

Failure Report

Scarlet FlashView

Tongxu Ge (20054696)

Sunny Yang (20075550)

Jinting Zhang (20059360)

Jinghong Xiao (20035787)

Nov 3rd, 2019

Our group chose the **approach2** testing method to test our source code in “*frontend/*”. We modified the “**test_main_approach2.py**” template to conduct the testing. We first changed the import module name to our test file, like changed “*import qa327.app as app*” to “*import frontend.Frontend as app*”. Then we changed the **sys.argv** which is the system arguments to provide with our test file name, like changing “*app.py*” to “*Frontend.py*”.

We create one template for each operation named as “*test_create_account.py*”, “*test_delete_account.py*”, “*test_deposit.py*”, “*test_login.py*”, “*test_logout.py*”, “*test_transfer.py*”, “*test_withdraw.py*” under the folder “*test_frontend/*”. For each of the 7 operations, we created test functions like “*def test-RIT1(capsys)*”, “*def test-RIT2(capsys)*” ... with a helper function in each test case. We then changed the **terminal_input** in the helper function to be the *real input* for each test case. Also, we changed the **expected_tail_of_terminal_output** to compare with the real terminal output especially the tail of each transaction. For the **input_valid_accounts**, we input the existing valid accounts, normally [‘1234567’] and [‘7654321’] for the test cases. For the **expected_output_transactions**, we input the real output transaction summary after ‘logout’ uses it to match the “transaction summary file.txt” outputted by our program for each test case.

Based on the feedback of assignment 1, apart from following the test table in Assignment 1, we add some extra test cases to make the test more complete.

***To run the test, type the command “**pytest**”. Picture taken for both successful pass with the new source code on Mac and Windows:

```
[lxpg0605deMacBook-Pro:Quinterac lxpg0605$ pytest]
===== test session starts =====
platform darwin -- Python 3.6.2, pytest-5.2.2, py-1.8.0, pluggy-0.13.0
rootdir: /Users/lxpg0605/Desktop/Quinterac
collected 102 items

test_frontend/test_create_account.py ..... [ 9%]
test_frontend/test_delete_account.py ..... [ 18%]
test_frontend/test_deposit.py ..... [ 36%]
test_frontend/test_login.py ..... [ 49%]
test_frontend/test_logout.py ..... [ 60%]
test_frontend/test_transfer.py ..... [ 81%]
test_frontend/test_withdraw.py ..... [100%]

===== 102 passed in 0.72s =====
```

```

===== 102 passed in 1.44s =====
=====
C:\Users\Kelly\Documents\GitHub\Quinterac>pytest
===== test session starts =====
platform win32 -- Python 3.7.2, pytest-5.2.2, py-1.8.0, pluggy-0.13.0
rootdir: C:\Users\Kelly\Documents\GitHub\Quinterac
collected 102 items

test_frontend\test_create_account.py ..... [ 9%]
test_frontend\test_delete_account.py ..... [ 18%]
test_frontend\test_deposit.py ..... [ 36%]
test_frontend\test_login.py ..... [ 49%]
test_frontend\test_logout.py ..... [ 60%]
test_frontend\test_transfer.py ..... [ 81%]
test_frontend\test_withdraw.py ..... [100%]

===== 102 passed in 1.38s =====

```

Login Test: test_login.py

Test name	Purpose (what it was testing)	Nature of Failure (How the output is wrong)	Error in code	How to Fix
R1T1 Login in System	check if the user can login	when input “login” the expected output message is: ‘There are two types of mode you can login:’ which is different from terminal: Failure	miss operation to stop the while loop, terminal shows EOF error	add a “try except” in mode_check() outside the loop of asking the user for the mode
R1T1 Login in System	check if the user can login	when input “login” the expected output message is: ‘There are two types of mode you can login:’ which is different from terminal: ‘Please enter your transaction operations:There are two types of mode you can login:’	miss “\n” at the start of “There are two types of mode you can login.”	add a “\n” before “There are two types of mode you can login:” in mode_check()
R2T1 logout before login	can’t logout before login	when input “logout”, the expected output message is: ‘Error prompt for login failed.’ which is different from terminal: Failure	open_system() missing 1 required positional argument: 'self'	Add a self argument into the open_system() function

R2T1 logout before login	can't logout before login	when input "logout", the expected output message is: 'Error prompt for login failed.' which is different from terminal: Failure	wrong import statement, failure to import TransactionFile module from Frontend	Add " frontend. " before " TransactionFile " module in the import statement to make it import from the " frontend " folder
R2T1 logout before login	can't logout before login	when input "logout", the expected output message is: 'Error prompt for login failed.' which is different from terminal: Failure	miss operation to stop the while loop, terminal shows EOF error	add a " try except " in open_system() outside the loop of asking the user for a valid operation
R4T1 over login in atm	cannot login again after login an account in atm mode	when input "login" in atm mode, and "login" again, the expected output message is 'Error prompt for multiple login' which is different from terminal: 'Please enter your transaction operations:Error prompt for multiple login.'	miss "\n" in the beginning of "Error prompt for multiple login"	add "\n" before "Error prompt for multiple login." in login()

Logout Test: test_logout.py

Test Name	Purpose (What it was testing)	Nature of Failure (How the output is wrong)	Error in code	How to Fix
R3T2 logout before a successful login	test for logout is not accepted before login	when input "logout", "login", "atm", "logout", the expected output message is "Error prompt for login failed. There are seven transactions:..." which is different from terminal "Error prompt for login failed."	miss a loop to continue processing the overall program after print out the error message	add the function self. .check_trans() in logout() to make it go back to the beginning of program

Create Account Test: test_create_account.py

Test Name	Purpose (What it was testing)	Nature of Failure (How the output is wrong)	Error in code	How to Fix
R1T1 ATM mode createacct	can not create account in ATM mode	when input “login, atm, createacct” , the expected output message is “ Error prompt for ATM create account t ” which is different from terminal asks for enter a valid transaction name	Wrong transaction operation name in code because the code sets the valid input transaction as “create account”	In file frontend.py , function open_system(self) , modify the list element in check_list from ‘create account’ to ‘createacct’

Delete Account Test: test_delete_account.py

Test Name	Purpose (What it was testing)	Nature of Failure (How the output is wrong)	Error in code	How to Fix
R1T1 ATM mode delete account	can not delete account in ATM mode	when input “login, atm, deleteacct” , the expected output message is “ Error prompt for ATM delete account ” which is different from the terminal asks for enter a valid transaction name ”	Wrong transaction name in code because the code sets the valid input transaction as “delete account”	In file frontend.py , function open_system() , modify the list element in check_list from ‘delete account’ to ‘deleteacct’

Deposit Test: test_deposit.py

Test Name	Purpose (What it was testing)	Nature of Failure (How the output is wrong)	Error in code	How to Fix
R4T2 Deposit in agent mode	can deposit successfully in agent mode within limit	when input deposit amount 999999.99 in agent mode, the expected output message is ' Deposit successfully and ask for enter your transaction operations ' which is different from the terminal ' asks for entering a valid deposit amount '	The amount limit is set in dollars not in cents	In frontend.py modify the limit amount in deposit() by multiplying 100 and make each amount integer

R3T1 ATM deposit over daily limit	check if the deposit amount over daily limit (\$5,000) for atm	when input deposit \$2000 3 times in atm mode, the expected output message is ' Error! Over daily deposit limit! ', which is different from the terminal is 'Enter your amount: Error! Over daily deposit limit	miss "\n" in the beginning of "Error! Over daily deposit limit!"	add "\n" in the beginning of "Error! Over daily deposit limit!" in function deposit() in Frontend class
R3T3 Deposit in atm mode	check for invalid input deposit amount with letter in atm mode	when input \$99f9 in the atm deposit, the expected output is ' Enter a valid amount ', which is different from the terminal is 'Enter your amount: Enter a valid amount!	miss "\n" in the beginning of "Enter a valid amount!"	add "\n" in the beginning of "Enter a valid amount!" in function get_amount() in Accountcheck class
R4T3 Deposit in agent mode	check for invalid input deposit amount with letter in agent mode	when input \$99f9 in the agent deposit, the expected output is ' Enter a valid amount ', which is different from the terminal is 'Enter your amount: Enter a valid amount!	miss "\n" in the beginning of "Enter a valid amount!"	add "\n" in the beginning of "Enter a valid amount!" in function get_amount() in Accountcheck class

Withdraw Test: test_withdraw.py

Test Name	Purpose (What it was testing)	Nature of Failure (How the output is wrong)	Error in code	How to Fix
R3T2 atm withdraw amount input with letter	check for invalid input withdraw amount with letter in atm mode	when input \$200aa@ in the atm withdraw, the expected output is ' Enter a valid amount ' which is different from the terminal is 'Enter your amount:Enter a valid amount!'	miss "\n" in the beginning of "Enter a valid amount!"	add "\n" in the beginning of "Enter a valid amount!" in function get_amount() in Accountcheck class
R4T2 agent withdraw amount input with letter	check for invalid input withdraw amount with letter in agent mode	when input \$200aa@ in the agent withdraw, the expected output is ' Enter a valid amount ' which is different from the terminal is 'Enter your amount:Enter a valid amount!'	miss "\n" in the beginning of "Enter a valid amount!"	add "\n" in the beginning of "Enter a valid amount!" in function get_amount() in Accountcheck class

Transfer Test: test_transfer.py

Test Name	Purpose (What it was testing)	Nature of Failure (How the output is wrong)	Error in code	How to Fix
R1T4 Invalid from account number not in valid accounts list file	check invalid input of from transfer account number not in valid accounts list file in atm	when input a non-existing number '8888888' in atm, the expected output message is: 'Transfer account not existed', 'Enter your account number:' which is different from the terminal: 'Transfer to ----->', 'Enter your account number'	missing a condition statement to check if the from account number is in valid accounts list file	add a while loop to check the from transfer account is in the valid accounts list in transfer() function
R2T4 Invalid to account number not in valid accounts list file	check invalid input of to transfer account number not in valid accounts list file in atm	when input an non-existing number '8888888' in atm, the expected output message is: 'Transfer account not existed', 'Enter your account number:' which is different from the terminal: 'Transfer to ----->', 'Enter your account number'	missing a condition statement to check if the "to account" number is in valid accounts list file	add a while loop to check the to transfer account is in the valid accounts list in transfer() function
R5T1 ATM transfer over daily limit	check if the atm transfer amount over daily limit (\$10,000)	when input \$450000000 in atm transfer, the expected output is 'Over ATM transfer daily limit, enter a valid amount' which is different from the terminal: 'Over withdraw limit, enter a valid amount!'	spelling error in print statement, where "withdraw" should be "transfer"	In frontend.py , correct the word "withdraw" to "transfer" in transfer() function
R6T1 Agent transfer over daily amount	check if the agent transfer amount over daily limit (\$999,999.99)	when input \$999999999 in agent transfer, the expected output is: 'Over agent transfer daily limit, enter a valid amount', 'Enter your amount' which is different from the terminal: 'Enter your amount:Over withdraw limit, enter a valid amount!', 'Enter your amount:'	spelling error in print statement, where "withdraw" should be "transfer"	In frontend.py , correct the word "withdraw" to "transfer" in transfer() function