What:

It is a program that allows the user to input the amount they spend or add to an account. It is a cheap check book program. Right now, it does not save to a file, unsure of how to do this right now. Other than that, there is a button that leads to a pop-up window that allows you to enter your amount of money and then remove money from the account.

Completed:

Version 0.9 Final version, 3-9-22 @ 11:44pm

Buttons: Four buttons in total. One that leads to the pop-up window, one that closes the program. The other two buttons allow the user to add a credit or debit (The amount of money to the account)

Two images one showing a plus the other a minus to help the user to understand what they are doing.

Date is now set up and is working.

The if statement to help with making sure the user has enough money in the account is working.

There is a pop-up window when the user tries to remove more money than they have in the account.

Possible Work in the future:

Setting up the program to save to a text file (a file in general) and then it reloads the file and allows the user to pick up where they last left out.

Validation Testing:

User will enter an amount, for the first test it is 50 to add to the account. Then they will try to remove 75 dollars from the account. ( I fixed this already when I made the if statement, because I did not like that the user could remove money from the account when they did not have it. (I know most of the time this is possible, but I am thinking about if the user was trying to figure out why their check book does not add up this might help.

The second test, leaving the program running to show how if the user was to remove an amount of money from the account.

The third test is just showing if the user was to try to input anything that isn’t a float(number).

The fourth test is showing the program at its fullest. So the user is going to input their money and treat it like they are doing it daily. (As you can see it only takes whole numbers, I will fix that it takes a float instead of.)

The fifth test is showing the updated, version where you can now input a float number. It even rounds up, so the user is allowed to input a larger number after the decimal. (I am unsure of how to stop the user from this part, but that is okay I would rather it only go two decimals anyway.)

Test 1:

Graphical user interface, application

Description automatically generated

Test 2:

Graphical user interface

Description automatically generated with medium confidence

Test 3:

A screenshot of a computer

Description automatically generated with medium confidence

Test 4:

Graphical user interface

Description automatically generated

Test 5:

A screenshot of a computer

Description automatically generated with medium confidence