Transaction Tracker

**Test Document**

Version 1.0.0

Prepared By

|  | **Student ID** | **Email Address** |
| --- | --- | --- |
| **Shemar Henry** | 180915 | Shemarhenry24@gmail.com |
| **Mark-Anthony Jones** | 180920 | Jmarkanthony.062@gmail.com |
| **Shemar Lundy** | 180916 | lundyshemar@yahoo.com |
| **Dejeon Battick** | 180915 | Dejeon.battick11@gmail.com |
|  |  |  |

Software Testing

Instructed by Thomas Canhao Xu

June 22, 2019

Table of Contents

TOC \o 2-2 \t "Heading, 3,TOC Heading, 3"

Table of Contents PAGEREF \_Toc \h 2

Test Plan PAGEREF \_Toc1 \h 7

Test Plan Identifier: PAGEREF \_Toc2 \h 7

Introduction PAGEREF \_Toc3 \h 7

Background PAGEREF \_Toc4 \h 8

Objectives of the Test Plan PAGEREF \_Toc5 \h 8

References PAGEREF \_Toc6 \h 9

Test Items PAGEREF \_Toc7 \h 9

Features to Be Tested PAGEREF \_Toc8 \h 9

Features Not to Be Tested PAGEREF \_Toc9 \h 9

Approach PAGEREF \_Toc10 \h 9

Pass/Fail Criteria PAGEREF \_Toc11 \h 10

Test Deliverables: PAGEREF \_Toc12 \h 10

Responsibilities: PAGEREF \_Toc13 \h 10

Schedule: PAGEREF \_Toc14 \h 10

Test Design Specification PAGEREF \_Toc15 \h 11

Test Case Specification PAGEREF \_Toc16 \h 11

Test Case Specification Identifier PAGEREF \_Toc17 \h 11

Test Items PAGEREF \_Toc18 \h 11

Input Specifications PAGEREF \_Toc19 \h 11

* + Data PAGEREF \_Toc20 \h 11

Output Specifications PAGEREF \_Toc21 \h 11

Environmental Needs PAGEREF \_Toc22 \h 11

Test Case Specification Identifier PAGEREF \_Toc23 \h 11

Test Items PAGEREF \_Toc24 \h 11

Input Specifications PAGEREF \_Toc25 \h 11

* + Data PAGEREF \_Toc26 \h 11

Output Specifications PAGEREF \_Toc27 \h 12

Environmental Needs PAGEREF \_Toc28 \h 12

Test Case Specification Identifier PAGEREF \_Toc29 \h 12

Test Items PAGEREF \_Toc30 \h 12

Input Specifications PAGEREF \_Toc31 \h 12

* + Data PAGEREF \_Toc32 \h 12

Output Specifications PAGEREF \_Toc33 \h 12

Environmental Needs PAGEREF \_Toc34 \h 12

Test Case Specification Identifier PAGEREF \_Toc35 \h 12

Test Items PAGEREF \_Toc36 \h 12

Input Specifications PAGEREF \_Toc37 \h 12

Output Specifications PAGEREF \_Toc38 \h 13

Environmental Needs PAGEREF \_Toc39 \h 13

Test Case Specification Identifier PAGEREF \_Toc40 \h 13

Test Items PAGEREF \_Toc41 \h 13

Input Specifications PAGEREF \_Toc42 \h 13

Output Specifications PAGEREF \_Toc43 \h 13

Environmental Needs PAGEREF \_Toc44 \h 13

Test Case Specification Identifier PAGEREF \_Toc45 \h 13

Test Items PAGEREF \_Toc46 \h 13

Input Specifications PAGEREF \_Toc47 \h 13

* + Data PAGEREF \_Toc48 \h 13

Output Specifications PAGEREF \_Toc49 \h 14

Environmental Needs PAGEREF \_Toc50 \h 14

Test Case Specification Identifier PAGEREF \_Toc51 \h 14

Test Items PAGEREF \_Toc52 \h 14

Input Specifications PAGEREF \_Toc53 \h 14

* + Data PAGEREF \_Toc54 \h 14

Output Specifications PAGEREF \_Toc55 \h 14

Environmental Needs PAGEREF \_Toc56 \h 14

Test Case Specification Identifier PAGEREF \_Toc57 \h 14

Test Items PAGEREF \_Toc58 \h 14

Input Specifications PAGEREF \_Toc59 \h 14

* + Data PAGEREF \_Toc60 \h 14

Output Specifications PAGEREF \_Toc61 \h 15

Environmental Needs PAGEREF \_Toc62 \h 15

Test Case Specification Identifier PAGEREF \_Toc63 \h 15

Test Items PAGEREF \_Toc64 \h 15

Input Specifications PAGEREF \_Toc65 \h 15

Output Specifications PAGEREF \_Toc66 \h 15

Environmental Needs PAGEREF \_Toc67 \h 15

Test Items PAGEREF \_Toc68 \h 15

Input Specifications PAGEREF \_Toc69 \h 15

Output Specifications PAGEREF \_Toc70 \h 15

Environmental Needs PAGEREF \_Toc71 \h 15

Test Case Specification Identifier PAGEREF \_Toc72 \h 15

Test Items PAGEREF \_Toc73 \h 16

Input Specifications PAGEREF \_Toc74 \h 16

* + Data PAGEREF \_Toc75 \h 16

Output Specifications PAGEREF \_Toc76 \h 16

Environmental Needs PAGEREF \_Toc77 \h 16

Test Incident Report PAGEREF \_Toc78 \h 16

Test Summary Report PAGEREF \_Toc79 \h 20

Document Version History

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Version | Date | Changes | Author | Approver |
| 1.0 | June 22, 2019 | Original Plan | Mark-Anthony Jones |  |

# Test Plan

## Test Plan Identifier:

This test plan is referred as Transition Tracker Test 1.0

## Introduction

## Background

Transaction Tracker, created by TPGD, is an application intended for the end user to track their expenses as well as create a budget and assist in making sound financial decisions. The application was developed using the Android Studio IDE with the Java programming language. The application features a Bottom Navigation Drawer with four menus, each representing Personal, Expenses, Income and Budget respectively. This application also features a SQLite database which stores user input. The end user can clear the database as they choose.

This test plan is intended to test the Transaction Tracker dynamic quality attributes, namely: reliability, correctness, completeness, consistency, usability and performance.

## Objectives of the Test Plan

The test plan encompasses the entirety of the Transaction Tracker application. What will be tested is the dynamic quality attributes of the software, focusing on specifics such as

* The probability of failure-free operation of the application
* How the application’s functionality corresponds to the documentation
* The presence of all the features laid out in the documentation
* How consistent the application performs
* Ease of use
* Performance under stress

For the testing process, four members will be recruited to test the application. A Software Requirements Specification document will be provided by the TPGD team. The Espresso framework in Android Studio will be used to extensively test the application. The testing method being used is white box testing as Espresso requires that the code be fully known to the tester to perform test cases. Expected deliverables include: a test incident report, a test log and a test summary report.

## References

The TPGD Transaction Tracker SRS Document

# Test Items

Testing will be done on the front end of the application using the Android Studio Espresso framework on Macintosh and Linux environments

## Features to Be Tested

* Daily Expenses and Transactions
* Income
* Expenses
* Chart
* Clear Expenses and Transactions

## Features Not to Be Tested

* “Spin the Wheel”
* Calculator
* Settings

# Approach

Domain testing, boundary value testing, requirements testing, robustness testing and functional testing will be employed in this test procedure.

# Pass/Fail Criteria

No more than 95% of test cases should fail and there should exist no critical bugs

# Test Deliverables:

Test Incident Report, Test Summary Report, Test Logs, Test Cases

# Responsibilities:

Shemar Henry will be responsible for testing the database and stress testing. Shemar Lundy will be responsible for domain testing and functional testing the graph as well as editing the transaction. Dejeon Battick will be testing entry fields for adding transactions to test for robustness as well as functional test if transactions can be cancelled. Mark-Anthony Jones will be testing functionality in deleting transactions, entering a budget and editing an expense. Domain testing will also be employed for all the test cases that relate to the requirements that include adding a transaction or an expense.

# Schedule:

|  |  |  |  |
| --- | --- | --- | --- |
| Date | Task | Duration | Expected Deliverables |
| May 13, 2019 | Elicit Requirements for the Transaction Tracker | 3 Days | Requirements Specification Document |
| May 17, 2019 | Design Test Cases and and Specify Methodology | 3 Days | Test Design Specification  Test Cases |
| May 20, 2019 | Rigorous Testing of the Application Using The Test Cases | 3 Days | Test Log  Test Incident Report |
| May 23, 2019 | Create Test Summary and Report | 1 Day | Test Summary |

# Test Design Specification

Specify testing techniques, methods of analyzing results, common attributes of test cases (input constraints)

# Test Case Specification

# Test Case Specification Identifier

AT\_1

# Test Items

Add Transaction using Daily Activity

# Input Specifications

# Data

* + - An integer for the transaction amount field where x=1040 , x=null and x=1039
    - A string of lengths 255, 256, and 0 for the transaction title
    - Date values of null and a single entry of “dd/mm/yyyy”
  + **Human actions**
    - Press EditText, DatePicker and Button

# Output Specifications

A RecyclerView Item containing the title of the transaction

# Environmental Needs

Hardware - Huawei Android Device

Software - Android Studio Espresso

# Test Case Specification Identifier

DT\_1

# Test Items

Delete Transaction using Daily Activity

# Input Specifications

# Data

* + - An integer for the transaction amount field where x=1040 , x=null and x=1039
    - A string of lengths 255, 256, and 0 for the transaction title
    - Date values of null and a single entry of “dd/mm/yyyy”
  + **Human actions**
    - Press EditText, DatePicker and Button

# Output Specifications

A RecyclerView Item containing the title of the transaction

# Environmental Needs

Hardware - Huawei Android Device

Software - Android Studio Espresso

# Test Case Specification Identifier

ET\_1

# Test Items

Edit Transaction using Daily Activity

# Input Specifications

# Data

* + - An integer for the transaction amount field where x=1040 , x=null and x=1039
    - A string of lengths 255, 256, and 0 for the transaction title
    - Date values of null and a single entry of “dd/mm/yyyy”
  + **Human actions**
    - Press EditText, DatePicker and Button

# Output Specifications

A RecyclerView Item containing the title of the transaction

# Environmental Needs

Hardware - Huawei Android Device

Software - Android Studio Espresso

# Test Case Specification Identifier

DT\_2

# Test Items

Delete All Transaction using Menu Widget

# Input Specifications

* + **Human actions**
    - Press Menu Icon, Dialog and Button

# Output Specifications

Empty Daily Activity

# Environmental Needs

Hardware - Huawei Android Device

Software - Android Studio Espresso

# Test Case Specification Identifier

CT\_1

# Test Items

Cancel Transaction using Daily Activity

# Input Specifications

* + **Human actions**
    - Press Button

# Output Specifications

The user should see the Daily Activity

# Environmental Needs

Hardware - Huawei Android Device

Software - Android Studio Espresso

# Test Case Specification Identifier

EB\_1

# Test Items

Enter Budget using Expenses Activity

# Input Specifications

# Data

* + - An integer for the total income field and amount spend field where x=1040 , x=null and x=1039
    - A string of lengths 255, 256, and 0 for the expense
  + **Human actions**
    - Press EditText and Button

# Output Specifications

A RecyclerView Item containing the title of an expense and the amount

# Environmental Needs

Hardware - Huawei Android Device

Software - Android Studio Espresso

# Test Case Specification Identifier

AE\_1

# Test Items

Add Expense using Expenses Activity

# Input Specifications

# Data

* + - An integer for the amount spend field where x=1040 , x=null and x=1039
    - A string of lengths 255, 256, and 0 for the expense
  + **Human actions**
    - Press EditText and Button

# Output Specifications

A RecyclerView Item containing the title of an expense and the amount

# Environmental Needs

Hardware - Huawei Android Device

Software - Android Studio Espresso

# Test Case Specification Identifier

EE\_1

# Test Items

Edit Expense using Expenses Activity

# Input Specifications

# Data

* + - An integer for the amount spend field where x=1040 , x=null and x=1039
    - A string of lengths 255, 256, and 0 for the expense
  + **Human actions**
    - Press EditText and Button

# Output Specifications

A RecyclerView Item containing the title of an expense and the amount

# Environmental Needs

Hardware - Huawei Android Device

Software - Android Studio Espresso

# Test Case Specification Identifier

CE\_1

# Test Items

Cancel Expense using Expenses Activity

# Input Specifications

* + **Human actions**
    - Press EditText and Button

# Output Specifications

The user will see the Expenses Activity

# Environmental Needs

Hardware - Huawei Android Device

Software - Android Studio Espresso

# Test Items

Delete Expense using Expenses Activity

# Input Specifications

* + **Human actions**
    - Press EditText and Button

# Output Specifications

The user will see the Expenses Activity

# Environmental Needs

Hardware - Huawei Android Device

Software - Android Studio Espresso

# Test Case Specification Identifier

GG\_1

# Test Items

Generate Graph using Chart Activity and Expenses Activity

# Input Specifications

# Data

* + - An integer for the amount spend field where x=1040 , x=null and x=1039
    - A string of lengths 255, 256, and 0 for the expense
  + **Human actions**
    - Press EditText and Button and Tap Screen

# Output Specifications

A Graph Widget with the expenses graphed

# Environmental Needs

Hardware - Huawei Android Device

Software - Android Studio Espresso

# Test Incident Report

Summary:

Application crashes and force closes upon later launches. Test cases AT\_1, GG\_1, AE\_1, ET\_1, EE\_1 and EB\_1 all gave this error when x=1040

Description:

* Inputs
  + - * + Integer and Real: x=1040
* Expected Results
  + - * + The tester expected the application to show an error that the field is empty
    - Actual Results
      * + The application crashed
    - Test Case(s)
      * + AT\_1, GG\_1, AE\_1, ET\_1, EE\_1 and EB\_1
* Tested by: Mark-Anthony Jones, Shemar Henry, Dejeon Battick, Shemar Lundy
  + - Severity: Major
    - Priority: Immediate

Summary:

Application crashes. Test cases AT\_1, GG\_1, AE\_1, ET\_1, EE\_1 and EB\_1 all gave this error when x=null

Description:

* Inputs
  + - * + Integer and Real: x=null
* Expected Results
  + - * + The tester expected the application to show an error that the field is empty
    - Actual Results
      * + The application crashed
    - Test Case(s)
      * + AT\_1, GG\_1, AE\_1, ET\_1, EE\_1 and EB\_1
* Tested by: Mark-Anthony Jones, Shemar Henry, Dejeon Battick, Shemar Lundy
  + - Severity: Major
    - Priority: Immediate

Summary:

Application saves transaction when Transaction Date is null. Test cases AT\_1 and ET\_1 and all gave this error.

Description:

* Inputs
  + - * + Date: x=null
* Expected Results
  + - * + The tester expected the application to show that the input field is empty
    - Actual Results
      * + The application accepted the empty field
    - Test Case(s)
      * + AT\_1, ET\_1
    - Tested by: Dejeon Battick and Shemar Lundy
    - Severity: Minor
    - Priority: Deferred

Summary:

Application displays an illegible graph. Test cases GG\_1 gave this error.

Description:

* Inputs
  + - * + Integer and Real: x=1039, x=0, x=10
* Expected Results
  + - * + The tester expected the application to show a clear graph with the input values displayed
    - Actual Results
      * + The application display an illegible graph
    - Test Case(s)
      * + GG\_1
* Tested by: Shemar Lundy
  + - Severity: Major
    - Priority: Immediate

Summary:

Application shows a negative value in the pie chart when the user enters an expense that is greater than the income. Test cases EB\_1 and AE\_1 and all gave this error.

Description:

* Inputs
  + - * + Integer and Real: x where x=Expense & x>y where Y=Total Income
* Expected Results
  + - * + The tester expected the application to show an error that the expense is greater than the income
    - Actual Results
      * + The application accepted the higher expense and generate a pie chart with a negative quadrant
    - Test Case(s)
      * + EB\_1, AE\_1
    - Tested by: Dejeon Battick and Shemar Lundy
    - Severity: Major
    - Priority: Delayed

# Test Summary Report

Summary

The Transaction Tracker app was tested by the Strange Energy team consisting of Shemar Henry, Shemar Lundy, Mark-Anthony Jones and Dejeon Battick. A few requirements were not tested, including the calculating cost of item and spin the wheel. All the others were rigorously tested using domain testing methods, boundary value analysis, robustness testing, functional testing and requirements testing. All deliverables were successfully created and the project was thoroughly tested.

Comprehensiveness assessment

Strings beyond 256 were not tested as the testers were unaware of the limit of the length of string allowed in Android. Aside from this, the application was tested as thorough as conceivable by the testers. Performance testing was omitted as the requirements stated that the application is meant for a user to input transactions at a slow rate. Other methods not stated in this document were omitted simply because they were not able to test the application.

Summary of results and Evaluation

Robustness was a huge problem in this application as empty fields were allowed to go through without any error checking. Limits were absent on the domain so EditText fields were able to take in exponentially high values that overloaded the application resulting in crashes or unappealing user interface appearances

Acknowledgements