

CS673 Software Engineering
Team 1 - Trackr
Software Test Document



<u>Team Member</u>	<u>Role(s)</u>	<u>Signature</u>	<u>Date</u>
Weijie Liang	QA Leader	<u>Weijie Liang</u>	<u>05/29/2022</u>
Jean Dorancy	Team Leader	<u>Jean Dorancy</u>	<u>06/12/2022</u>

Revision history

<u>Version</u>	<u>Author</u>	<u>Date</u>	<u>Change</u>
<u>0.1</u>	<u>Weijie Liang</u>	<u>05/29/2022</u>	<u>Initial Test Result Added</u>
<u>0.2</u>	<u>Weijie Liang</u>	<u>06/11/2022</u>	<u>Updates on Manual Test and Automated Test Report</u>

[Testing Summary](#)

[Manuel Tests Reports](#)

[Automated Testing Reports](#)

[Testing Metrics](#)

[References](#)

[Glossary](#)

● Testing Summary

○ Unit Testing:

The purpose of unit testing is to verify that each unit of the application is developed and executed as designed in the user story. Developers use JUnit5 and Mockito to develop tests and reuse unit tests when units change.

○ Integration testing:

The purpose of integration testing is to test the integrated components to verify that they work as expected. These tests are performed manually. All results will be documented in this document. Unit tests can be part of a continuous integration framework.

○ System Testing:

The purpose of system testing is to test the system as a whole. All components are integrated to verify that the system works as expected. These tests are performed automatically.

○ Acceptance Testing:

The purpose of Acceptance Testing is to verify how well the system meets major functional requirements written in the user stories and some nonfunctional requirements. These tests are performed manually.

○ Regression Testing:

The purpose of Regression Tests is to re-run functional and non-functional tests to ensure that previously developed and tested software works as expected after changes are made. These tests are performed automatically every time we open a pull request through GitHub Actions.

● Manual Testing Report

We have a sample of the manual test cases below in this document. Please go to the sheet for all the cases: [CS673 STD TestCase team1](#)

Test case name	New or Old	Test item	Test priority	Dependencies	Preconditions
Create user account	New	Add a new user account	High	None	A user account information that has never been used before is used to successfully create a user
Login	New	Use existing user information to login	High	None	user account needs to exist to test its successful login
Test invalid scenarios 1(Create Duplicate user names)	New	Test the corresponding function with a value that does not match the input conditions	High	None	A used user account information is used in order to successfully report an error
Test invalid scenarios 2(wrong login info)	New	Test the corresponding function with a value that does not match the input conditions	High	None	An incorrect user account information is used in order to successfully report an error

Test case name	Input data	Test steps	Postconditions
Create user account	First name(WEIJIE); Last name(LIANG); Username(WEL157); Email address(weijiel@bu.edu); Password(123456)	1. Start the page by entering: https://trackr-dev.herokuapp.com/ 2. Fill in the input box with the appropriate information 3. Click "Sign Up!" Button 4. See If a new account is successfully created	The user is able to log in with the created account

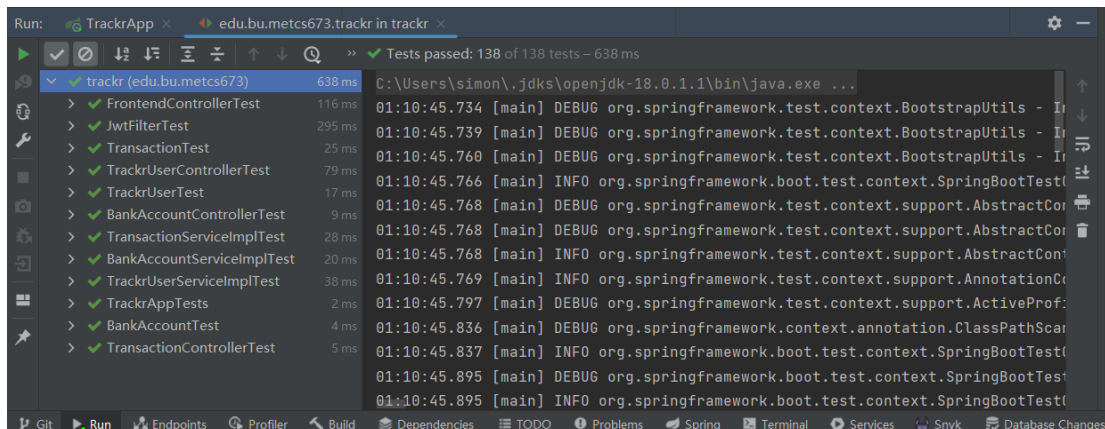
Login	Username(WEL157); Password(123456)	<ol style="list-style-type: none"> 1. Start the page by entering: https://trackr-dev.herokuapp.com/ 2. Click the "Login" Button at the top of the page 3. Fill in the input box with the appropriate information 4. Click the "Login" Button 5. See if the account login successfully 	The user login to their own account
Test invalid scenarios 1(Create Duplicate user names)	First name(WEIJIE); Last name(LIANG); Username(WEL157); Email address(weijiel@bu.edu); Password(123456)	<ol style="list-style-type: none"> 1. After the former registration, click the "Home" button. 2. Fill in the input box with the appropriate information 3. Click "Sign Up!" Button 4. See If a new account is successfully created 	The account should not be created
Test invalid scenarios 2(wrong login info)	Username(WEL157); Password(987654)	<ol style="list-style-type: none"> 1. Start the page by entering: https://trackr-dev.herokuapp.com/ 2. Click the "Login" Button at the top of the page 3. Fill in the input box with the appropriate information 4. Click the "Login" Button 5. See if the account login successfully 	The authentication should not be passed

Test case name	Expected output	Actual output	Pass or Fail	Bug id/link	Additional notes
Create user account	Account successfully created!	Account successfully created!	Pass	None	None
Login	Successfully authenticated!	Successfully authenticated!	Pass	None	None
Test invalid scenarios 1(Create Duplicate user names)	Invalid USERNAME value. Please use another value.	Invalid USERNAME value. Please use another value.	Pass	None	Also tested the case of different user names but the same mailbox, account added successfully, need to follow up to discuss whether to reject duplicate mailboxes
Test invalid scenarios 2(wrong login info)	Invalid Login Credentials	Invalid Login Credentials	Pass	None	None

● Automated Testing Report

Describe briefly the automated testing you have done, including where the test code resides in your code repository, what test frameworks are used, and the screen shots or generated testing report.

To test our API, we will use the Mockito framework to write our automation tests. There are 138 automation tests completed so far, and they are geared toward the Bank Account, User, and Transaction aspects of the functionality.



The screenshots above show the results of the automated tests we have performed.

● Testing Metrics

In this section, you shall report any metrics used for the evaluation, e.g. # of test cases, test coverage, defects rate, etc.

Unit Test Code Coverage:

Current scope: all classes

Overall Coverage Summary

Package	Class, %	Method, %	Line, %
all classes	88.9% (24/27)	74.3% (110/148)	65.3% (216/331)

Coverage Breakdown

Package ▲	Class, %	Method, %	Line, %
edu.bu.metcs673.trackr	100% (1/1)	66.7% (2/3)	66.7% (2/3)
edu.bu.metcs673.trackr.api	100% (1/1)	63.6% (7/11)	50% (7/14)
edu.bu.metcs673.trackr.bankaccount	100% (6/6)	77.1% (27/35)	57.5% (42/73)
edu.bu.metcs673.trackr.common	50% (2/4)	44.4% (4/9)	35.7% (5/14)
edu.bu.metcs673.trackr.controller	100% (2/2)	42.9% (3/7)	23.1% (3/13)
edu.bu.metcs673.trackr.security	100% (3/3)	100% (12/12)	100% (63/63)
edu.bu.metcs673.trackr.transaction	100% (5/5)	82.9% (29/35)	74.6% (53/71)
edu.bu.metcs673.trackr.user	80% (4/5)	72.2% (26/36)	51.2% (41/80)

generated on 2022-06-10 02:18