

EZSustain Mobile Application

# Test Case Document

## **Table of Contents**

Introduction.....	2
Constraints.....	2
Test Items and environment.....	2
Approach.....	2
Test Cases.....	2-5
References.....	5

## **Introduction**

The purpose of testing this mobile application is to make sure it is working properly before presenting it to our users for use. The goal is to get all the bugs out of the program so that it will run the way it was designed to run. In addition, we are using a controlled batch of beta users in our immediate area, so that we can fact check information edits about the areas waste management program with ease.

## **Constraints**

A constraint with this software is the complexity of the designed system, if any piece fails, or doesn't work properly, it will reduce the functionality and intent of the app. The intent is to have all US states and areas in those states' information regarding waste management in our database, however, there are so many areas, and variables, it will be hard to fact check users' information updates for every area, as well as populating the database with the required info. This app is designed in mind to work on any android device, however, we've specifically only tested it in an Android 4.4 KitKat emulator and higher, assuring it works as planned, but with the diversity of android devices, firmware versions, and hardware revisions, this makes assuring the app will run on all devices Android 4.4 or later without fault a major task.

## **Test Items & Environment**

Our main approach to testing is an Android device emulator, provided by Android Studio. We've configured emulators for every major release since Android 4.4 Kit Kat to test the current build of the app. In addition, we have our android phones (Galaxy Note 8, Galaxy S5, Nexus 6, LG V10, and an Nvidia Shield Tablet), to test our app in practicality on real devices, using android studios build in debugger and logger. Everyone working on the code will upload their changes and works in progress to GitHub, for all users to see, while testers will record their findings in testing, and record when and if problems occur while testing, which will also be upload to GitHub and forwarded to the programmers as needed.

## **Approach**

The type of testing we are using for this mobile application is manual black box testing, with users in the app team's local area

## **Test Cases**

### **Test Case ID TC001 "Main Screen":**

**Summary:** Verify the application loaded correctly on the device

**Prerequisite:** User clicks on the application

**Instructions:** Start the application to make sure the app runs correctly

### **Test Data and Expected Results**

1. Start the application with the app icon, the main screen should appear, asking for first time registration of an account
2. If this isn't the first time, and the user checked don't show this again, make sure the application doesn't display the request to make an account screen.

#### **Test Case ID TC002 “Access the Camera”:**

**Summary:** Verify the application can access the camera correctly

**Prerequisite:** User is on the main screen

**Instructions:** Click the “Scan barcode by camera” button

#### **Test Data and Expected Results**

1. When the button is clicked, the system should ask upon first time running the app if they'll allow camera access
2. Allowing camera access should bring the user to the barcode scanning camera function
3. In the case that the device doesn't have a camera, the app will bring up a message box stating that the camera cannot be used on the device

#### **Test Case ID TC003 “Access the manual search function”:**

**Summary:** Verify the application goes to the manual search screen

**Prerequisite:** User is on the main screen

**Instructions:** Click the “Manual Search” button

#### **Test Data and Expected Results**

1. When the button is clicked, the app should bring up a screen with a search bar waiting for the users search item to be typed in
2. After this, a displayed list of items matching the search should be listed

#### **Test Case ID TC004 “View more info on item”:**

**Summary:** Verify the application can display data on a listed item

**Prerequisite:** User has scanned a barcode, or found the information manually of an item

**Instructions:** Click on an item in the list

#### **Test Data and Expected Results**

1. After the user scans the barcode, or enters a search manually, a list should be displayed using results from the camera, or the search
2. When the user clicks on one of these items, it should go to the items description page

#### **Test Case ID TC005 “Create an account”:**

**Summary:** Verify the application can create user accounts

**Prerequisite:** User is on the account registration page

**Instructions:** Click the menu icon in the corner of the main page, an “account registration” option should be in the menu, click on that, which will bring you to a login screen, click “First time user?” to get to the registration page for account creation

**Test Data and Expected Results**

1. The user will supply their name, email, and username to create an account
2. Afterwards, a toast message should popup either alerting the user their account creation was successful or failed

**Test Case ID TC006 “Login”:**

**Summary:** Verify the application can login a registered user

**Prerequisite:** User is on the login screen

**Instructions:** Click the menu icon in the corner of the main page, select “login”

**Test Data and Expected Results**

1. When the login button is pressed, a screen asking for the user’s username and email should appear
2. After the user enters supplied credentials, a toast message should say logged in as “username” or “Login failed, please check credentials”

**Test Case ID TC007 “Edit area’s waste management info”**

**Summary:** Verify the application can pull location info, and contact info for the location

**Prerequisite:** User is on the main screen, and logged into a valid user account

**Instructions:** Click the “Edit waste management info” button on the main screen

**Test Data and Expected Results**

1. After clicking the edit button, the user should be on a screen displaying the areas waste management information, and contact info, based on location of the user’s device
2. An edit info button should be present, clicking on it will bring the user to an screen similar to the displayed info screen, only with editable text fields
3. The user will press “submit for approval” so that an app administrator may approve the changes

**Test Case ID TC008 “Edit account info”**

**Summary:** Verify the user can edit their account info if necessary

**Prerequisite:** User is on the account info page

**Instructions:** Click the menu bar at the top of the app, “View account info” will be an option

**Test Data and Expected Results**

1. After clicking on view account info, a screen should appear with editable text fields currently populated with the user’s current information

2. A check button will submit the changes to the account database
3. If the passwords don't match, a toast message will alert the user, and the changes won't be submitted
4. If the password fields are left blank, a toast message will alert the user to input the passwords into the field

#### **Test Case ID TC009 “View areas waste management info”**

**Summary:** Verify the user can look at the waste management info for their area

**Prerequisite:** User is on the main screen

**Instructions:** Click the menu bar at the top of the app, view areas waste management info will be an option,

#### **Test Data and Expected Results**

1. After the “View waste management info” option is pressed, the application should pull the nearest area's public works info from the waste management database using the user's GPS location
2. After this data is pulled, a data screen should be displayed, with the contact info, materials taken, and recycling method of the nearest area being displayed

#### **Test Case ID TC010 “View menu options”**

**Summary:** Verify the user can access the sidebar menu

**Prerequisite:** User is on the main screen

**Instructions:** Click the menu bar at the top of the app, a side bar with specific options (Login, Register account, View waste management info, view account info, and edit account info)

#### **Test Data and Expected Results**

1. When the sidebar menu button is pressed, a sidebar with the above options should appear

#### **References**

The Requirements Analysis Document (RAD) is available on the GitHub page here.

(<https://github.com/Killasnipawoof/CIS460AppDocs/>)