SE 3XA3: Test Plan
Ohm: Resistor Scanner

Team 4, ohm
Jonathan Brown, brownjs2
Graeme Crawley, crawleg
Ryan Marks, marksr2

October 31, 2016

Contents

List of Tables

Table 1: **Revision History**

Date	**Version**	**Notes**
October 31st 2016	0.0	First Revision of Testing Plan

1 General Information

1.1 Purpose

This document will describe the testing procedure used to ensure the correct functionality of Group 4's 3XA3 Project, Ohm. While the implementaion of the project is not complete, it is important to have plan tests that verify that the project complies with the specifications and requirements set out in the SRS. These tests are necessary in order to produce a high quality end product, as well as track and manage the progress of the group.

1.2 Scope

The tests prescribed in the test plan should verify the efficacy of the resistor band detection, the colour selection and the resistor body detection (note: the resistor body detection has not yet been implemented).

1.3 Acronyms, Abbreviations, and Symbols

1.4 Overview of Document

The Test Plan will contain five main sections excluding this introductory one, as well as an appendix.

2 Plan

2.1 Software Description

Ohm will allow users with a desktop computer or smartphone equipped camera to determine the resistance of a standard 4-band resistor by placing it within the camera frame. The software will, using Opencv, detect and read the colour bands of the resistor automatically to determine it's resistance. This software will be implemented in Java.

2.2 Test Team

All members of Group 4, Jonathan Brown, Graeme Crawley, and Ryan Marks will be responsible for testing components of the application as well as the application as a whole.

2.3 Automated Testing Approach

2.4 Testing Tools

JUnit 4 will be the tool for unit testing of the software. JCov will be used to ensure 95%+ testing coverage.

2.5 Testing Schedule

INTSERT TABLE HERE

See Gantt Chart at the following url ...

3 System Test Description

3.1 Tests for Functional Requirements

3.1.1 User Input

Warning Box

1. F-FDM-1
   Type: Functional, Dynamic, Manual

   Initial State: App is not open

   Input: App is launched

   Output: Camera starts, warning box appears over camera view

   How test will be performed: The user will launch the app which will automatically display a warning box upon opening.

2. F-UDM-2
   Type: Unit, Dynamic, Manual

   Initial State: Warning box is displayed

   Input: ”X” on the box is pressed

   Output: Warning box closes, showing camera

   How test will be performed: The user will press the ”X” on the warning box, which will then close the warning box.

3. F-UDM-3
   Type: Unit, Dynamic, Manual

   Initial State: Warning box is displayed

   Input: Screen is pressed anywhere that isn't the ”X” on the warning box

   Output: No output

   How test will be performed: The user will press the space around ”X” on the warning box, which will do nothing.

Screen

1. F-UDM-4
   Type: Unit, Dynamic, Manual

   Initial State: App is open, ready to scan

   Input: Screen is tapped

   Output: No output

   How test will be performed: The user will tap the screen while the app is running, resulting in no change

3.1.2 Resistor Scanning

1. F-FDM-5
   Type: Functional, Dynamic, Manual

   Initial State: App is open, camera is on

   Input: Camera is pointed at resistor in incorrect orientation

   Output: No output

   How test will be performed: The user will point the camera at a resistor without lining up the red line on the app to intersect any of the four bands of the resistor.

2. F-FDM-6
   Type: Functional, Dynamic, Manual

   Initial State: App is open, camera is on

Input: Camera is pointed at resistor in correct orientation with red line crossing through first stripe from either side

Output: No output

How test will be performed: The user will point the camera at a resistor with the red line intersecting one of the four bands of the resistor.

3. F-FDM-7
   Type: Functional, Dynamic, Manual

   Initial State: App is open, camera is on

   Input: Camera is pointed at resistor in correct orientation with red line crossing through two stripes from either side

   Output: No output

   How test will be performed: The user will point the camera at a resistor with the red line intersecting two of the four bands of the resistor.

4. F-FDM-8
   Type: Functional, Dynamic, Manual

   Initial State: App is open, camera is on

   Input: Camera is pointed at resistor in correct orientation with red line crossing through three stripes from either side

   Output: No output

   How test will be performed: The user will point the camera at a resistor with the red line intersecting three of the four bands of the resistor.

5. F-FDM-9
   Type: Functional, Dynamic, Manual

   Initial State: App is open, camera is on

   Input: Camera is pointed at resistor in correct orientation with red line crossing through four stripes from either side

   Output: Resistance value is displayed on the screen

   How test will be performed: The user will point the camera at a resistor with the red line intersecting all four bands of the resistor.

6. F-FDM-10
   Type: Functional, Dynamic, Manual

   Initial State: App is open, camera is on

   Input: Camera is on with no resistor visible on screen

   Output: No output

   How test will be performed: The user will point the camera so that no resistor is visible on screen.

3.2 Tests for Nonfunctional Requirements

3.2.1 User Tests

1. NF-FDM-11
   Type: Functional, Dynamic, Automated

   Initial State: App is open, camera is on

   Input/Condition: 100 resistors scanned

   Output/Result: 95 resistors or more will be identified accurately

   How test will be performed: 100 resistors will be scanned consecutively, their values being displayed on screen. The values will be recorded and compared to he manually calculated values of each resistor. At least 95 resistors will be identified accurately.

2. NF-FDM-12
   Type: Functional, Dynamic, Manual

   Initial State: App is open, camera is on

   Input: Give the app and a resistor to a user between the ages of 8 and 70

   Output: The user will be able to identify the value of the resistor

   How test will be performed: The user will be given a resistor and the app. The app will be easy to use and allow the resistance to be measured.

3. NF-FDM-13
   Type: Functional, Dynamic, Manual

   Initial State: App is open, camera is on

Input: Give the app to a user who doesn't have internet connection

Output: The user will be able to identify the value of the resistor

How test will be performed: The user will be given a resistor and the app. The app will be easy to use and allow the resistance to be measured without the need for internet.

3.2.2 Launch Tests

1. NF-FDM-14
   Type: Functional, Dynamic, Manual

   Initial State: App is not open

   Input: App is launched

   Output: The warning box will display text

   How test will be performed: The app will be launched and display a warning box which contains text. The text will notify the user that the app should not be used in safety critical systems.

2. NF-FDM-15
   Type: Functional, Dynamic, Manual

   Initial State: App is not open

   Input: App is launched

   Output: The app will display the warning box within * seconds

   How test will be performed: The app be launched and display the warning box within * seconds

4 Tests for Proof of Concept

4.1 Band Identification

1. PC-BI-1

   Type: Functional, Dynamic, Manual

   Initial State: Software not running

   Input: Run Command

Output: Software displays static image of resistor with no bands selected.

How test will be performed: Manual execution.

2. PC-BI-2

   Type: Functional, Dynamic, Manual.

   Initial State: Software running, no axis selected

   Input: Select axis directly across resistor body.

   Output: A ring centered around each band of the appropriate colour.

   How test will be performed: Manual execution.

3. PC-BI-3

   Type: Functional, Dynamic, Manual

   Initial State: Software not running

   Input: Select axis incorrectly.

   Output: Exception is thrown

   How test will be performed: Manual execution.

4. PC-BI-4

   Type: Functional, Dynamic, Manual.

   Initial State: Software running, no axis selected

   Input: Select axis directly across resistor body.

   Output: A ring centered around each band of the appropriate colour.

   How test will be performed: Manual execution.

5 Comparison to Existing Implementation

6 Unit Testing Plan

JUnit4 will be used to perform unit testing for this application.

6.1 Unit testing of internal functions

Unit testing will be performed on any independant functions where comparison to expected results is possible. This project features several prime candidates for such testing: expected band locations, and expected colour selection. Once this unit testing is performed these individual modules can be combined and automated dyamic and functional testing can be performed to verify the final resistance output. Modules featuring image processing (the majority of them) will be tested using a stub to pass them static images in place of camera input. The application behaves quite linearly, allowing for complete test coverage.

6.2 Unit testing of output files

This project does not feature any output files.