List Of Test Cases

Group Members: Aaron James, Ayaan Mustafa, Arad Ayntabli

Test Cases

Test Name	Purpose
Login Test Cases	
Login as admin	This is to test if the admin login works
Login as standard user	This is to test if the standard user can login
Try logging in after logging in	This is to test if you can’t login twice without logging out
Test admin functions with standard account	This is to test if the standard user is unable to use functions they shouldn’t be able to access
Admin user can perform all functions	The admin user should be able to perform all functions, if they are unable to perform something, there’s an error
Try using any function other than logging in, before logging in	The user should not be able to use any of the other transactions before they log in.
Try logging without the optional name element	This is to test if the name for the account is actually necessary.
Log out Test Cases	
Write transactions log file on logout	Test to see if the log file is created successfully in the correct format
Stop accepting transactions after logout	The front end should stop accepting transactions
After logout allow for user to login again	Test to see if the front end allows for the user to log back in after logging out

After logout return front end UI to default (pre-login state)	Check if the front end returns to default state after user logs out
Logout when not logged in	Tests to see if the front end allows the logout when there is no user logged in
Withdrawal Test Cases	
Enter account holder’s name as admin user	This is to test if the admin user is asked for and can input the account holders name during a withdrawal
Enter the account number	This is to test if the withdrawal function can correctly take their account number as an input
Enter the amount to withdraw	This is to test if the function can correctly take the amount as an input
The user fails to enter their account number or amount to withdraw.	This is to test if the function can handle cases where users forget to input everything
The user enters an account number that is invalid for the account they are logged into	This is to make sure that the user can only withdraw from their account
User attempts to withdraw between $1-$500	This is to test if the user is able to withdraw up to the maximum
User attempts to withdraw $501	This is to test if the function doesn’t allow withdrawals more than $500
User tries to withdraw more than they have in their account	This is to test if the function won’t allow their account balance to be less than $0
Transfer Test Cases	
Ask for account holder’s name	Ask for account holder’s name only if they are logged in as the admin
Admin fail to provide account holder’s name	If the name provided is invalid, ask the admin to try entering the name x more times. After x failures exit the transfer menu
Ask for account number the money will be transferred from	Ask for the account number of the account that money will be transferred from
Fail to provide account number of the account money will be transferred from	If the account number provided does not belong to an account of the account holder

	currently logged in, ask the user to try entering the name x more times. After x failures exit the transfer menu
Ask for account number the money will be transferred to	Ask for the account number of the account that money will be transferred to
Fail to provide account number of the account money will be transferred to	If the account number provided is not listed in the bank system, ask the user to try entering the account number x more times. After x failures exit the transfer menu
Check amount to transfer	Test to see if the user entered a valid amount of money to transfer. Value is valid if ● positive ● greater than 0 ● up to two decimal points ● If the user is in standard mode ○ less than or equal to $1000.00 ● If the user is in admin mode ○ less than or equal to funds in account
Save transaction information in log file	Check that transaction log file is created successfully and in the correct format
Paybill Test Cases	
Enter valid account holder’s name as admin user	This is to test if the admin user is asked for and can input the valid account holders name during a paybill transaction
Enter invalid account holder’s name as admin user	This is to make sure the paybill function can only take valid account holder’s name, and can handle invalid inputs without crashing.
Enter the valid account number	This is to test if the paybill function can correctly take their account number as an input
Enter an invalid account number, or an account number not associated with the account they are logged into	This is to make sure the paybill function can only take valid account numbers for the account currently logged in, and it can handle invalid inputs without crashing.
Enter a valid company name	This is to test if the paybill function can correctly take the company name as an input
The user enters a company name that is not	This is to make sure the paybill function only

““The Bright Light Electric Company (EC)”, “Credit Card Company Q (CQ)” or “Fast Internet, Inc. (FI)”	takes valid inputs for company name, if they input an invalid name the program shouldn’t crash
Enter the amount to pay	The user is able to enter an amount to pay their bill
User fails to provide any of the inputs for this transaction by leaving the input blank	The user shouldn’t be able to proceed with the transaction with any empty inputs
User attempts to pay between $0-$2000	The user is able to pay their bills up to the maximum amount per transaction
User attempts to pay $2001	The user should not be able to pay more than the maximum
User attempts to pay with more money than they have in their account	The paybill transaction should not proceed if the users account balance will be less than $0 after the bill is paid.
Deposit Test Cases	
Ask for the account holder’s name	ask for the account holder’s name only if logged in as the admin
Ask For the account number	check if the account number entered is for an account that belongs to the currently logged in user
If the user enters an invalid account number	If the user enters an account number that does match the account number of any accounts they have, ask the user to try entering the account number in x more times. After x failures exit the transaction
Ask for amount to deposit	If the user enters an invalid amount, ask them to enter a valid amount. An amount is valid if ● positive ● up to 2 decimal points
Save transaction information in log file	Ensure that the transactions log files are created successfully in the correct format
Ensure that deposited funds can not be used this session	Prevent the user from making payments, transferring or withdrawing the funds dpayuring the current session.
Create Test Cases	

A non-admin user tries to use this transaction	This tests to see if the permissions of this transaction are correct
The account number entered is already in use and is not unique	This tests if the transaction can properly handle an invalid input.
The entered account name is more than 20 characters	This tests if the transaction can properly handle an invalid input
The initial balance is more than $99999.99	This tests if the transaction can properly handle an invalid input
An admin user tries to use this transaction	This tests if the permissions of this transaction are correct
The entered account number is a unique number	This tests the results of a valid input
The entered account name is at most 20 characters	This tests the results of a valid input
The account balance is between $0 and $99999.99	This tests the results of a valid input
The account being created tries to use another transaction while in this session	This tests to see if the new account is able to use other transactions while it is being created, it should not be able to.
The admin account fails to provide either of the inputs while creating the account	The transaction shouldn't be able to proceed.
Delete Test Cases	
Admin user attempts to use delete transaction	Test to see if admin users can access the delete transaction
Non-admin user attempts to use delete transaction	Test to see if non-admin users are denied access to delete transaction
Name entered matches the name of an account holder	Test to see if delete transaction can recognize correct names
Name entered does not match name of an account holder	Test to see if delete transaction stops if incorrect name entered
Account number entered matches account held by that user	Test to see if delete transaction can recognize correct account numbers
Account number entered does not match account held by that user	Test to see if delete transaction stops if incorrect account number entered

After account is deleted no more transactions should be accepted	Test to see if the account can still take part in transactions after deleting
Save transaction in log file	Test to see if the transaction log file is successfully created in the correct format
Disable Test Cases	
Admin user	Test to see if admin users can access disable transaction
Non-admin user	Test to see if non-admin users cannot access disable transaction
Correct account name	Test to see if the disable transaction can recognize an account name in the system
Incorrect account name	Test to see if the disable transaction exits if account name does not match any account names in system
Correct account number	Test to see if the disable transaction recognize accounts that belong to a user
Incorrect account number	Test to see if the disable transaction exits if the account number entered does not match any account held by the user
Disable bank account	Test to see if account is successfully marked as disabled
Prevent transaction involving disabled account	Test to see if transactions can be conducted involving the disabled account.
Save transaction in log file	Test to see if the transaction log file is created successfully in the correct format
Changeplan Test Cases	
A non-admin user tries to use this transaction	This tests to see if the permissions of this transaction are correct
The entered name is not of an existing account holder	This tests if the transaction can properly handle an invalid input.
The entered account number is not of the account holder specified	This tests if the transaction can properly handle an invalid input.

An admin user tries to use this transaction	This tests to see if the permissions of this transaction are correct
The entered name is the name of an existing account holder	This tests the results of a valid input
The entered account number is the number of the specified account holder	This tests the results of a valid input
The user fails to provide either of the inputs	The transaction shouldn’t be able to proceed.
Bank Account Transaction File	
Test file exits in correct directory	Checks to see if the bank account transaction file exists with the correct name in the correct directory
Test that each line of text in current bank accounts file is of the correct format	Test to see if the follows the correct conventions ● exactly 40 characters plus newline ○ first 2 are transaction code ■ 01 - withdrawal ■ 02 - transfer ■ 03 - paybill ■ 04 - deposit ■ 05 - create ■ 06 - delete ■ 07 - disable ■ 08 - changplane ■ 00 - end of session ○ next 20 are the account holder’s name ○ next 5 is account number ○ next 8 amount of funds involved in the transaction in canadian dollars ○ Last 2 are miscellaneous info in the transaction ● each set of characters is separated by a space ● numeric fields are right justified with zeros ● alphabets are left justified ● unused numeric fields are filled with zeros ● unused alphabet fields are filled with spaces ● sequence of transactions ends with end of session transaction code (00)

Current Bank Accounts File Test Cases	
Test file exits in correct directory	Checks to see if the current bank accounts file exists with the correct name in the correct directory
Test that each line of text in current bank accounts file is of the correct format	Test to see if the current bank accounts files follows the correct conventions ● exactly 37 characters plus newline ○ first 5 are integers for bank account number ○ next 20 are the account holder’s name ○ next 1 is account status ○ last 8 are the current account balance in canadian dollars ● each set of characters is separated by a space ● numeric fields are right justified with zeros ● alphabets are left justified ● unused numeric fields are filled with zeros ● unused alphabet fields are filled with spaces ● file ends with special bank account with holders name as “END_OF_FILE”

Issues In Requirements

Withdrawal Function

- There is no minimum amount set for the withdrawal, this means the user can withdraw $0 or even a negative amount

Transfer

- No maximum listed for admin accounts

Paybill

- No minimum amount set for paying bills, this means the user can pay $0 or pay a negative amount

Create

- There is no minimum amount set for the initial balance of the account, so the admin could start the account with a negative balance.