Michael du Toit

2193161

Budgiecoin

SD6501 - Testing Documentation

Table of Contents

[Table of Figures 2](#_Toc74251142)

[Unit Testing 2](#_Toc74251143)

[Unit Test Cases 2](#_Toc74251144)

[Unit Tests Screenshots 7](#_Toc74251145)

[Espresso Testing 8](#_Toc74251146)

[Espresso Test Cases 8](#_Toc74251147)

[Espresso Tests Screenshots 11](#_Toc74251148)

[User Acceptance Testing 15](#_Toc74251149)

[Survey Form and Questions: 15](#_Toc74251150)

[User Acceptance Testing Survey Results: 16](#_Toc74251151)

# Table of Figures

[Figure 1 - Unit Testing Results for all the AccountUnitTests Test Cases 7](#_Toc74251152)

[Figure 2 - Unit Test Results for TransactionUnitTests Test Cases 7](#_Toc74251153)

[Figure 3 - Unit Test Results for UserUnitTests Test Cases 8](#_Toc74251154)

[Figure 4 - Espresso Testing Results for AccountBalancesActivityTest 11](#_Toc74251155)

[Figure 5 - Espresso Testing Results for AddAccountActivityTest 11](#_Toc74251156)

[Figure 6 - Espresso Testing Results for LoginActivityTest 12](#_Toc74251157)

[Figure 7 - Espresso Testing Results for MainActivityTest 12](#_Toc74251158)

[Figure 8 - Espresso Testing Results for RegistrationActivityTest 12](#_Toc74251159)

[Figure 9 - Espresso Testing Results for SettingsActivityTest 13](#_Toc74251160)

[Figure 10 - Espresso Testing Results for ViewAccountActivityTests 13](#_Toc74251161)

[Figure 11 - Espresso Testing Results for ViewTransactionsActivityTests 13](#_Toc74251162)

[Figure 12 - Espresso Testing Results for TransactionActivityTests 14](#_Toc74251163)

[Figure 13 – User Acceptance Testing survey was collected via a Google Form 15](#_Toc74251164)

# Unit Testing

## Unit Test Cases

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **Test ID** | **Test Class** | **Test Case / Test Method** | **Test Description** | **Expected Result** | **Actual Result** | **Status** | **Image** |
| UT01 | Account UnitTests | account\_defaultConstructor | Create an instance of the Account object using the default constructor and check that the fields are set as expected. | Account.id = 0 Account.name = "Test Acccount" Account.balance = 0.0 | Account.id = 0 Account.name = "Test Acccount" Account.balance = 0.0 | Pass | 1 |
| UT02 | Account UnitTests | account\_nameConstructor | Create an instance of the Account object passing the string "Test Account" to the constructor, and check that the fields are set as expected. | Account.id = 0 Account.name = "Test Acccount" Account.balance = 0.0 | Account.id = 0 Account.name = "Test Acccount" Account.balance = 0.0 | Pass | 1 |
| UT03 | Account UnitTests | account\_allParamsConstructor | Create an instance of the Account object passing the values: int 1, string "Test Account", long 10.00 to the constructor, and check that the fields are set as expected. | Account.id = 1 Account.name = "Test Acccount" Account.balance = 10.00 | Account.id = 1 Account.name = "Test Acccount" Account.balance = 10.00 | Pass | 1 |
| UT04 | Account UnitTests | account\_testAllGetterSetter Methods | Create an instance of the Account object using the default constructor, then use all the getter and setter methods of the class to set and get all the fields and check that they are as expected. | Account.id = 0 Account.name = "Test Acccount" Account.balance = 0.0 | Account.id = 0 Account.name = "Test Acccount" Account.balance = 0.0 | Pass | 1 |
| UT05 | Transaction UnitTests | transaction\_defaultConstructor | Create an instance of the Transaction object using the default constructor and check that the fields are set as expected. | Transaction.id = 0 Transaction.name = null Transaction.value = 0.0 Transaction.account = 0 Transaction.note = null Transaction.datetime = Calendar instance | Transaction.id = 0 Transaction.name = null Transaction.value = 0.0 Transaction.account = 0 Transaction.note = null Transaction.datetime = Calendar instance | Pass | 2 |
| UT06 | Transaction UnitTests | transaction\_allParamsExceptId Constructor | Create an instance of the Transaction object using the constructor that takes values for all the fields except ID and check that the fields are set as expected. | Transaction.id = 0 Transaction.name = "Test Transaction UT" Transaction.value = 10.0 Transaction.account = 1 Transaction.note = "Unit Testing" Transaction.datetime = Calendar instance | Transaction.id = 0 Transaction.name = "Test Transaction UT" Transaction.value = 10.0 Transaction.account = 1 Transaction.note = "Unit Testing" Transaction.datetime = Calendar instance | Pass | 2 |
| UT07 | Transaction UnitTests | transaction\_allParamsConstructor | Create an instance of the Transaction object using the constructor that takes values for all the fields and check that the fields are set as expected. | Transaction.id = 1 Transaction.name = "Test Transaction UT" Transaction.value = 10.0 Transaction.account = 1 Transaction.note = "Unit Testing" Transaction.datetime = Calendar instance | Transaction.id = 1 Transaction.name = "Test Transaction UT" Transaction.value = 10.0 Transaction.account = 1 Transaction.note = "Unit Testing" Transaction.datetime = Calendar instance | Pass | 2 |
| UT08 | Transaction UnitTests | transaction\_testAllGetterSetter Methods | Create an instance of the Transaction object using the default constructor, then use all the object's set methods to set the fields and use the get methods to validate that the fields are set as expected. | Transaction.id = 1 Transaction.name = "Test Transaction UT" Transaction.value = 10.0 Transaction.account = 1 Transaction.note = "Unit Testing" Transaction.datetime = Calendar instance | Transaction.id = 1 Transaction.name = "Test Transaction UT" Transaction.value = 10.0 Transaction.account = 1 Transaction.note = "Unit Testing" Transaction.datetime = Calendar instance |  |  |
| UT09 | UserUnitTests | user\_testDefaultConstructor | Create an instance of the User object using its default constructor and check that the fields are set as expected. | User.id = 0 User.username = null User.pinNumber = null | User.id = 0 User.username = null User.pinNumber = null | Pass | 3 |
| UT10 | UserUnitTests | user\_testExcludingIdConstrutor | Create an instance of the User object using the constructor that takes two strings (username and pin number), pass test values to it and check that the fields are set as expected. | User.id = 0 User.username = "Test User" User.pinNumber = "1234" | User.id = 0 User.username = "Test User" User.pinNumber = "1234" | Pass | 3 |
| UT11 | UserUnitTests | user\_testAllParamsConstructor | Create an instance of the User object using the constructor that takes three values (id, username, pin number) and pass test values to it. Check that the fields are set as expected. | User.id = 1 User.username = "Test User" User.pinNumber = "1234" | User.id = 1 User.username = "Test User" User.pinNumber = "1234" | Pass | 3 |
| UT12 | UserUnitTests | user\_testAllGetterSetterMethods | Create an instance of the User object using the default constructor, then use all of the object's set methods to set the fields and use the get methods to validate that the fields are set as expected. | User.id = 2 User.username = "Test User" User.pinNumber = "1234" | User.id = 2 User.username = "Test User" User.pinNumber = "1234" | Pass | 3 |

## Unit Tests Screenshots

A screenshot of a computer

Description automatically generated with medium confidence

Figure 1 - Unit Testing Results for all the AccountUnitTests Test Cases

A screenshot of a computer

Description automatically generated with medium confidence

Figure 2 - Unit Test Results for TransactionUnitTests Test Cases

A screenshot of a computer

Description automatically generated with medium confidence

Figure 3 - Unit Test Results for UserUnitTests Test Cases

# Espresso Testing

## Espresso Test Cases

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Test ID** | **Test Class** | **Test Case / Test Method** | **Test Description** | **Status** | **Figure** |
| ES01 | AccountBalancesActivityTest | checkAllViewElementsPresent | Checks that all the UI view elements that should be visible are displayed. | Pass | 4 |
| ES02 | AddAccountActivityTest | checkAllViewElementsPresent | Checks that all the UI view elements that should be visible are displayed. | Pass | 5 |
| ES03 | AddAccountActivityTest | testAccountCreationPresent | Enters a data for creating an account, clicks the “Create” button and then checks that the account has been created. | Pass | 5 |
| ES04 | AddAccountActivityTest | testAccountCancellationPresent | Finds the Cancel button on the Create Account view, then clicks cancel and checks that the app returns to the Main Activity. | Pass | 5 |
| ES05 | LoginActivityTest | checkAllViewElementsPresent | Checks that all the UI elements that should be visible are displayed. | Pass | 6 |
| ES06 | LoginActivityTest | testLoginInvalidLogin | Enters invalid login credentials into the input fields, clicks the “Login with Pin” button and checks that the login error message is displayed. | Pass | 6 |
| ES07 | LoginActivityTest | testLoginValidLogin | Enters valid login credentials into the input fields, clicks the “Login with Pin” button and checks that the app proceeds to the Main Activity. | Pass | 6 |
| ES08 | LoginActivityTest | testEmptyUsernameLogin | Enter a valid pin into the pin input field but leave the username field empty. Check that error message is displayed on Login Button click. | Pass | 6 |
| ES09 | LoginActivityTest | testRegisterButton | Clicks the Register button and checks that it goes to the Register Activity. | Pass | 6 |
| ES10 | MainActivityTest | checkAllViewElementsPresent | Checks that all the UI view elements that should be visible are displayed. | Pass | 7 |
| ES11 | MainActivityTest | testAddTransactionButton | Checks that when the “Add Transaction” button is clicked that it navigates to the Transaction Activity | Pass | 7 |
| ES12 | MainActivityTest | testViewTransactionsButton | Checks that when the “View Transactions” button is clicked that it navigates to the View Transactions Activity | Pass | 7 |
| ES13 | MainActivityTest | testViewAccountBalances | Checks that when the “View Account Balances” button is clicked that it navigates to the Account Balances Activity | Pass | 7 |
| ES14 | RegistrationActivityView | checkAllViewElementsPresent | Checks that all the UI view elements that should be visible are displayed. | Pass | 8 |
| ES15 | RegistrationActivityView | testCancelButton | Checks that the “Cancel” button is visible and when clicked on goes to the Login Activity. | Pass | 8 |
| ES16 | RegistrationActivityView | testRegisterButton | Enters details into the edit text fields to create a user, clicks the “Register” button and checks that it goes to the Main Activity. | Pass | 8 |
| ES17 | ViewAccountActivityTest | testAllViewElementsPresent | Checks that all the UI view elements that should be visible are displayed. | Pass | 10 |
| ES18 | ViewTransactionsActivityTest | testAllViewElementsPresent | Checks that all the UI view elements that should be visible are displayed. | Pass | 11 |
| ES19 | ViewTransactionsActivityTest | testFloatingActionButton | Check that the Floating Action Button is visible, then click on it and check that it takes the user to the Add Transaction View. | Pass | 11 |
| ES20 | AccountBalancesActivityTest | testFloatingActionButton | Check that the Floating Action Button is visible, then click on it and check that it takes the user to the AddAccount view. | Pass | 4 |
| ES21 | SettingsActivityTest | testAllViewElementsDisplayed | Check that the Username and Pin elements are shown to the user. | Pass | 9 |
| ES22 | TransactionActivityTest | checkViewElementsIncomeDisplayed | Check, on the Income Tab, that the view elements are displayed. | Pass | 12 |
| ES23 | TransactionActivityTest | checkViewElementsExpenseDisplayed | Check, on the Expense Tab, that the view elements are displayed. | Pass | 12 |
| ES24 | TransactionActivityTest | createIncomeTransaction | Go to the Income Tab, enter data into all the input fields, then click the Save button. Check that it navigates to the View Transactions Activity. | Pass | 12 |
| ES25 | TransactionActivityTest | createExpenseTransaction | Go to the Expense Tab, enter data into all the input fields, then click the Save button. Check that it navigates to the View Transactions Activity. | Pass | 12 |

## Espresso Tests Screenshots

A screenshot of a computer

Description automatically generated

Figure - Espresso Testing Results for AccountBalancesActivityTest

Text

Description automatically generated

Figure - Espresso Testing Results for AddAccountActivityTest

Text

Description automatically generated

Figure - Espresso Testing Results for LoginActivityTest

Text

Description automatically generated

Figure - Espresso Testing Results for MainActivityTest

Text

Description automatically generated

Figure - Espresso Testing Results for RegistrationActivityTest

Text

Description automatically generated

Figure - Espresso Testing Results for SettingsActivityTest

Text

Description automatically generated

Figure - Espresso Testing Results for ViewAccountActivityTests

A screenshot of a computer

Description automatically generated

Figure - Espresso Testing Results for ViewTransactionsActivityTests

Text

Description automatically generated

Figure - Espresso Testing Results for TransactionActivityTests

# User Acceptance Testing

## Survey Form and Questions:

|  |  |
| --- | --- |
| Graphical user interface, application, Teams  Description automatically generated  Figure 13 – User Acceptance Testing survey was collected via a Google Form | Graphical user interface, application, Teams  Description automatically generated |

## User Acceptance Testing Survey Results:

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **User** | **Could you register for a new user account easily?** | **Could you update your user login details?** | **Was all the text readable to you?** | **Could you create an Account?** | **Could you create a Transaction?** | **Was navigating the application intuitive? Did it not take too much thinking on what steps to do to accomplish your goals?** | **Did the application do what you expected?** |
| Melissa | Yes | Yes | Yes | Yes | Yes | Yes browsing was intuitive and the application easy to use to accomplish goals like changing user details and performing transactions. | Yes |
| Philip | Yes | Yes | Yes | Yes | Yes | Yes, easy to use | Yes |
| Joseph | Yes | Yes | Yes | Yes | Yes | It was easy and quick. | Yes |