**Software Testing Application**

**Sensor-Based Hand Soap Machine**

**Application Name:** Sensor-Based Hand Soap Machine

**Document:** Software Testing Scenarios and Test Cases

**Author:** Amulya Pabbu

**Table of Contents:**

1. Application Overview

2. Test Scenarios and Test Cases

2.1 Device Initialization and Setup

2.2 Soap Dispensing Mechanism

2.3 Sensor Detection

2.4 Notification System

2.5 Maintenance and Error Handling

**Application Overview:**

The Sensor-Based Hand Soap Machine is an innovative device designed to dispense soap automatically when a hand is detected under the sensor. The primary features include device initialization and setup, automatic soap dispensing, accurate sensor detection, notifications for low soap levels, and error handling for maintenance purposes.

**Features:**

1. Device Initialization and Setup
2. Soap Dispensing Mechanism
3. Sensor Detection
4. Notification System
5. Maintenance and Error Handling

**Test Scenarios and Test Cases:**

1. Device Initialization and Setup

Test Scenario 1: Successful Device Initialization

Positive Test Cases:

1.1.1: Initialize the device with a full soap container.

1.1.2: Initialize the device with a properly charged battery.

1.1.3: Verify that the device powers on and performs a self-check.

1.1.4: Verify that the device connects to the app (if applicable).

1.1.5: Verify that the device indicates readiness with a green LED light.

Negative Test Cases:

1.1.6: Initialize the device with an empty soap container.

1.1.7: Initialize the device with a low battery.

1.1.8: Verify that an error is shown if the device fails the self-check.

1.1.9: Attempt to initialize the device without completing the required setup steps.

1.1.10: Verify that the device shows a red LED light if there is an initialization error.

2. Soap Dispensing Mechanism

Test Scenario 2: Soap Dispensing Operation

Positive Test Cases:

2.2.1: Dispense soap when a hand is detected under the sensor.

2.2.2: Verify that the correct amount of soap is dispensed.

2.2.3: Dispense soap multiple times to ensure consistency.

2.2.4: Verify that the soap container can be easily refilled.

2.2.5: Verify that the device stops dispensing when the hand is removed.

Negative Test Cases:

2.2.6: Attempt to dispense soap with an empty soap container.

2.2.7: Verify that no soap is dispensed if the sensor is blocked.

2.2.8: Attempt to dispense soap with a faulty dispensing mechanism.

2.2.9: Verify that an error is shown if the soap fails to dispense.

2.2.10: Attempt to dispense soap while the device is performing maintenance.

3. Sensor Detection

Test Scenario 3: Sensor Functionality

Positive Test Cases:

3.3.1: Verify that the sensor detects a hand at the correct distance.

3.3.2: Verify that the sensor responds quickly to hand detection.

3.3.3: Verify that the sensor can detect hands of different sizes.

3.3.4: Verify that the sensor works in various lighting conditions.

3.3.5: Verify that the sensor does not falsely trigger without a hand.

Negative Test Cases:

3.3.6: Attempt to trigger the sensor with an object other than a hand.

3.3.7: Verify that the sensor does not detect a hand beyond its effective range.

3.3.8: Verify that the sensor fails to detect a hand if the sensor is dirty.

3.3.9: Attempt to trigger the sensor in complete darkness.

3.3.10: Verify that an error is shown if the sensor malfunctions.

4. Notification System

Test Scenario 4: Notification for Low Soap Level

Positive Test Cases:

4.4.1: Verify that the device sends a notification when the soap level is low.

4.4.2: Verify that the notification includes instructions for refilling the soap.

4.4.3: Verify that the notification is sent to the connected app (if applicable).

4.4.4: Verify that the device shows a yellow LED light for low soap level.

4.4.5: Verify that the notification is received promptly after the soap level drops.

Negative Test Cases:

4.4.6: Verify that no notification is sent if the soap level is adequate.

4.4.7: Verify that an error is shown if the notification system fails.

4.4.8: Attempt to disable notifications in the app settings and verify the result.

4.4.9: Verify that the notification is not duplicated for the same low soap event.

4.4.10: Verify that the notification does not contain incorrect or incomplete information.

5. Maintenance and Error Handling

Test Scenario 5: Maintenance and Error Notifications

Positive Test Cases:

5.5.1: Verify that the device indicates when maintenance is needed.

5.5.2: Verify that the device provides clear instructions for maintenance.

5.5.3: Verify that the device resumes normal operation after maintenance.

5.5.4: Verify that the device shows a red LED light for critical errors.

5.5.5: Verify that the device logs errors for troubleshooting.

Negative Test Cases:

5.5.6: Attempt to operate the device without performing necessary maintenance.

5.5.7: Verify that the device shows an error if maintenance is overdue.

5.5.8: Attempt to clear error logs without fixing the issues.

5.5.9: Verify that an error is shown if the device fails to log errors correctly.

5.5.10: Attempt to use the device with multiple active errors and verify the result.