UNIVERSITY OF ONTARIO INSTITUTE OF TECHNOLOGY

SOFTWARE QUALITY ASSURANCE
(CSCI3060/SOFE3980)

"How to Git"
Banking System

Phase 6: Integration and Delivery

April 11th, 2016

Taylor Smith (100372402)
Luisa Rojas (100518772)

OBJECTIVE

The purpose of this last phase is to integrate the front end and the back end of two different banking systems into a working day-by-day Banking System.
In order to achieve this, two scripts will be developed: a daily script and a weekly script, each of which will allow text files to be read and outputted by both ends of the system, as originally required by the project requirements.

It's important to note that both programs will be invoked separately and will not be merged into one.

DESCRIPTION

The Front End used for this phase belongs to the Watermelon group, while the Back End program will be taken from the AT&T group.

DEFECT REPORT

Problems uncovered while integrating and running the Banking System can be found below, as well as their corresponding solutions:

PROBLEM FOUND	HOW IT WAS FIXED
Tried to run the Watermelon Front end with the AT&T tests, which resulted in a series of malfunctions. Among these, it was found that the Watermelon did not support spaces in account holder names.	Instead of piping in variables, substrings were used for reading in the current bank accounts file.
The AT&T group's test files included account numbers that were not strictly sequentially assigned, while the Watermelon's front end expected sequential account numbers.	The account numbers in the AT&T tests were accommodated to the format the front the used (Watermelon's) expected.
Attempted to use the Watermelon's test input files with their own front end; however, since the way the file is to be read in was changed in the source code, the tests no longer worked.	These tests needed to be accommodated (padded) as well.
The newly deposited funds class member in the front end was causing	The vector, as well as every other numerical member in the Standard

some valid transactions to fail.	class was initialized to zero.
AT&T's back end master bank accounts file inserts "00000.00" whenever a transaction with no monetary relevance is called (e.g. login,); however, the Watermelon front end outputs " " for this section instead.	In the back end, added a check where, if an empty string is found, do not parse it; otherwise, do.
Both the front end and the back end assign account numbers.	Front end account number assignation was removed - the back end will entirely take care of the process.
The back end had the last two fields inverted: <plan flag> <number of transactions> instead of <number of transactions> <plan flag>.	The fields mentioned were swapped back to they way the front end outputted them: <number of transactions> <plan flag>.