

Project Name : Project2, Voting System
Test Stage : Unit 1
Test Case ID#: 1

Team# : 7
Test Date : 12/9/2019
Name(s) of Testers:
Danny, Raymond, Raymond, Sanfer

Test Description:

This tests the Audit class and all of its methods

Indicate where are you storing the tests (what file) and the name of the method/functions being used. AuditTest.java,
/testing/audit.txt

Automated : Yes **No**

Results : **Pass** Fail

Preconditions for Test:

No audit file of the same name as the one given to the constructor has already been created in the same directory as the program, and the “.txt” extension is included in the file name.

Step #	Test Step Description	Test Data	Expected Result	Actual Result	Notes
1	Create an Audit file with a file name and teller and call writeTeller for CPL file	same info as TestFiles.SMALL_CPL_TELLER	TestFiles.SMALL_CPL_TELLER	TestFiles.SMALL_CPL_TELLER	

2	Create an Audit file with a file name and teller and call writeTeller for OPL file	same info as TestFiles.SMALL_OPL_TELLER	TestFiles.SMALL_OPL_TELLER	TestFiles.SMALL_OPL_TELLER	
2	call tallyBallot and then call writeBallotWithWinner	ballot = ',,,,,1,,' winner = 'Bob'	,,,,,1,, (Bob)	,,,,,1,, (Bob)	

Post conditions for Test:

Audit file is created (test.txt) with the proper data and new lines filled in.

Project Name : Project2, Voting System
Test Stage : Unit 1
Test Case ID#: 2

Team# : 7
Test Date : 12/9/2019
Name(s) of Testers:
Danny, Raymond, Raymond, Sanfer

Test Description:

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

Automated : **Yes** No
Results : **Pass** Fail

Preconditions for Test:

class objects don't exist

Step#	Test Step Description	Test Data	Expected Result	Actual Result	Notes
1	create OPL election file parser object and check for its existence	TestFiles.SMALL_OPL	object	object	
2	same but for CPL	TestFiles.SMALL_CPL	object	object	
3	create OPL teller and check for its existence	same as 1	object	object	

4	same but for CPL	same as 2	object	object	
5	parse thru OPL and check audit file	TestFiles.SMALL_OPL_AUDIT	file contents	same file contents	
6	same but for CPL	TestFiles.Small_CPL_AUDIT	file contents	same file contents	

Post conditions for Test:

class objects exist

Project Name : Project2, Voting System
Test Stage : Unit 1
Test Case ID#: 3

Team# : 7
Test Date : 12/9/2019
Name(s) of Testers:
Danny, Raymond, Raymond, Sanfer

Test Description:

This tests the ElectionFile class and all of its methods.

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

Automated : Yes

Results : Pass Fail

Preconditions for Test:

An ElectionFile exists and is in a valid format, the assumption is that any file passed to this class has already been validated by FileNameValidator

Step#	Test Step Description	Test Data	Expected Result	Actual Result	Notes
1	Create an election file with test data call getElectionFileType() check to ensure it returns the correct type	CPLTestFile.csv	CPL	CPL	

2	Create an election file with test data call <code>getElectionFileType()</code> check to ensure it returns the correct type	OPLTestFile.csv	OPL	OPL	
3	Create an election file with test data and call <code>getNextLine()</code>	TestFiles.OPL_T WO_LINES	"This is the second line."	"This is the second line."	
4	Create an election file with test data and call <code>hasNextLine()</code>	CPLTestFile.csv	True	True	
5	same as 4 but there is no next line	same as 4	False	False	
6	Create an election file with test data and call <code>hasNextLine()</code>	OPLTestFile.csv	True	True	
7	same as 6 but there is no next line	same as 6	False	False	

Post conditions for Test:

An election file is created.

Project Name : Project2, Voting System

Test Stage : Unit 1

Test Case ID#: 4

Team# : 7

Test Date : 12/9/2019

Name(s) of Testers:

Danny, Raymond, Raymond, Sanfer

Test Description:

This tests the Election class which runs the entire system

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

Automated : Yes **No**

Results : **Pass** Fail

Preconditions for Test:

Election hasn't run

Step#	Test Step Description	Test Data	Expected Result	Actual Result	Notes
1	call runElection	TestFiles.SMALL_OPL	correct audit file contents	correct audit file contents	

Post conditions for Test:

Election is run

Project Name : Project2, Voting System

Team# : 7

Test Stage : Unit 1

Test Date : 12/9/2019

Test Case ID#: 5

Name(s) of Testers: Danny, Raymond

Test Description: Ensure that the FileNameValidator class works correctly and can validate file names

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

FileNameValidatorTest.java

Automated : Yes No

Results : Pass Fail

Preconditions for Test: Have a file name

Step#	Test Step Description	Test Data	Expected Result	Actual Result	Notes
1	Call Validate (test data)	string "I do not exist"	False	False	
2	Call Validate (test data)	TestFiles.SMALL_CPL	True	True	
3	Call Validate (test data)	null	False	False	
4	Call Validate (test data)	""	False	False	

5	Call Validate (test data)	"\n"	False	False	
6	Call Validate (test data)	TestFiles.TESTING_DIRECTORY	False	False	

Post conditions for Test:

Know if file name is valid meaning said file exists.

Project Name : Project2, Voting System

Test Stage : Unit 1

Test Case ID#: 6

Team# : 7

Test Date : 12/9/2019

Name(s) of Testers:

Danny, Raymond, Raymond, Sanfer

Test Description:

This tests the LargestRemainderSeatAllocatorFormatter class

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

Automated : Yes No

Results : Pass Fail

Preconditions for Test:

Have a valid SeatAllocator

Step#	Test Step Description	Test Data	Expected Result	Actual Result	Notes
1	create a class object with the SeatAllocator and call toMediaFormat	CPLParitiesEqual sBallots.csv	media format	media format	
2	create a class object with the SeatAllocator and call toSummaryFormat	CPLParitiesEqual sBallots.csv	summary format	summary format	

3	create a class object with the SeatAlloactor and call toAuditFormat	CPLParitiesEqual sBallots.csv	audit format	audit format	
---	---	----------------------------------	--------------	--------------	--

Post conditions for Test:

The proper values are returned and in the formats that are specified.

Project Name : Project2, Voting System

Test Stage : Unit 1

Test Case ID#: 7

Team# : 7

Test Date : 12/9/2019

Name(s) of Testers:

Danny, Raymond, Raymond, Sanfer

Test Description:

This tests the LargestRemainderSeatAllocator class

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

Automated : Yes No

Results : Pass Fail

Preconditions for Test:

the object fields are all valid and not null

Step#	Test Step Description	Test Data	Expected Result	Actual Result	Notes
1	create allocator and test first round for CPL	mock object	results	same results	
2	same as 1 but for OPL	mock object	results	same results	
3	test second round for CPL	mock object	results	same results	

4	test second round for OPL	mock object	results	same results	
5	test placement for CPL	mock object	results	same results	
6	test placement for OPL	mock object	results	same results	

Post conditions for Test:

The object fields are returned and the returned results are equal to the actual fields.

Project Name : Project2, Voting System
Test Stage : Unit 1
Test Case ID#: 8

Team# : 7
Test Date : 12/9/2019
Name(s) of Testers:
Danny, Raymond, Raymond, Sanfer

Test Description:

This tests the SeatAllocator class for ties

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

Automated : Yes No
Results : Pass Fail

Preconditions for Test:

2 candidates are tied for votes

Step#	Test Step Description	Test Data	Expected Result	Actual Result	Notes
1	Call breakTie	CPLTeller from parser test	percentage of wins in tiebreaker < 12.5% of the time	percentage of wins in tiebreaker < 12.5% of the time	

Post conditions for Test:

tie is broken fairly

Project Name : Project1, Voting System
Test Stage : Unit 1
Test Case ID#: 9

Team# : 7
Test Date : 12/9/2019
Name(s) of Testers:
Danny, Raymond, Raymond, Sanfer

Test Description:

This tests the Summarizer class

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

Automated : Yes No
Results : Pass Fail

Preconditions for Test:

No information has been put into the summary and media files.

Step#	Test Step Description	Test Data	Expected Result	Actual Result	Notes
1	Create a summarizer class object and call summarize	TestFiles.SMALL_OPL_SUMMARY	summary formatted file	summary formatted file	Can use any other test file as well
2	same as 1	same as 1	media formatted file	media formatted file	Can use any other test file as well

Post conditions for Test:

Information from the formatter has been output to the summary and media files.

Project Name : Project2, Voting System

Test Stage : Unit 1

Test Case ID#: 10

Team# : 7

Test Date : 12/9/2019

Name(s) of Testers:

Danny, Raymond, Raymond, Sanfer

Test Description:

This tests the Teller class (OPL and CPL too) and its functions

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

Automated : Yes No

Results : Pass Fail

Preconditions for Test:

Have all of the parameters necessary to create an object

Step#	Test Step Description	Test Data	Expected Result	Actual Result	Notes
1	create CPL teller and call Teller.toString()	CPL teller from parser test	string	same string	
2	same as 1 but with OPL teller	OPL teller	string	same string	
3	CPL test tallyBallots	TestFiles.SMALL_CPL	correct tallies	correct tallies	

4	same as 3 but with OPL	OPL	correct tallies	correct tallies	
5	test getPartyTallies	same as 3	correct tallies	correct tallies	
6	same as 5 but with OPL	same as 4	correct tallies	correct tallies	
7	CPL test seatCount	same as 3	5	5	
8	same as 7 but with OPL	same as 4	5	5	
9	CPL test getPartyCandidates	same as 3	list of candidates	list of candidates	
10	same as 9 but iwth OPL	same as 4	list of candidates	list of candidates	

Post conditions for Test:

an object with the given parameters is created

Project Name : Project2, Voting System
Test Stage : Unit 1
Test Case ID#: 1

Team# : 7
Test Date : 12/9/2019
Name(s) of Testers:
Danny, Raymond, Raymond, Sanfer

Test Description:

This tests the Audit class and all of its methods

Indicate where are you storing the tests (what file) and the name of the method/functions being used. AuditTest.java,
/testing/audit.txt

Automated : Yes **No**

Results : **Pass** Fail

Preconditions for Test:

No audit file of the same name as the one given to the constructor has already been created in the same directory as the program, and the “.txt” extension is included in the file name.

Step #	Test Step Description	Test Data	Expected Result	Actual Result	Notes
1	Create an Audit file with a file name and teller and call writeTeller for CPL file	same info as TestFiles.SMALL_CPL_TELLER	TestFiles.SMALL_CPL_TELLER	TestFiles.SMALL_CPL_TELLER	

2	Create an Audit file with a file name and teller and call writeTeller for OPL file	same info as TestFiles.SMALL_OPL_TELLER	TestFiles.SMALL_OPL_TELLER	TestFiles.SMALL_OPL_TELLER	
2	call tallyBallot and then call writeBallotWithWinner	ballot = ',,,,,1,,' winner = 'Bob'	,,,,,1,, (Bob)	,,,,,1,, (Bob)	

Post conditions for Test:

Audit file is created (test.txt) with the proper data and new lines filled in.

Project Name : Project2, Voting System

Test Stage : Unit 2

Test Case ID#: 11

Team# : 7

Test Date : 12/9/2019

Name(s) of Testers:

Danny, Raymond, Raymond, Sanfer

Test Description: Testing the user input file.

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

Automated : Yes No

Results : Pass Fail

Preconditions for Test: Have a file name

Step#	Test Step Description	Test Data	Expected Result	Actual Result	Notes
1	test getFileName	a valid file name	file name	file name	
2	test getFileName	invalid file name	null	null	

Post conditions for Test:

input is received