Project Name: Project 1: Voting System **Team#**

Test Stage: Unit __ System _x_ **Test Date: 3/27/2024**

Test Case ID#: ST_01 **Name(s) of Testers: Derrick Dischinger**

Test Description: Manually testing the system with an accurately formatted CPL document

Indicate where are you storing the tests (what file) and the name of the method/functions being used.

Automated: yes___ no _x__

Results: Pass __x___ Fail_________

Preconditions for Test: A properly formatted CPL document is in the BallotFile folder, all classes are assumed to have been completed

<table>
<tr><th>Step
#
1</th><th>Test Step
Description</th><th>Test
Data</th><th>Expected
Result</th><th>Actual
Result</th><th>Notes</th></tr>
<tr><td>2</td><td>Run main.java with no arguments</td><td>Please enter the name of the ballot file:</td><td>“Please enter the name of the ballot file:” is printed to the terminal</td><td>“Please enter the name of the ballot file:” is printed to the terminal</td><td></td></tr>
<tr><td>3</td><td>Input the file name</td><td>“CPLInput01.txt</td><td>Generating Results...
Audit File Saved:
…/Documents/AuditFiles/CPL_Election_Results_xxx.txt</td><td>Generating Results...
Audit File Saved:
/Users/derrickdischinger/Documents/AuditFiles/CPL_Election_Results_xxx.txt</td><td>The xxx at the end are numbers of the filename that change based on the computer system time</td></tr>
<tr><td>4</td><td>Check output file</td><td>CPL_Election_Results_1711576440435.txt</td><td>should be ResultDataCPL.toString()</td><td>It is equal to the ResultDataCPL.toString()</td><td></td></tr>
<tr><td></td><td>Terminal also outputs the direct results</td><td>terminal output</td><td>---
Winning | Seat | Seat
Parties | Winners | Won

Republican | Allen | 1</td><td>---
Winning | Seat | Seat
Parties | Winners | Won

Republican | Allen | 1
Green | Sally | 2
Democratic | Mary | 3
---</td><td></td></tr>
</table>

			Green | Sally | 2 Democratic | Mary | 3 -- -----------		

Post condition(s) for Test:

Properly formatted output document

Project Name: Project 1: Voting System   Team#

Test Stage: Unit __ System _x_

Test Date: 3/27/2024

Test Case ID#: ST_02

Name(s) of Testers: Derrick Dischinger

Test Description: Manually testing the system with an accurately formatted OPL document

Indicate where are you storing the tests (what file) and the name of the method/functions being used.

Automated: yes ____ no _x__

Results: Pass __x___ Fail________

Preconditions for Test: A properly formatted OPL document is in the BallotFile folder, all classes are assumed to have been completed

Step # 1	Test Step Description	Test Data	Expected Result	Actual Result	Notes
2	Run main.java with no arguments	Please enter the name of the ballot file:	“Please enter the name of the ballot file:” is printed to the terminal	“Please enter the name of the ballot file:” is printed to the terminal	
3	Input the file name	“OPLInput01.txt	Generating Results... Audit File Saved: Audit File	Generating Results... Audit File Saved: /Audit File Saved:	The xxx at the end are numbers of the filename

<table>
<tr><td></td><td></td><td></td><td>Saved:
…/Documents/AuditFiles/OPL_Election_Results_xxx.txt</td><td>/Users/derrickdischinger/Documents/AuditFiles/OPL_Election_Results_xxx.txt</td><td>that change based on the computer system time</td></tr>
<tr><td>4</td><td>Check output file</td><td>OPL_Election_Results_1711576440435.txt</td><td>should be ResultDataOPL.toString()</td><td>It is equal to the ResultDataOPL.toString()</td><td></td></tr>
<tr><td></td><td>Terminal also outputs the direct results</td><td>terminal output</td><td>--
----
Winning | Seat
| Seat | Number
Parties | Winners
| Won | Of Votes

Pluto | Mariah
| 1 |24136
Green | Jonah
| 2 |24014
Green | Radius
| 3 |24006
Pluto | Becky
| 4 |23971

-----------------</td><td>--
Winning | Seat | Seat |
Number
Parties | Winners | Won
| Of Votes

Pluto | Mariah | 1
|24136
Green | Jonah | 2
|24014
Green | Radius | 3
|24006
Pluto | Becky | 4
|23971
--</td><td></td></tr>
<tr><td></td><td></td><td></td><td></td><td></td><td></td></tr>
</table>

Post condition(s) for Test:

Properly formatted output document

Project Name: Project 1: Voting System   Team#

Test Stage: Unit __ System _x_

Test Date: 3/27/2024

Test Case ID#: ST_03

Name(s) of Testers: Derrick Dischinger

Test Description: Manually testing the system with an accurately formatted OPL document but the user inputs an incorrect filename first

Indicate where are you storing the tests (what file) and the name of the method/functions being used.

Automated: yes___ no _x__

Results: Pass __x___ Fail________

Preconditions for Test: A properly formatted OPL document is in the BallotFile folder, all classes are assumed to have been completed

<table>
<tr><th>Step
#
1</th><th>Test Step
Description</th><th>Test
Data</th><th>Expected
Result</th><th>Actual
Result</th><th>Notes</th></tr>
<tr><td>2</td><td>Run main.java with no arguments</td><td>Please enter the name of the ballot file:</td><td>“Please enter the name of the ballot file:” is printed to the terminal</td><td>“Please enter the name of the ballot file:” is printed to the terminal</td><td></td></tr>
<tr><td>3</td><td>Input the incorrect file name</td><td>terminal output, user input</td><td>Error: Invalid file. Input file must be a valid CSV
Please enter another file name or the letter "q" to quit</td><td>Error: Invalid file. Input file must be a valid CSV
Please enter another file name or the letter "q" to quit</td><td></td></tr>
<tr><td>4</td><td>Input the correct file name</td><td>“OPLInput01.txt</td><td>Generating Results...
Audit File Saved: Audit File Saved: …/Documents/AuditFiles/OPL_Election_Results_xxx.txt</td><td>Generating Results...
Audit File Saved: /Audit File Saved: /Users/derrickdischinger/Documents/AuditFiles/OPL_Election_Results_xxx.txt</td><td></td></tr>
<tr><td></td><td>Check output file</td><td>OPL_Election_Results_xxx.txt</td><td>should be ResultDataOPL.toString()</td><td>It is equal to the ResultDataOPL.toString()</td><td></td></tr>
<tr><td></td><td>Terminal also outputs the direct results</td><td>terminal output</td><td>---------------------------------------
----
 Winning | Seat
 | Seat | Number
 Parties | Winners
 | Won | Of Votes

 Pluto | Mariah
 | 1 |24136
 Green | Jonah
 | 2 |24014
 Green | Radius
 | 3 |24006
 Pluto | Becky
 | 4 |23971
---------------------------------------</td><td>--

 Winning | Seat |
Seat | Number
 Parties | Winners |
Won | Of Votes

 Pluto | Mariah | 1
 |24136
 Green | Jonah | 2
 |24014
 Green | Radius | 3
 |24006
 Pluto | Becky | 4
 |23971
--</td><td></td></tr>
</table>

			------------------	-----	

Post condition(s) for Test:
Properly formatted output document

Project Name: Project 1: Voting System **Team#**

Test Stage: Unit __ System _x_ **Test Date:** **3/27/2024**

Test Case ID#: ST_04 **Name(s) of Testers: Derrick Dischinger**

Test Description: Manually testing the system with an accurately formatted CPL document but the user inputs an incorrect filename first

Indicate where are you storing the tests (what file) and the name of the method/functions being used.

Automated: yes no _x__

Results: Pass __x___ Fail________

Preconditions for Test: A properly formatted CPL document is in the BallotFile folder, all classes are assumed to have been completed

Step # 1	Test Step Description	Test Data	Expected Result	Actual Result	Notes
2	Run main.java with no arguments	Please enter the name of the ballot file:	“Please enter the name of the ballot file:” is printed to the terminal	“Please enter the name of the ballot file:” is printed to the terminal	

<table>
<tr><td>3</td><td>Input the incorrect file name</td><td>terminal output, user input</td><td>Error: Invalid file. Input file must be a valid CSV
Please enter another file name or the letter "q" to quit</td><td>Error: Invalid file. Input file must be a valid CSV
Please enter another file name or the letter "q" to quit</td><td></td></tr>
<tr><td>4</td><td>Input the correct file name</td><td>“CPLInput01.txt</td><td>Generating Results...
Audit File Saved: Audit File Saved: …/Documents/AuditFiles/OPL_Election_Results_xxx.txt</td><td>Generating Results...
Audit File Saved: /Audit File Saved: /Users/derrickdischinger/Documents/AuditFiles/OPL_Election_Results_xxx.txt</td><td></td></tr>
<tr><td></td><td>Check output file</td><td>CPL_Election_Results_xxx.txt</td><td>should be ResultDataCPL.toString()</td><td>It is equal to the ResultDataCPL.toString()</td><td></td></tr>
<tr><td></td><td>Terminal also outputs the direct results</td><td>terminal output</td><td>--
----
 Winning | Seat
| Seat | Number
 Parties | Winners
| Won | Of Votes

 Pluto | Mariah
| 1 |24136
 Green | Jonah
| 2 |24014
 Green | Radius
| 3 |24006
 Pluto | Becky
| 4 |23971

------------------</td><td>--

 Winning | Seat |
Seat | Number
 Parties | Winners |
Won | Of Votes

 Pluto | Mariah | 1
|24136
 Green | Jonah | 2
|24014
 Green | Radius | 3
|24006
 Pluto | Becky | 4
|23971

-----</td><td></td></tr>
</table>

Project Name: Project 1: Voting System **Team#**

Test Stage: Unit __ System _x_ **Test Date: 3/27/2024**

Test Case ID#: ST_05 **Name(s) of Testers: Derrick Dischinger**

Test Description: Manually testing the system with an incorrect formatted CPL document

Indicate where are you storing the tests (what file) and the name of the method/functions being used.

Automated: yes___ no _x__

Results: Pass __x___ Fail_________

Preconditions for Test: A not formatted CPL document is in the BallotFile folder, all classes are assumed to have been completed

Step # 1	Test Step Description	Test Data	Expected Result	Actual Result	Notes
2	Run main.java with no arguments	Please enter the name of the ballot file:	“Please enter the name of the ballot file:” is printed to the terminal	“Please enter the name of the ballot file:” is printed to the terminal	
3	Input the file name	“CPLInput02.txt	Generating Results... Not enough digits	Generating Results... Not enough digits	program exits
4					

Post condition(s) for Test:

program exits gracefully

Project Name: Project 1: Voting System **Team#**

Test Stage: Unit _x_ System __ **Test Date: 3/27/2024**

Test Case ID#: ST_06 **Name(s) of Testers: Derrick Dischinger**

Test Description: Manually testing the main method with a q as the first input quitting the program

Indicate where are you storing the tests (what file) and the name of the method/functions being used.

Automated: yes___ no _x__

Results: Pass __x___ Fail________

Preconditions for Test: A not formatted CPL document is in the BallotFile folder, all classes are assumed to have been completed

Step # 1	Test Step Description	Test Data	Expected Result	Actual Result	Notes
2	Run main.java with no arguments	Please enter the name of the ballot file:	“Please enter the name of the ballot file:” is printed to the terminal	“Please enter the name of the ballot file:” is printed to the terminal	
3	Input q		program exits	program exits	
4					

Post condition(s) for Test:

program exits gracefully

Project Name: Project 1: Voting System **Team#**

Test Stage: Unit __ System _x_ **Test Date:** 3/27/2024

Test Case ID#: ST_07 **Name(s) of Testers: Derrick Dischinger**

Test Description: Manually testing the system with an input document that has no votes

Indicate where are you storing the tests (what file) and the name of the method/functions being used.

Automated: yes no _x

Results: Pass __x___ Fail_______

Preconditions for Test: A voteless CPL document is in the BallotFile folder, all classes are assumed to have been completed

Step # 1	Test Step Description	Test Data	Expected Result	Actual Result	Notes
2	Run main.java with no arguments	Please enter the name of the ballot file:	“Please enter the name of the ballot file:” is printed to the terminal	“Please enter the name of the ballot file:” is printed to the terminal	
3	Input filename	filename	Generating Results... Not enough digits	Generating Results... Not enough digits	
4					

Post condition(s) for Test:

program exits gracefully

Project Name: The project #, name of your system, and the team#

Test Stage: Indicate whether it is a unit test or a system test.

Test Date: The date the test was performed.

Test Case ID#: A unique ID is required. Decide on a naming convention and use numbering. Example: Ballot_Shuffle_1

Name(s) of Testers: List the names of anyone involved in running this test case.

Test Description: Describe briefly the test objective.

Automated: Indicate if the test is completely automated or being checked manually. (If you have methods running the tests and checking results, select "yes". If you are manually checking results, indicate manual by selecting the "no.")

Results: Indicate if the test passed or failed.

Step #: You will be listing the test steps in order. This number is the step number in the process.

Test Step Description: Details of the test step.

Test Data: What the test data will be for this step. Be clear on what the input data will be. If using a specific file, be clear on the name.

Expected Result: What result are you expecting from the program component or system.

Actual Result: What result were returned based on the test.

Post condition for Test: What will be true after the test has been run? Has the state of the system changed in any way?

Notes: Comments and notes for you and your team members.