

Transaction	Test Name	Intension
Login	login_01	Test if program asks for kind of session (such as standard and admin)
	login_02	If user selected standard session, program should require account holders name
	login_03	Test if user inputs gives invalid input (more than 20 characters, integers, special characters) for account holder name
	login_04	Tests to see if other transactions are being accepted before login (such as deposit, withdrawal etc)
	login_05	Test to see if after kind of session and name are inputted, program displays bank account file
	login_06	Test if standard user requests for privileged transaction and declines user of such action
	login_07	Test if admin allowed both privileged and unprivileged transactions

Test case code	Description
withdrawl_8	Test successful withdrawal with valid account holder name, account number, and sufficient balance.
withdrawl_9	Test withdrawal exceeding the \$500 session limit in standard mode.
withdrawl_10	Test withdrawal where the account balance would go below \$0.
withdrawl_11	Test withdrawal with invalid account details (e.g., incorrect account number).

withdrawl_12	Test withdrawal as an admin by providing account holder name, account number, and amount.
--------------	---

Transaction	Test Name	Intension
Deposit	deposit_13	If user is admin, program should ask for account holder's name, account number and amount to deposit
	deposit_14	Test if invalid input for account holder name (more than 20 characters, spaces, integers, special characters)
	deposit_15	Test if invalid input for account number (more than 5 numbers, letters, spaces, special characters)
	deposit_16	Test if invalid input given for deposit (negative or zero amount)
	deposit_17	Test if deposited funds are not available for use in the same session

Test case code	Description
bank_create_18	Check that the account being used to create an account is an admin account.
bank_create_19	Create an account with the name provided by the user
bank_create_19a	Check if the name is longer than 20 characters
bank_create_19b	If longer than 20 characters display an error which asks for a new name with less than 20 characters.
bank_create_19c	Generate a unique account number
bank_create_19d	check it against all already existing account numbers

bank_create_19e	If a number matches an existing account number, generate a new one and eliminate the previous number.
bank_create_20	Deposit amount of money the user wants to add to the account
bank_create_20a	Return error if amount is greater then \$99999.99
bank_create_21	Save account and info to account transaction file & Disable all function for the account until logout

Test case code	Description
bank_changePlan_22	Check that the account being used to change the plan of a different account is an admin account
bank_changePlan_23	Ask for account name
bank_changePlan_23a	Check if name is already used
bank_changePlan_24	Ask for account number
bank_changePlan_24a	Check if account number matches the account name
bank_changePlan_25	Change account from student to non-student
bank_changePlan_25a	Save info to account transaction file

Test case code	Description
bank_delete_26	Check that the account being used to delete a different account is an admin account
bank_delete_27	Ask for account name
bank_delete_27a	Check if the account exists
bank_delete_28	Ask for account number
bank_delete_28a	Check if account number matches the account name
bank_delete_29	Ask for confirmation from user before deleting account
bank_delete_30	Delete account specified by user
bank_delete_31	Save info to account transaction file

Test case code	Description
bank_disable_32	Check that the account being used to disable a different account is an admin account
bank_disable_33	Ask for account name
bank_disable_33a	Check if the account exists
bank_disable_34	Ask for account number
bank_disable_34a	Check if account number matches the account name
bank_disable_35	Change specified account from active (A) to disabled (D)
bank_disable_36	Save info to account transaction file

Test case code	Description
logout_37	Test that the logout transaction ends the session correctly.
logout_38	Test that no transaction other than login is accepted after logout.
logout_39	Test that logout is accepted only when logged in.

Test case code	Description
Transfer_40	Check if account begin used is a admin or standard
Transfer_40a	If admin ask for account holder's name

Transfer_41	Ask for account number that money is being transferred from
Transfer_41a	Return error if account number is not a valid account for account holder
Transfer_42	Ask for account number that money is being transferred to
Transfer_42a	Return error if account number is not a valid account for account holder
Transfer_43	Ask for amount the person wishes to transfer
Transfer_43a	If in standard mode return error if more the \$1000 is being transferred
Transfer_43b	Check if the account the money is being transferred from has at least that much and return an error if it does not
transfer_44	Save changes to bank account transaction file

Test case code	Description
payBill_45	Check if account being used is a admin or standard
payBill_45a	If admin ask for account holder's name
payBill_46	Ask for account number of the account being used to make payment
payBill_46a	Return error if account number is not a valid account for account holder
payBill_47	ask for the company to whom the bill is being paid
payBill_47a	Check if the company being paid is: The Bright Light Electric Company (EC)", "Credit Card Company Q (CQ)" or "Fast Internet, Inc. (FI)
payBill_48	Ask for amount to pay
payBill_48a	If standard mode return error if amount to be

	paid is greater than \$2000
payBill_48b	Check if the account being used has at least the amount needed to make the payment. Return error if it does not

### **Test Organization**

Tests are organized into a single directory structured by transaction type. Each test case is named starting with the transaction name followed by a sequential number (e.g., login\_01, withdrawal\_10, deposit\_15).

Each test case consists of the following files:

Test Input Files (.inp): These contain a sequence of commands simulating user interaction.

Expected Bank Account Transaction File Outputs (.etf): These hold the expected transaction details and final account balances.

Expected Terminal Output Files (.oup): These provide the expected output messages shown to the user.

### **Test Run Plan**

Tests will be executed using a batch script that automates running the banking system's front end with each test input file. The script follows these steps:

Executes the system with each .inp test file.

Captures the system-generated transaction file output.

Compares the transaction output with the expected .etf file.

Checks if the terminal output matches the expected .oup file.

Reports any discrepancies.

Example Test Execution Workflow:

```
for each test_case in test_directory:
    run_banking_system(test_case.inp) > output.etf
    compare output.etf expected.etf
    compare terminal_output.oup expected.oup
```

### **Requirements Problems**

In the requirements we found that it did not outline any limitations for the account name other than the fact that it had to be 20 characters or less. Assume numbers & special characters and

not non - English characters not allowed

The requirements mention that the change plan function changes the plan from student to non-student. It did not mention if the reverse is also required. Assume that the change plan function can do the reverse.

The requirements specify functionality to disable accounts but not to re-enable them. Assume that the disable function can do both.

The requirements mention a \$500 session withdrawal limit for standard users, but it is unclear if this applies across multiple transactions within a session or per individual withdrawal. Assumed that the \$500 limit is per session.

The requirements indicate that an overdraft fee notification is sent via email, but the exact overdraft limit or penalty is unspecified. Assumed that no overdraft is allowed, and any attempt will result in a declined transaction with a notification sent.

It is unclear if an admin can override withdrawal limits or withdraw funds from any account. Assumed that admins have no transaction limits but cannot withdraw from other users' accounts without authorization. Logout

The logout requirements state that the session should be terminated, but it is not specified if an auto-logout feature exists after a period of inactivity. Assumed that the user must manually execute the logout transaction.

The requirements do not specify whether a user attempting a transaction after logout should receive a generic error message or a specific logout-related error. Assumed that any attempt to transact post-logout results in a clear error message prompting re-login.