

Test Report Withdraw & CREATEACCT



|  |  |
| --- | --- |
| Name | Student Number |
| Muhammad Usman Majeed | 10086980 |
| Jessica Nahulan | 10029341 |
| Johan Cornelissen | 10098176 |

Contents

[WITHDRAW Transaction 3](#_Toc499188689)

[Test method of choice: Decision Coverage Testing 3](#_Toc499188690)

[Source: 3](#_Toc499188691)

[Table of test cases and their inputs: 4](#_Toc499188692)

[Test Case Analysis: 5](#_Toc499188693)

[test\_1 5](#_Toc499188694)

[test\_2 5](#_Toc499188695)

[test\_3 6](#_Toc499188696)

[test\_4 7](#_Toc499188697)

[Test Report: 8](#_Toc499188698)

[CREATEACCT Transaction 10](#_Toc499188699)

[Test method of choice: White Box Input Partitioning Testing 10](#_Toc499188700)

[Source: 10](#_Toc499188701)

[Input Partitions: 10](#_Toc499188702)

[Table of test cases and their inputs: 11](#_Toc499188703)

[Test Case Analysis: 12](#_Toc499188704)

[test\_1 (Covering partition P1) 12](#_Toc499188705)

[test\_2 (Covering partition P2) 13](#_Toc499188706)

[test\_3 (Covering partition P3) 14](#_Toc499188707)

[test\_4 (Covering partition P4) 14](#_Toc499188708)

[test\_5 (Covering partition P5) 15](#_Toc499188709)

[test\_6 (Covering partition P6) 16](#_Toc499188710)

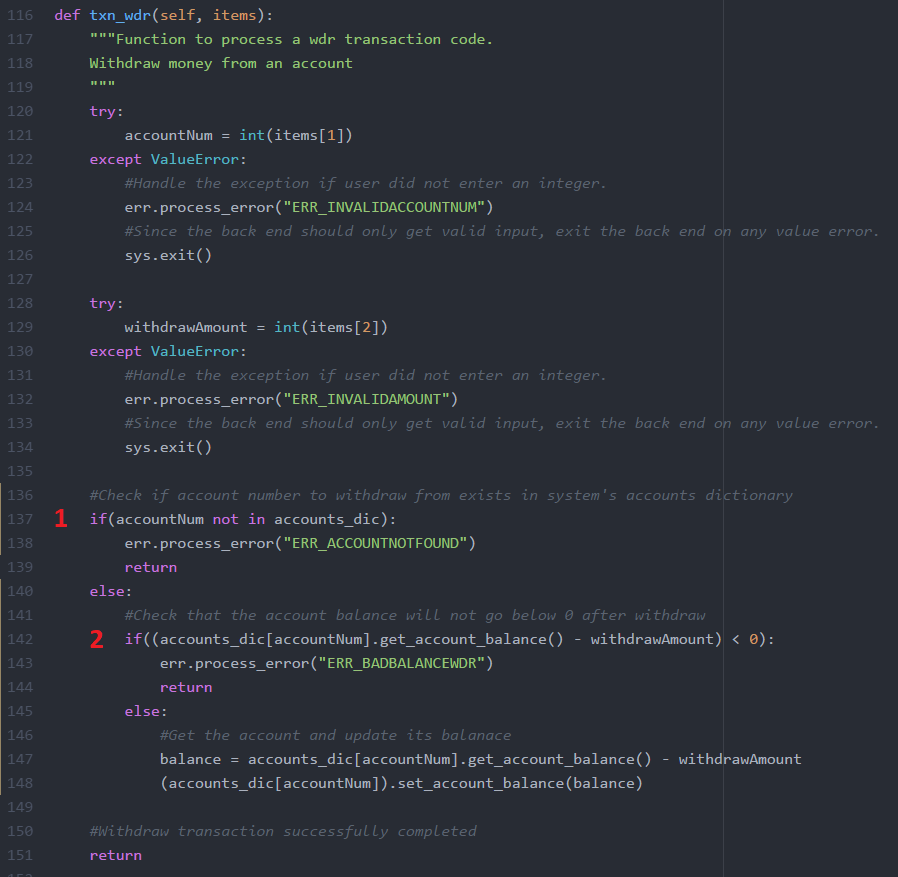
[Test Report: 17](#_Toc499188711)

[Team Contribution: 21](#_Toc499188712)

# WITHDRAW Transaction

# Test method of choice: Decision Coverage Testing

# Source:



# Table of test cases and their inputs:

The following provides a table identifying the test cases necessary to exercise decision coverage on the withdraw transaction process for the back office. The decisions below relate to the source image portrayed above and the identified (if) decisions.

In addition to identifying the various test cases, the table provides the inputs used to exercise each of the various test cases and a summary of the intended behavior.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Test | Decision | Input 1 | Input 2 | Test Case Summary |
| test\_1 | 1 : True | accounts\_dic {1234567,7777777}  *accounts\_dic is populated by input master accounts file* | accountNum:  1111111  *accountNum is provided in transaction summary input* | accountNum to withdraw from is not a valid account provided by master accounts file.  Resulting in decision 1 being “True” |
| test\_2 | 1 : False | accounts\_dic: {1234567,7777777}  *accounts\_dic is populated by input master accounts file* | accountNum:  1234567  *accountNum is provided in transaction summary input* | accountNum to withdraw from is a valid account found in the master accounts file.  Resulting in decision 1 being “False” |
| test\_3 | 2 : True | accounts\_dic[1234567] .getAccountBalance() :  1234  *Balance for account number 1234567 provided by master accounts file* | withdrawAmount:  1235  *Withdraw amount provided in transaction summary input* | withdrawAmount for account number: 1234567 is “1235” which is more than existing balance “1234” resulting in negative balance.  Resulting in decision 2 being “True” |
| test\_4 | 2 : False | accounts\_dic[1234567] .getAccountBalance() :  1234  *balance for account number 1234567 provided by master accounts file* | withdrawAmount:  1000  *Withdraw amount provided in transaction summary input* | withdrawAmount for account number: 1234567 is “1000” which is less than existing balance “1234” resulting in valid withdraw amount.  Resulting in decision 2 being “False” |

# Test Case Analysis:

## test\_1

Relevant transaction summary file line:

**WDR 1111111 1000 0000000 \*\*\***

Master accounts file contents:

**1234567 1234 JESSICA**

**7777777 3210 USMAN**

Referring to the source image decision [1] , this test case intends to trigger this decision as **true** that the provided account num to be withdrawn from is not a valid account provided by the master accounts file.

The expected output is that the withdraw transaction will not take place as invalid data has been provided and the output master accounts file will still depict the original balance for the relevant account number (Balance of 1234 in input master accounts file remains 1234 in output master accounts file for account number: 1234567).

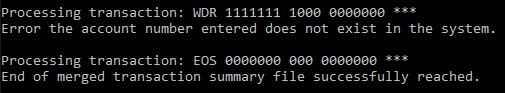
**Expected** **New Master Accounts File**:

The new master accounts file can be found in Testing/WITHDRAW/Expected Output Files/output\_maf\_test\_1.txt

**Expected** **New Valid Accounts File**:

The new valid accounts file can be found in Testing/WITHDRAW/Expected Output Files/output\_maf\_test\_1.txt

**Expected Std output**:



## test\_2

Relevant transaction summary file line:

**WDR 1234567 1000 0000000 \*\*\***

Master accounts file contents:

**1234567 1234 JESSICA**

**7777777 3210 USMAN**

Referring to the source image decision [1] , this test case intends to trigger this decision as **false** that the provided account num to be withdrawn from is a valid account provided by the master accounts file.

The expected output is that the withdraw transaction will take place as the account number provided is valid and the output master accounts file will depict the updated balance for the relevant account number (Balance of 1234 in input master accounts file becomes 234 in output master accounts file for account number: 1234567 as 1000 was successfully withdrawn from the account).

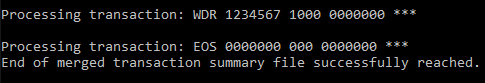
**Expected** **New Master Accounts File**:

The new master accounts file can be found in Testing/WITHDRAW/Expected Output Files/output\_maf\_test\_2.txt

**Expected** **New Valid Accounts File**:

The new valid accounts file can be found in Testing/WITHDRAW/Expected Output Files/output\_maf\_test\_2.txt

**Expected Stdoutput:**



## test\_3

Relevant transaction summary file line:

**WDR 1234567 1235 0000000 \*\*\***

Master accounts file contents:

**1234567 1234 JESSICA**

**7777777 3210 USMAN**

Referring to the source image decision [2] , this test case intends to trigger this decision as **true** that the amount to be withdrawn is more than the existing balance in the relevant account (that if the withdraw were to take place, a negative balance would result).

The expected output is that the withdraw transaction will not take place as the amount to be withdrawn “1235”, specified by the transaction summary input, is more than the existing balance for the account, balance specified by the master accounts file. The expected output is that due to the potential negative balance, the withdraw transaction will not take place and the output master accounts file will still depict the original balance amount for account number 1234567 as was in the original input master accounts file.

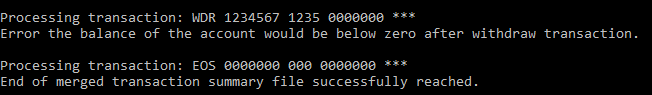
**Expected** **New Master Accounts File**:

The new master accounts file can be found in Testing/WITHDRAW/Expected Output Files/output\_maf\_test\_3.txt

**Expected** **New Valid Accounts File**:

The new valid accounts file can be found in Testing/WITHDRAW/Expected Output Files/output\_maf\_test\_3.txt

**Expected Std output:**



## test\_4

Relevant transaction summary file line:

**WDR 1234567 1000 0000000 \*\*\***

Master accounts file contents:

**1234567 1234 JESSICA**

**7777777 3210 USMAN**

Referring to the source image decision [2] , this test case intends to trigger this decision as **false** that the amount to be withdrawn is less than the existing balance in the relevant account.

The expected output is that the withdraw transaction will successfully take place as the amount to be withdrawn “1000”, specified by the transaction summary input, is less than the existing balance for the account, balance specified by the master accounts file. The expected output is that the withdraw transaction will successfully take place for account number 1234567 and the new master accounts fill depict a new balance of 1234 – 1000 = 234.

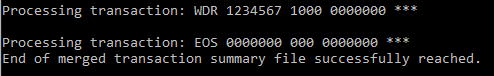
**Expected** **New Master Accounts File**:

The new master accounts file can be found in Testing/WITHDRAW/Expected Output Files/output\_maf\_test\_4.txt

**Expected** **New Valid Accounts File**:

The new valid accounts file can be found in Testing/WITHDRAW/Expected Output Files/output\_maf\_test\_4.txt

**Expected Std output:**



# Test Report:

#####################TESTING STATISTICS#####################

Tests Passed: 4

Tests Failed: 0

Tests Run: 4

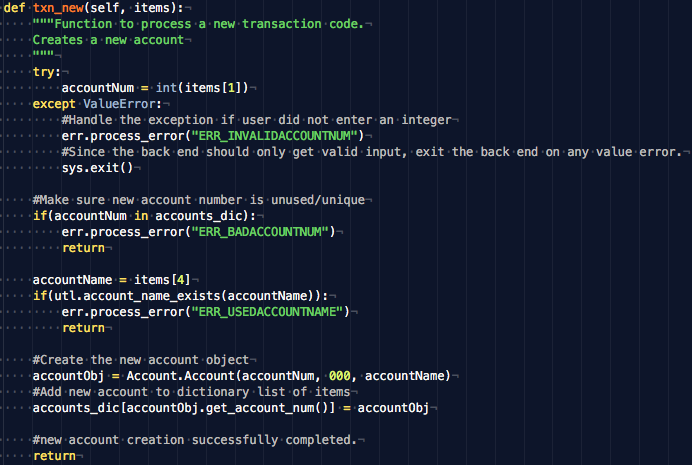
###########################################################

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Test# | Test Area | Test Name | Result | Acceptance Criteria | Error in output, Error in code, and fix applied | Re-Test details |
| 1 | Withdraw | test\_1 | Pass | When input transaction file for withdraw transaction includes an account number not present in the master accounts file, the transaction does not take place and no change is made to the account’s balance in the output master accounts file.  Std output should indicate error message for account number not found. | N/A | N/A |
| 2 | Withdraw | test\_2 | Pass | When input transaction file for withdraw transaction includes an account number present in the master accounts file, the transaction should successfully take place and the new updated balance should be present in the output master accounts file. | N/A | N/A |
| 3 | Withdraw | test\_3 | Fail | When input transaction file indicates an amount to be withdrawn that is larger than the balance present in the account, the transaction should not take place. No change should be made to the balance in the output master accounts file.  Std output should indicate error message for a bad/negative balance. | **Error in output**:  The output master accounts file reflected a changed balance amount when the transaction should not have occurred, and no change should have been made.  Error message for std output was not observed but should have been  **Error in code:**  **Line number: 142**  The withdraw amount was not being subtracted from the original balance, so the amount was always greater than 0 and the decision was always true.  **Fix applied:**  The withdraw amount was subtracted from the original balance before checking if this new amount is less than 0. | Performed manual unit test on several WITHDRAW transactions to ensure an being withdrawn that is greater than the existing balance results in an error and an incomplete/unsuccessful transaction.  Re-ran tests to ensure test failure was no longer observed. |
| 4 | Withdraw | test\_4 | Pass | When input transaction file indicates an amount to be withdrawn that is less than the balance present in the account, the transaction should successfully take place. The output master accounts file should reflect an updated balance with the withdraw amount subtracted from the original balance. | N/A | N/A |

# CREATEACCT Transaction

# Test method of choice: White Box Input Partitioning Testing

# Source:



# Input Partitions:

The method that processes the createacct transaction takes in a list of space separated items that make up the transaction command. Where one createacct command is in the following format:

“NEW <accountToBeCreated> 000 0000000 <accountName>”

**\*Note**, an assumption was made that BOTH the account number and account name need to remain unique when creating a new account. This constraint was not explicitly stated in the assignment documentation, and therefore a decision was made by the team to keep it as a constraint.

The input can be partitioned as follows:

|  |  |  |
| --- | --- | --- |
| Partition | Account Number to be created | Account Name to be created |
| P1 | Used Account number | Valid Account Name |
| P2 | Used Account number | Invalid Account Name |
| P3 | Valid Account Number | Invalid Account Name |
| P4 | Valid Account Number | Valid Account Name |
| P5 | Non-integer Account Number | Invalid Account Name |
| P6 | Non-integer Account Number | Valid Account Name |

# Table of test cases and their inputs:

The following provides a table identifying the test cases necessary to exercise white box input partitioning testing on the create account transaction process for the back office. The decisions below relate to the source image portrayed above and the identified input partitions seen above.

In addition to identifying the various test cases, the table provides the inputs used to exercise each of the various test cases and a summary of the intended behavior. For each input partition a single input combination was chosen to cover that input partition case, therefore the 6 input partitions identified above relate directly to the 6 test cases seen below.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Test | Partition | Input 1 | Input 2 | Test Case Summary |
| test\_1 | P1 | *accountNum(equivalent to items[1]):*  *7777777* | *accountName(equivalent to items[4]):*  *BUSH* | The account number provided already exists in the master accounts file being passed into the back office. Therefore, the createacct transaction should not occur even if the account name is valid. |
| test\_2 | P2 | *accountNum(equivalent to items[1]):*  *7777777* | *accountName(equivalent to items[4]):*  *OBAMA* | The account number provided already exists in the master accounts file being passed into the back office. Additionally, the account name being provided is also already used by a previous account in the master accounts file. Therefore, the createacct transaction should not occur. |
| test\_3 | P3 | *accountNum(equivalent to items[1]):*  *1234567* | *accountName(equivalent to items[4]):*  *OBAMA* | The account name provided already exists in the master accounts file being passed into the back office. Therefore, the createacct transaction should not occur even if the account number is valid. |
| test\_4 | P4 | *accountNum(equivalent to items[1]):*  *1234567* | *accountName(equivalent to items[4]):*  *BUSH* | The account number and the account name provided are not yet in use by previous accounts. Since both inputs are valid integer and string inputs respectively, the createacct transaction should proceed. |
| test\_5 | P5 | *accountNum(equivalent to items[1]):*  *777HJ77* | *accountName(equivalent to items[4]):*  *OBAMA* | The account name provided is not a valid 7-digit integer indicating that the constraints for a createacct transaction are not followed. Additionally the account name provided is already in use by an existing account in the master accounts file. Therefore, the createacct transaction should not proceed. |
| test\_6 | P6 | *accountNum(equivalent to items[1]):*  *777HJ77* | *accountName(equivalent to items[4]):*  *BUSH* | The account name provided is not a valid 7-digit integer indicating that the constraints for a createacct transaction are not followed. Therefore, the createacct transaction should not occur even if the account name is valid. |

# Test Case Analysis:

## test\_1 (Covering partition P1)

Relevant transaction summary file line:

**NEW 7777777 000 0000000 BUSH**

Master accounts file contents:

**7777777 3210 OBAMA**

Referring to the input partition P1, this test case intends to check if an existing account number with a new unique account name can be used to create a new account. Even though the account name is unique, the already used account number should cause the createacct transaction to be denied.

The expected output is that the createacct transaction will not take place and the output master accounts file will still depict the original list of accounts (no new account added to the list). Similarly, the valid accounts file created should also only depict the accounts created before the createacct transaction was trigged.

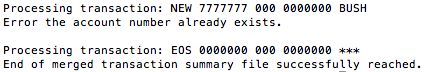
**Expected** **New Master Accounts File**:

The new master accounts file can be found in Testing/CREATEACCT/Expected Output Files/output\_maf\_test\_1.txt

**Expected** **New Valid Accounts File**:

The new valid accounts file can be found in Testing/CREATEACCT/Expected Output Files/output\_maf\_test\_1.txt

**Expected** **Std output**:



## test\_2 (Covering partition P2)

Relevant transaction summary file line:

**NEW 7777777 000 0000000 OBAMA**

Master accounts file contents:

**7777777 3210 OBAMA**

Referring to the input partition P2, this test case intends to check if an existing account number with an existing account name can be used to create a new account. In this case, both the account number and the account name are already in use so a createacct transaction should be denied. Either of these conditions would cause the transaction to not proceed in a defect free back office system.

The expected output is that the createacct transaction will not take place and the output master accounts file will still depict the original list of accounts (no new account added to the list). Similarly, the valid accounts file created should also only depict the accounts created before the createacct transaction was trigged.

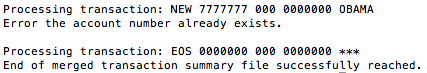
**Expected** **New Master Accounts File**:

The new master accounts file can be found in Testing/CREATEACCT/Expected Output Files/output\_maf\_test\_2.txt

**Expected** **New Valid Accounts File**:

The new valid accounts file can be found in Testing/CREATEACCT/Expected Output Files/output\_maf\_test\_2.txt

**Expected** **Std output**:



## test\_3 (Covering partition P3)

Relevant transaction summary file line:

**NEW 1234567 000 0000000 OBAMA**

Master accounts file contents:

**7777777 3210 OBAMA**

Referring to the input partition P3, this test case intends to check if a new (not already existing) account number with an existing account name can be used to create a new account. In this case, even though the account number cannot be found in the Master Accounts File (it is unique), the account name already exists (is in the Master Accounts File) and therefore a createacct transaction should be denied.

The expected output is that the createacct transaction will not take place and the output master accounts file will still depict the original list of accounts (no new account added to the list). Similarly, the valid accounts file created should also only depict the accounts created before the createacct transaction was trigged.

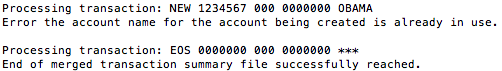
**Expected** **New Master Accounts File**:

The new master accounts file can be found in Testing/CREATEACCT/Expected Output Files/output\_maf\_test\_3.txt

**Expected** **New Valid Accounts File**:

The new valid accounts file can be found in Testing/CREATEACCT/Expected Output Files/output\_maf\_test\_3.txt

**Expected** **Std output**:



## test\_4 (Covering partition P4)

Relevant transaction summary file line:

**NEW 1234567 000 0000000 BUSH**

Master accounts file contents:

**7777777 3210 OBAMA**

Referring to the input partition P4, this test case intends to check if a new (not already existing) account number with a new unused account name can be used to create a new account. In this case, both the account number and the account name will not yet exist in the Master Accounts File and are unique. Therefore, it is expected that the createacct transaction will be allowed to complete.

The expected output is that the createacct transaction will take place and the output master accounts file will depict the original list of accounts in addition to the newly created account. Similarly, the valid accounts file created should also depict the previously created accounts as well as the account number of the account created through this test.

**Expected** **New Master Accounts File**:

The new master accounts file can be found in Testing/CREATEACCT/Expected Output Files/output\_maf\_test\_4.txt

**Expected** **New Valid Accounts File**:

The new valid accounts file can be found in Testing/CREATEACCT/Expected Output Files/output\_maf\_test\_4.txt

**Expected** **Std output**:

createacctScreenshots/test4stdout_createacct.png

## test\_5 (Covering partition P5)

Relevant transaction summary file line:

**NEW 777HJ77****000 0000000 OBAMA**

Master accounts file contents:

**7777777 3210 OBAMA**

Referring to the input partition P5, this test case intends to check if an invalid non-integer account number with an existing account name can be used to create a new account. Either of these conditions would cause the transaction to not proceed in a defect free back office system. A non-integer input should not be accepted as the account number must be a 7-digit number. The account name is in the correct format however is already in use by an existing account and therefore should not be accepted. Therefore, since both these conditions are invalid, it is expected that the createacct transaction will be denied.

The expected output is that the createacct transaction will not take place and the output master accounts file will still depict the original list of accounts (no new account added to the list). Similarly, the valid accounts file created should also only depict the accounts created before the createacct transaction was trigged.

**Expected** **New Master Accounts File**:

The new master accounts file can be found in Testing/CREATEACCT/Expected Output Files/output\_maf\_test\_5.txt

**Expected** **New Valid Accounts File**:

The new valid accounts file can be found in Testing/CREATEACCT/Expected Output Files/output\_maf\_test\_5.txt

**Expected** **Std output**:

createacctScreenshots/test5stdout_createacct.png

## test\_6 (Covering partition P6)

Relevant transaction summary file line:

**NEW 777HJ77****000 0000000 BUSH**

Master accounts file contents:

**7777777 3210 OBAMA**

Referring to the input partition P6, this test case intends to check if an invalid non-integer account number with a new unused account name can be used to create a new account. A non-integer input should not be accepted as the account number must be a 7-digit number. Even though the account name provided is a valid unused account name, it is expected that the createacct transaction will be denied due to the invalid account number provided.

The expected output is that the createacct transaction will not take place and the output master accounts file will still depict the original list of accounts (no new account added to the list). Similarly, the valid accounts file created should also only depict the accounts created before the createacct transaction was trigged.

**Expected** **New Master Accounts File**:

The new master accounts file can be found in Testing/CREATEACCT/Expected Output Files/output\_maf\_test\_6.txt

**Expected** **New Valid Accounts File**:

The new valid accounts file can be found in Testing/CREATEACCT/Expected Output Files/output\_maf\_test\_6.txt

**Expected** **Std output**:

createacctScreenshots/test6stdout_createacct.png

# Test Report:

#####################TESTING STATISTICS#####################

Tests Passed: 6

Tests Failed: 0

Tests Run: 6

###########################################################

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Test# | Test Area | Test Name | Result | Acceptance Criteria | Error in output, Error in code, and fix applied | Re-Test details |
| 1 | Createacct | test\_1 | Pass | When input transaction file for createacct transaction includes an account number already present in the master accounts file, and an **unused** (new) account name, the transaction does not take place. No change is made to the master accounts file or the valid accounts file. Std output should indicate error message for account number already exists. | N/A | N/A |
| 2 | Createacct | test\_2 | Pass | When input transaction file for createacct transaction includes an account number already present in the master accounts file, and an already used account name, the transaction does not take place. No change is made to the master accounts file or the valid accounts file. Std output should indicate error message for account number already exists OR error message for account name already exists. | N/A | N/A |
| 3 | Createacct | test\_3 | Fail | When input transaction file for createacct transaction includes an account number **not** already present in the master accounts file, and an already used account name, the transaction does not take place. No change is made to the master accounts file or the valid accounts file. Std output should indicate error message for account name already exists. | **Error in output**:  The outputted master accounts file and valid accounts file reflected a new account had been created even though no account creation should have been permitted.  Error message in std output was not observed but should have been.  **Error in code:**  The code only checked for already used account numbers (that already exist in master accounts file) but not for already used account names. Not having a check for the account name, allowed any valid string account name to be used.  **Fix applied:**  A check was added in the create account transaction method (txn\_new) to call a new method called utl.account\_name\_exists. This call would return true if the account name provided already exists and cause an error message to be presented on STDOUT as well as cancel the createacct transaction. | Performed manual unit test on several createacct transactions to ensure an existing account name (that exists in the master accounts file) causes and error message to be printed and for the transaction to be cancelled. Ensured that now new account was created in the new master accounts file and the new valid accounts file.  Re-ran tests to ensure test failure was no longer observed. |
| 4 | Createacct | test\_4 | Pass | When input transaction file for createacct transaction includes an account number **not** already present in the master accounts file, and a **not** already present account name, the transaction should take place. The new account created should be added to the master accounts file or the valid accounts file. Std output should not indicate any error messages related to the createacct transaction. | N/A | N/A |
| 5 | Createacct | test\_5 | Pass | When input transaction file for createacct transaction includes a **non-integer** account number not already present in the master accounts file, and an already present account name, the transaction should not take place. No change is made to the master accounts file or the valid accounts file. Std output should indicate error message for invalid account number format. | N/A | N/A |
| 6 | Createacct | test\_6 | Pass | When input transaction file for createacct transaction includes a **non-integer** account number not already present in the master accounts file, and a new **not** already present account name, the transaction should not take place. No change is made to the master accounts file or the valid accounts file. Std output should indicate error message for invalid account number format. | N/A | N/A |

# Team Contribution:

|  |  |  |
| --- | --- | --- |
| Name | Hours Spent | Tasks |
| Muhammad Usman Majeed | 6 hours | * Listed source for WITHDRAW transaction testing. * Defined decision partitions for WITHDRAW transaction testing. * Defined test cases for WITHDRAW transaction testing. * Corrected any failures encountered during WITHDRAW tests. |
| Jessica Nahulan | 6 hours | * Participated in meeting to decide on testing methods and splitting up tasks. * Analyzed test cases for WITHDRAW transaction testing. * Ran and documented WITHDRAW tests. * Created overall documentation and reporting skeleton |
| Johan Cornelissen | 6 hours | * Participated in meeting to decide on testing methods and splitting up tasks. * Listed source for CREATEACCT transaction testing. * Defined input partitions for CREATEACCT transaction testing. * Defined and analyzed test cases for CREATEACCT transaction testing. * Ran CREATEACCT tests and corrected any failures encountered. |