Venn Diagram Testing Document

Reuben Ninan, Eric Kwok, Edward Nwogwugwu

Test Cases.....2
Test Case 1: New User Button.....2
Test Case 2: Login Button.....3
Test Case 3: Delete User Button.....4
Test Case 4: Entry Button.....5
Test Case 5: Set Title Button.....6
Test Case 6: Change Title Button.....7
Test Case 7: Color Button.....8
Test case 8: Clear All Button.....9
Test case 9: Export Button.....10
Test case 10: Import Button.....11
Test case 11: Undo Button.....12
Test case 12: Redo Button.....13
Test case 13: Game Mode Button.....14
Test Case Sufficiency.....15
Test Case 1 Sufficiency: New User Button.....15
Test Case 2 Sufficiency: Login Button.....15
Test Case 3 Sufficiency: Delete User Button.....15
Test Case 4 Sufficiency: Entry Button.....15
Test Case 5 Sufficiency: Set Title Button.....15
Test Case 6 Sufficiency: Change Title Button.....16
Test Case 7 Sufficiency: Color Button.....16
Test Case 8 Sufficiency: Clear All Button.....16
Test Case 9 Sufficiency: Export Button.....16
Test Case 10 Sufficiency: Import Button.....17
Test Case 11 Sufficiency: Undo Button.....17
Test Case 12 Sufficiency: Redo Button.....17
Test Case 13 Sufficiency: Game Mode Button.....17
Test Coverage.....18

Test Cases

Test Case 1: New User Button

Test Case 1:

1) **Type into Username:** Tom
2) **Type into Password:** Uni
3) **Click New User Button**

Expected Result

1) Expected result should be “New user: Tom has been created” When the user clicks ok, the Venn Diagram window will be displayed

Test Case 2:

1) **Close Previous Session**
2) **Type into Username:** Tom
3) **Type into Password:** Uni
4) **Click New User Button**

Expected Result

1) Expected result should be “Please enter a valid entry”

Test Case 2: Login Button

Test Case 1:

1) **Based On Previous Session**
2) **Type into Username:** Tom
3) **Type Into Password:** Uni
4) **Click the Login Button**

Expected Results

1) Expected result should be "**Welcome**" and the Venn Diagram Window should Pop Up

Test Case 2:

1) **Close Previous Session**
2) **Type into Username:** Tom
3) **Type Into Password:** Uni123
4) **Click the Login Button**

Expected Results

1) Expected result should be "**Incorrect**"

Test Case 3: Delete User Button

Test Case 1:

1) **From Previous session**
2) **Type into Username:** Tom
2) **Type into Password:** Uni
3) **Click the Delete User Button**

Expected Output

1) The expected output should be "User: Tom has been deleted"

Test Case 2:

1) **From Previous Session**
2) **Type into Username:** Tom
3) **Type into Password:** Uni
4) **Click the Delete User Button**

Expected Output

1) The expected output should be "User: Tom does not exist"

Test Case 4: Entry Button

Test Case 1:

1) **Use Test Case 1.1 Again**
2) **Input:** EECS2311
3) **Press "enter**"
4) **Drag the entry outside the venn diagram**
5) **Click Enter**
6) **Drag the entry back into the left circle**

Expected Result

1) A new entry is generated with the content "EECS2311"
2) A warning sign should pop up at step 4
3) Entry is inside the left circle

Test Case 2:

1) **Default Screen**
2) **Press**"enter"

Expected Result

1) Text pop up displaying "Please enter valid entry"

Test Case 5: Set Title Button

Test Case 1:

1) **User Input:** Software engineering
2) **Click the Set Title button**

Expected Results:

1) The title of the category was successfully set to "Software engineering"

Test Case 2:

1) **From Previous Test Case**
2) **Click the Change Title Button**
3) **Empty the textfield**
4) **Click the Set Title Button**

Expected Results:

The title will not change, and display "Please enter a valid entry."

Test Case 6: Change Title Button

Test Case 1:

1) **Input into Title:** Software Tools
2) **Click the Set Title Button**
3) **Press Enter**
4) **Click the Change Title Button**

Expected Result

1) "Title is now Unlocked"
2) Type in new Title and Click on Set Title

Test Case 2:

1) **Make the title field Empty**
2) **Click the Change Title Button**

Expected Result

1) "No Title Yet"

Test Case 7: Color Button

Test case 1:

1) **Click on the Circle 1 Color Button**
2) **Set Color to Yellow**

Expected Result

1) The color Yellow appears on the Left Circle

Test case 2:

1) **Click on the Circle 2 Color Button**
2) **Set Color to Green**

Expected Result

1) The color Green appears on the Right Circle

Test Case 8: Clear All Button

Test Case 1:

1) **Use Test Case 4 four times**
2) **Drag two of the entries into the left side**
3) **Drag two entries into the right side**
4) **Click Clear All Button**

Expected Results:

1) The entries are cleared and you are back the the default screen

Test Case 2:

1) **Nothing is done yet**
2) **Click the Set Clear All Button**

Expected Results:

1) The software will display “It is Already Empty”.

Test Case 9: Export Button

Test Case 1:

1) **Create four entries one by one**
2) **Entries are “one”, “two”, “three”, “four”**
3) **Drag 1 and 2 to the left of the venn diagram**
4) **Drag 3 in the middle of the venn diagram**
5) **Drag 4 in to the right of the venn diagram**
6) **Click Export Button**
7) **Save the file name**

Expected Results:

1) The software will display "Entry saved", then a new window will pop up, the user can select the directory to save the content.
2) Upon opening the xlsx file the corresponding entries are in their respective columns that indicate what side they were on

Test Case 2:

1) **Nothing is done**
2) **Click Export Button**

Expected Results:

1) The software will tell the user that there are no entries to be saved

Test Case 10: Import Button

Test Case 1:

1) **Click on the Import Button**
2) **Choose a 'xlsx' file that is INSIDE of the project folder and has the same formatting as a exported file (Refer to User Manual)**
3) **Click 'Save'**

Expected Results:

1) The User will be prompted a warning whether to import the file since all previous changes will be removed
2) If the user decides to do so their previous changes will be lost and they will be given the information from the file chosen

Test Case 2:

1) **Click on Import Button**
2) **Choose Invalid File**

Expected Results:

1) The software will tell the user that the file is invalid

Test Case 11: Undo Button

Test Case 1:

1) **Press the Circle 1 Color Button**
2) **Press any color**
3) **Create a new Entry called “Apple”**
4) **Press the Undo Button**
5) **Press the Undo Button**
6) **Press the Circle 2 Color Button**
7) **Press any Color**
8) **Press the Undo Button**

Expected Results:

1) The expected result is that the screen is back to its default settings without anything added onto it all the previous colors are entries are gone

Test Case 2:

1) **Press the Undo Button**

Expected Results:

1) Nothing happens because there are no actions that can be undone

Test Case 12: Redo Button

Test Case 1:

1) **Follow the steps on the Undo Button Test Case #1**
2) **Press the Redo Button**
3) **Press the Redo Button**
4) **Press the Redo Button**

Expected Results:

1) The expected result is that the screen is filled with two colors on circle1 and circle2. Also there should be a entry “Apple” on the Venn Diagram

Test Case 2:

1) **Press the Redo Button**

Expected Results:

2) Nothing happens because there are no actions that can be redone

Test Case 13: Game Mode Button

Test Case 1:

1) **Press the Game Mode Button**
2) **Press Ok**

Expected Results:

1) The expected result is that after pressing Ok a new window should appear prompting the user to the new game mode. The user should be able to close this mode at any time and return to their previous normal session

Test Case Sufficiency

Test Case 1 Sufficiency: Login Button

These test cases are sufficient because it is needed to check if the login is consistent with the users logging in so that they can access the venn diagram window.

Test Case 2 Sufficiency: New User Button

It is important to test the new user button to enable users create a new account and avoid having the same username.

Test Case 3 Sufficiency: Delete User Button

Delete Button test case is sufficiency, because we need to know whether the user can successfully delete the existing account when the user wants to clear an account, and the data in the data is also erased

Test Case 4 Sufficiency: Entry Button

Entry Button test case is sufficient because it is very important for the creation of a venn diagram entry. We already confirm that the user can input and automatically generate draggable entries.

Test Case 5 Sufficiency: Set Title Button

Set Title Button is sufficient, we already make sure the client can use Set Title Button to set a valid name, except the empty string, and make sure only Change Title Button can overwrite it.

Test Case 6 Sufficiency: Change Title Button

The change title button test case is sufficient because it is important for users, if they want to change the title and to alert them when there is no title at all.

Test Case 7 Sufficiency: Color Button

The Color Button test case is sufficient because we have guaranteed that users can print the three categories of venn diagrams with different background colors. This already includes the situation that all customers will want to implement the color button

Test Case 8 Sufficiency: Clear All Button

The Clear All button is sufficient because it is needed when users want to start a new venn diagram session or when they are just done with a session. The test case is important because it sees if the user's entries have been completely reset to default.

Test Case 9 Sufficiency: Export Button

The Export Button test case is sufficient because we need to test that entries can accurately be transferred into a csv file without being on the wrong side or having entries missing. It is important to make sure that the user can close the application knowing their information is accurately being stored somewhere.

Test Case 10 Sufficiency: Import Button

The Import Button test case is sufficient because we need to test whether the contents on a csv file can be accurately transferred into the application. The user should be able to take and bring back any changes made before in a previous session onto the current one.

Test Case 11 Sufficiency: Undo Button

The Undo Button is sufficient because users should be able to change their mind about certain actions that they previously made without having to trace their steps. Pressing this button should bring them to their previous step

Test Case 12 Sufficiency: Redo Button

The Redo Button is sufficient because users should be able to change their mind about a certain change that they already reverted; however, they would like to change back. Pressing this button should revert that action.

Test Case 13 Sufficiency: Game Mode Button

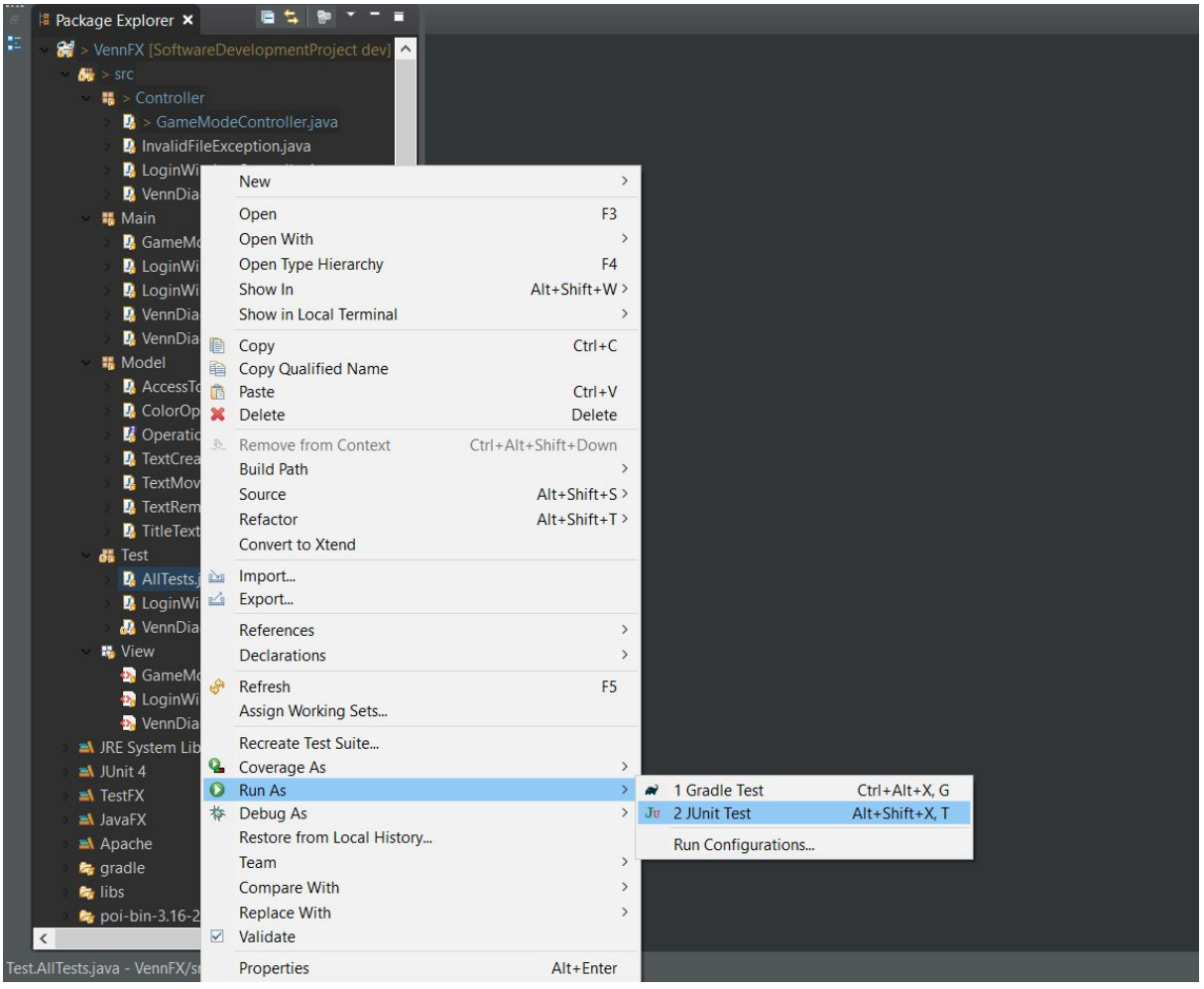
The Game Mode Button is sufficient because the user should be able to join the game mode without losing their changes in the normal mode session. After pressing the button they should be welcomed to the game mode with a new window.

Test Coverage

In our current version extensive JUnit testing is implemented and the testing is done Automatically with our testing class "AllTests.java"

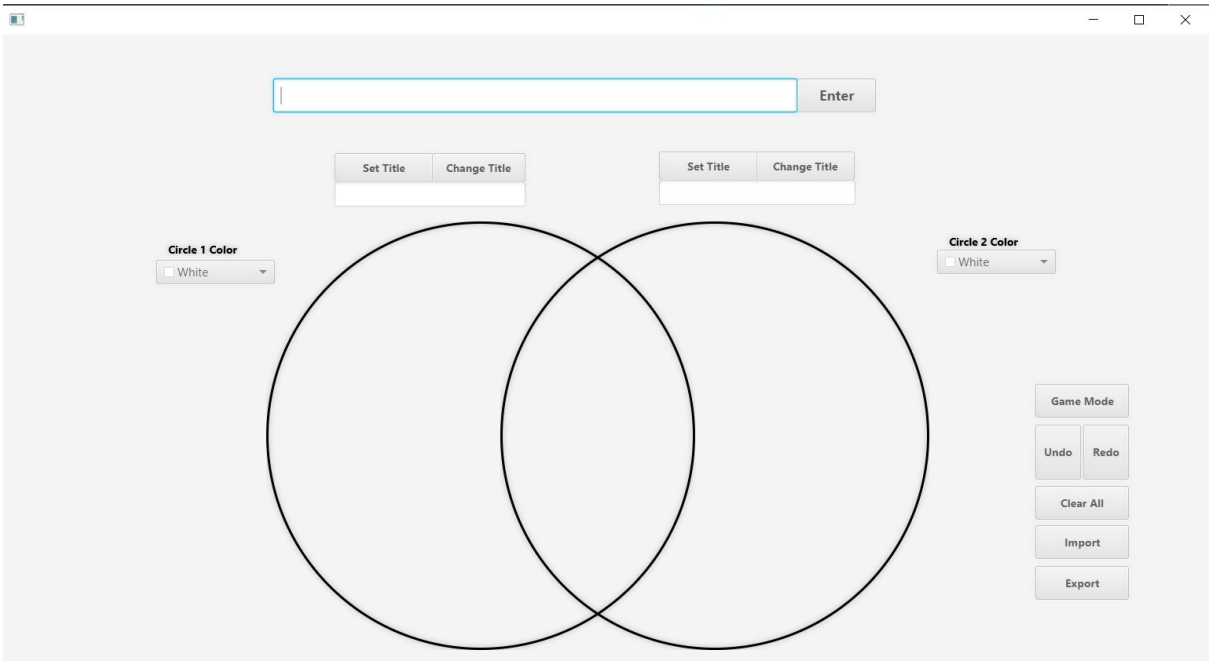
Steps to use this testing class:

1) Open Project in eclipse IDE
2) Click on the Coverage Button Options and then Right-Click the Class AllTests.java and Press Run



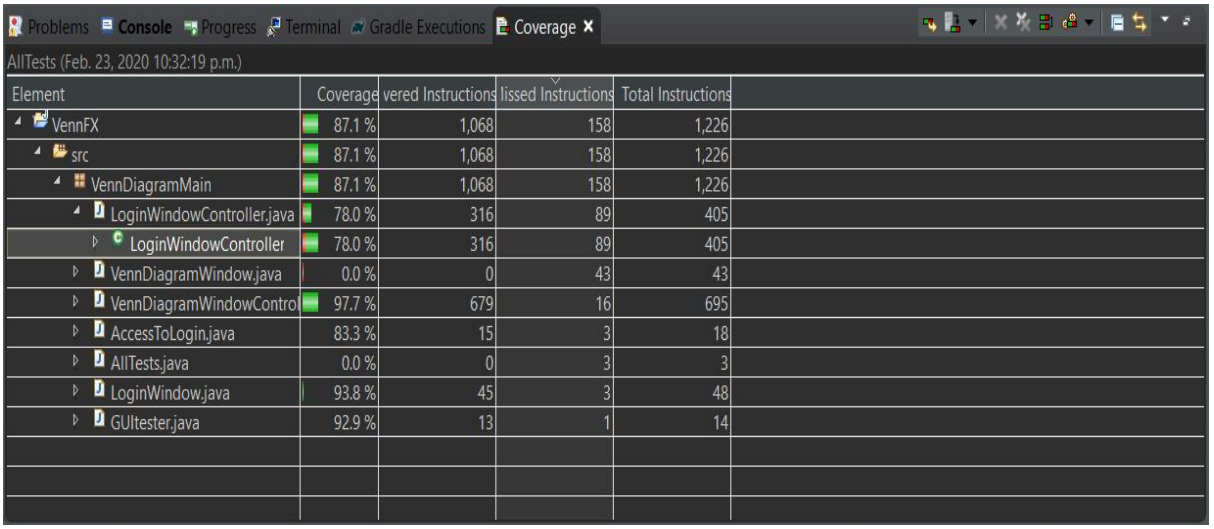
Coverage is the Run Icon With The Green And Red Bar Underneath

3) Now the test should be running and all you need to do is watch them run



This Is What Should Pop Up

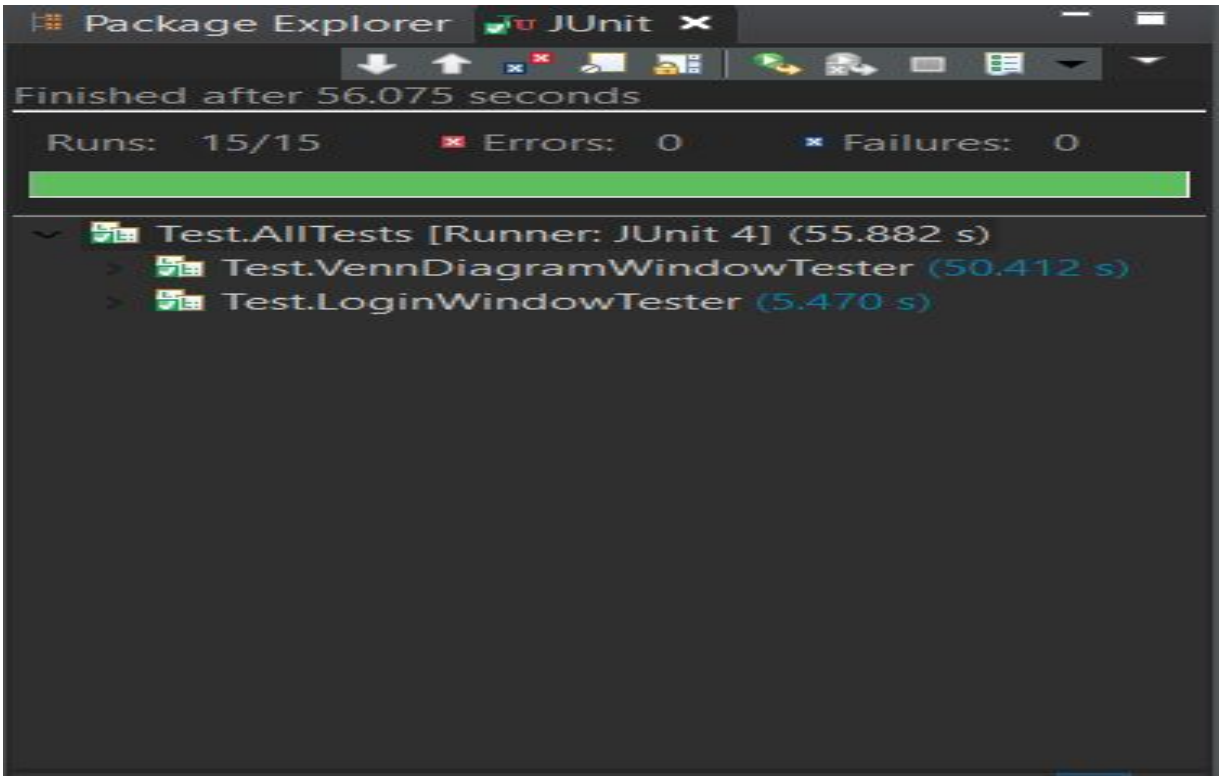
4) Now to test the functions in the application you can use the test cases that we provided above in the test cases section. (Keep in mind that once you close the application then the coverage results will show). So try to test the functions in a manner so that everything can be covered before closing.



Element	Coverage	vered Instructions	lissed Instructions	Total Instructions
VennFX	87.1 %	1,068	158	1,226
src	87.1 %	1,068	158	1,226
VennDiagramMain	87.1 %	1,068	158	1,226
LoginWindowController.java	78.0 %	316	89	405
LoginWindowController	78.0 %	316	89	405
VennDiagramWindow.java	0.0 %	0	43	43
VennDiagramWindowContro	97.7 %	679	16	695
AccessToLogin.java	83.3 %	15	3	18
AllTests.java	0.0 %	0	3	3
LoginWindow.java	93.8 %	45	3	48
GUItester.java	92.9 %	13	1	14

Example of a good amount of Coverage from testing the GUI

If you run the class without coverage, it is expected that all the test cases pass so the tests screen should appear as all green in this case



All Test Cases Passed

These tests will be automated and users won't have to manually complete the tests themselves to determine the coverage. The coverage that we got from testing most of the test provided above gave a good indication that we covered most of the code we wrote and that the majority of it was used in the testing thus, validating the completeness of our tests and programm.