Items

2.1 Features to be Tested

The following features will be tested:

- User registration and authentication
- Booking a ride
- Real-time tracking of the vehicle
- Notifications and alerts

2.2 Features not to be Tested

The scope of this testing activity will not include:

- Any payment systems/operations currently in place.
- Any infrastructure components will not be tested for functionality as they are assumed to work as designed

3. Test Deliverables

3.1 Test Cases

A document containing detailed test cases for each feature, including positive and negative scenarios.

3.2 Test Data

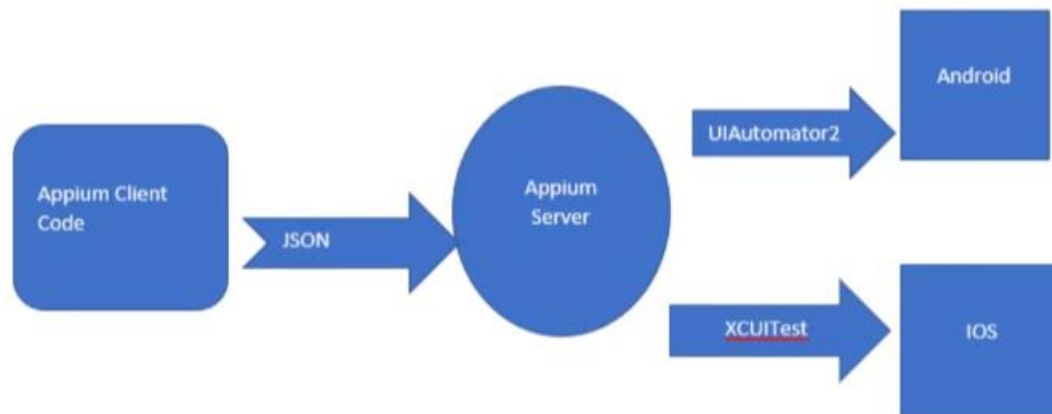
Sample data for testing, including valid and invalid inputs, mock locations.

4. Testing Strategy/ Approach

Our testing philosophy is based on risk-based testing where each test case will have a priority such as low, medium, high, and critical, and the test cases with the higher priority will be executed first.

We will be using Eclipse IDE to write our code with Appium Java library using Emulator/ Simulators.

4.1 Testing Architecture Diagram



Here:

- Client code: Automation code/Scripts preferred in Java Language
- Appium UiAutomator2 Driver is a test automation framework for Android devices
- XCUItest is a framework provided by Apple for UI testing of iOS apps

4.2 Testing Levels

- Unit Testing: Testing individual functions within the app.(Developer side)
- Integration Testing: Verifying interactions between different pages/sections.
- System Testing: Ensuring the entire system works cohesively.

4.3 Testing Methods

- Manual testing for user interface and user experience.
- Automated testing for regression testing and repetitive tasks.

5. Test Environment

5.1 Hardware

- Android and iOS devices of various models and versions.

5.2 Software

- Android and iOS operating systems.(Mobile right)
- Testing tools/ Software for automated testing like Eclipse IDE, Java, NodeJS, Appium, Android Studio, Jira.
- Automation testing Libraries - TestNG, Cucumber, Selenium, Appium Client dependencies.

6. Defect Tracking

6.1 Defect Logging

- Use Jira tool to log and track defects.

7. Risks and Assumptions

7.1 Risks

- Network issues affecting real-time tracking.
- Platform-specific challenges on Android or iOS.

7.2 Assumptions

- Users have stable internet connectivity and Bluetooth connectivity.
- The application adheres to privacy and security standards.

8. Conclusion

This test plan outlines the testing approach and strategy for testing the Umob application. Any deviations/ Suggestions/feedback for this plan will always be welcome.