

<p>The PBI, the Task Description (from Sprint Log) with Unique Testing Number:</p> <p>PBI 001: Bring in multiple files into the election system Task Description: Modify the Election.jav to bring in multiple files for CPL and IR election. Testing Number: W1_01</p>
<p>Team Member(s) Responsible:</p> <p>Chiemeka Nwakama</p>
<p>Inputs:</p> <ol style="list-style-type: none"> 1. Test files for CPL and IR election stored in the /testing folder 2. Multiple files input into the terminal for either CPL or IR election
<p>Tests:</p> <ol style="list-style-type: none"> 1. The selected election type is running without errors 2. The election results is display to the terminal 3. Audit file is generated
<p>Outputs:</p> <ol style="list-style-type: none"> 1. Multiples files were able to process without errors OR Error message display for invalid files name 2. Audit.txt was generated for the selected election 3. The correct results was displayed
<p>Passed or Failed</p>
<p>April 28, 2023</p>

<p>The PBI, the Task Description (from Sprint Log) with Unique Testing Number:</p> <p>PBI 002: Bring in multiple files into the IR election system Task Description: Modify the IR.java to bring in multiple files for IR election. Testing Number: W1_02</p>
<p>Team Member(s) Responsible:</p> <p>Phuong Lieu</p>
<p>Inputs:</p> <ol style="list-style-type: none"> 1. An IR object containing a single file (IRfile1.csv) 2. An IR object containing multiple (IRfile1.csv, IRfile2.csv, IRfile3.csv)
<p>Tests:</p> <p>* Please see the Previous IR_Testing.docx for details on these JUnit tests.</p> <ul style="list-style-type: none"> • testInitialData - test the initial data is used in the constructor to see if it populated the

<p>data correctly</p> <ul style="list-style-type: none"> • testpopulateData - test to see if data is populated correctly for the ballots • testgetCandidates - test that candidates are correctly populated • testgetNumCandidates - test that the total number of candidates are correctly populated • testgetNumBallots - test that the total ballots are correctly stored • testUpdateCurrentVote - test that the candidate's vote are correctly updated • testsSetElectionStatus - test that the leading candidate is correctly updated • testMakeLoser - test that the losers in the election are updated correctly • testCoinToss - test that the candidate choosing in a tie election is fair • testPoolSelect - test that the candidate choosing in a tie election is fair
<p>Outputs:</p> <ol style="list-style-type: none"> 1. JUnit tests passed 2. Multiples files were able to process without errors OR Error message display for invalid files name 3. Audit.txt was generated for the selected election 4. The correct election results is displayed
<p>Passed or Failed</p>
<p>April 28, 2023</p>

<p>The PBI, the Task Description (from Sprint Log) with Unique Testing Number:</p> <p>PBI 003: Process multiple files into the CPL election system Task Description: Process multiple files into the CPL election system Testing Number: W1_03</p>
<p>Team Member(s) Responsible:</p> <p>Sidney Nguyen</p>
<p>Inputs:</p> <ol style="list-style-type: none"> 1. A CPL object containing a single file 2. A CPL object containing three files (testCPL.csv, CPLfile1.csv, CPLfile2.csv) <p>The CPL objects were set up before the tests, and all the tests used specific aspects of the CPL objects.</p>
<p>Tests:</p> <ul style="list-style-type: none"> • JUnit tests: <ul style="list-style-type: none"> ○ testPopulateParties - tests that parties of multiple files are correctly populated. ○ testPopulateParties - tests that parties of a single file are correctly populated even with the multiple file feature. ○ testPopulateCandidates - tests that candidates of multiple files are correctly populated

- testPopulateCandidates - tests that Candidates of a single file are correctly populated even with the multiple file feature.
- testPopulateBallots - tests that all ballots from multiple files are combined together in one data structure correctly
- testPopulateBallots - tests that ballots from a single file are correctly populated in one data structure correctly even with the multiple file feature
- testgetNumParties - tests that the number of parties remains the same even with multiple files processed
- testgetNumParties - tests that the number of parties is correct with a single file even with the multiple file feature
- testgetParty - tests the actual parties are correct even with multiple files processed
- testgetParty - tests the actual parties are correct for a single file even with the multiple file feature
- testgetNumCandidates - tests the number of candidates remains the same even with multiple files processed
- testgetNumCandidates - tests the number of candidates is correct with a single file even with the multiple file feature
- testgetTotalSeats - tests the total number of seats remains the same even with multiple files processed
- testgetTotalSeats - tests the total number of seats is correct with a single file even with the multiple file feature
- testAssignBallots - tests the assignment of ballots to parties is correct and is a secondary test to populating the ballots correctly from multiple files
- testAssignBallots - tests the assignment of ballots is correct from a single file even with the multiple file feature

Outputs:

1. JUnit tests passed
2. File(s) were able to process without error
 - a. The correct results are displayed
 - b. An audit file is generated with correct information
3. A file contained an error in one or more aspect and an error message is displayed
 - a. The incorrect results may be displayed
 - b. An audit file is generated with incorrect information

Passed or Failed

April 26th, 2023

The PBI, the Task Description (from Sprint Log) with Unique Testing Number:

PBI 004: A table that shows each round of the IRV

Task Description: Modify the IR.java to add new functions to output a table display for the IR election process.

Testing Number: W1_04

Team Member(s) Responsible:
Ted Casey
Inputs:
1. An IR object containing a single file (IRfile1.csv)
Tests:
<ul style="list-style-type: none"> JUnit test: <ul style="list-style-type: none"> testTable - test to see if the table were able to produce with correct values System test: <ul style="list-style-type: none"> Check if the display is able to produced to the terminal Check if the information in the election is correct Check if election process (votes transfer) is correct
Outputs:
1. JUnit test passed 2. Multiples files were able to process without errors OR Error message display for invalid files name 3. Audit.txt was generated for the selected election 4. The correct election results is displayed 5. The table is displayed to the terminal
Passed or Failed
April 28, 2023

Incomplete

The PBI, the Task Description (from Sprint Log) with Unique Testing Number: PBI 005: View what votes are added and what votes are decreased Task Description: Modify the IR.java to track each round's vote total for each candidates in IR election process. Testing Number: W1_05
Team Member(s) Responsible:
Ted Casey
Inputs:
1. An IR object containing a single file (IRfile1.csv)
Tests:
1.
Outputs:
1.

Passed or Failed

April 28, 2023

The PBI, the Task Description (from Sprint Log) with Unique Testing Number:

PBI 006: Process multiple files into the PO election system

Task Description: Create PO.java to process multiple files in construction.

Testing Number: **W2_06**

Team Member(s) Responsible:

Chiemeka Nwakama

Inputs:

1. An PO object containing single files (POfile1.csv)
2. An PO object containing multiple files (POfile1.csv, POfile2.csv, POfile3.csv)

Tests:

- JUnit test: testTable
 - test to see if the table were able to produce with correct values
 - Check if election can bring in a single file
 - Check if election can bring in multiple files
 - Check file names corrected processed
- System test:
 - Check if election can process a single file
 - Check if election can process multiple files

Outputs:

1. JUnit test passed
2. File(s) were able to process without error
 - a. The correct results are displayed
 - b. An audit file is generated with correct information
3. A file contained an error in one or more aspect and an error message is displayed
 - a. The incorrect results may be displayed
 - b. An audit file is generated with incorrect information

Passed or Failed

April 28, 2023

Incomplete

The PBI, the Task Description (from Sprint Log) with Unique Testing Number:

PBI 007: Candidate votes are accounted for

Task Description: Modify the IR.java with new functions to accounted invalidate vote that is not ranked for at least half the candidates.

Testing Number: W2_07
Team Member(s) Responsible: Ted Casey
Inputs: 1. An IR object containing multiple files (IRtest.csv)
Tests: 1.
Outputs: 1.
Passed or Failed
April 28, 2023

Incomplete

The PBI, the Task Description (from Sprint Log) with Unique Testing Number: PBI 007: Candidate votes are accounted for Task Description: Modify the IR.java with new functions to account invalidate vote that is not ranked for at least half the candidates. Testing Number: W2_07
Team Member(s) Responsible: Ted Casey
Inputs: 1. An IR object containing multiple files (IRtest.csv)
Tests: 1.
Outputs: 1.
Passed or Failed
April 28, 2023

The PBI, the Task Description (from Sprint Log) with Unique Testing Number: PBI 008: IRV ballots to have at least half the candidates ranked for it to be valid

Task Description: Modify the IR.java to change populateData() to exhaust invalid ballots and add those to a file. Testing Number: W2_08
Team Member(s) Responsible: Ted Casey
Inputs: 1. An IR object containing multiple files (IRfile1.csv)
Tests: <ul style="list-style-type: none"> JUnit test: testExhaustIncomplete() <ul style="list-style-type: none"> Check if exhausted votes are extracted out of ballots Check if the candidates votes are updated correctly
Outputs: 1. Junit test passed
Passed or Failed
April 28, 2023

The PBI, the Task Description (from Sprint Log) with Unique Testing Number: PBI 009: A new file containing all of the invalidated votes Task Description: Produce the invalidated_dateofelection.xxx containing invalidated votes from populateData(). Testing Number: W2_09
Team Member(s) Responsible: Phuong Lieu
Inputs: 1. An IR object containing multiple files (IRfile1.csv, IRfile2.csv, IRfile3.csv)
Tests: <ul style="list-style-type: none"> System test: <ul style="list-style-type: none"> Check if the invalidated_ddateofelection.txt is produced Check if the file contain the correct information on invalid votes
Outputs: 1. File(s) were able to produced without error <ul style="list-style-type: none"> The correct results are displayed An audit file is generated with correct information
Passed or Failed

April 28, 2023

The PBI, the Task Description (from Sprint Log) with Unique Testing Number:

PBI 010: The PO election determines a single winner when there are no ties

Task Description: Modify the PO.java with new functions that allows to to declare candidate winner when there is no tie.

Testing Number: W2_10

Team Member(s) Responsible:

Chiemeka Nwakama

Inputs:

1. An PO object containing single files (POfile1.csv)
2. An PO object containing multiple files (POfile1.csv, POfile2.csv, POfile3.csv)

Tests:

- Junit tests:
 - selectWinner - test that the corect winner is selected in the election for a single file
 - selectWinner - test that the corect winner is selected in the election for multiple files

Outputs:

1. Junit tests passed

Passed or Failed

April 28, 2023

The PBI, the Task Description (from Sprint Log) with Unique Testing Number:

PBI 011: An unbiased system to break ties for PO election

Task Description: Modify the PO.java with new functions, coinToss() or poolSelect(), that allows unbiased system to to declare candidate winner when there is a tie.

Testing Number: W2_11

Team Member(s) Responsible:

Chiemeka Nwakama

Inputs:

1. An PO object containing single files (POfile1.csv)
2. An PO object containing multiple files (POfile1.csv, POfile2.csv, POfile3.csv)

Tests:

- JUnit tests:
 - testselectWinnerPool - test the fairness of pool select for candidates in a tie election with a single file
 - testselectWinnerPool - test the fairness of pool select for candidates in a tie election with multiple files
 - testselectWinnerCoinToss - test the fairness of coin toss for candidates in a tie election with a single file
 - testselectWinnerCoinToss - test the fairness of coin toss for candidates in a tie election with multiple files

Outputs:

1. JUnit tests passed

Passed or Failed

April 28, 2023

The PBI, the Task Description (from Sprint Log) with Unique Testing Number:

PBI 012: The PO election winner and stats displayed on the screen

Task Description: Modify the PO.java with new functions, display(), that allows the results of the PO election to display on the terminal.

Testing Number: **W2_12**

Team Member(s) Responsible:

Phuong Lieu

Inputs:

1. An PO object containing single files (POfile1.csv)
2. An PO object containing multiple files (POfile1.csv, POfile2.csv, POfile3.csv)

Tests:

- System test:
 - Check if the election result is displayed to the terminal
 - Check if correct election result is showing
 - Check if the vote's percentages for each candidate is showing

Outputs:

1. Election result is displayed in the terminal
2. Correct election result is showing
3. Vote's percentages is showing for each candidate

Passed or Failed

April 28, 2023

The PBI, the Task Description (from Sprint Log) with Unique Testing Number:

PBI 013: Run the entire PO election

Task Description: Running and checking that PO election is running correctly.

Testing Number: W2_13

Team Member(s) Responsible:

Chiemeka Nwakama

Inputs:

1. An PO object containing single files (POfile1.csv)
2. An PO object containing multiple files (POfile1.csv, POfile2.csv, POfile3.csv)

Tests:

- JUnit tests:
 - testPopulateCandidates - tests that candidates of multiple files are correctly populated
 - testPopulateCandidates - tests that Candidates of a single file are correctly populated even with the multiple file feature.
 - testPopulateBallots - tests that all ballots from multiple files are combined together in one data structure correctly
 - testPopulateBallots - tests that ballots from a single file are correctly populated in one data structure correctly even with the multiple file feature
 - testgetNumCandidates - tests the number of candidates remains the same even with multiple files processed
 - testgetNumCandidates 0 tests the number of candidates is correct with a single file even with the multiple file feature
 - testgetBallots - test that the number of parties is being stored correctly in a single file
 - testgetBallots - test that the number of parties is being stored correctly in multiple files
 - testAssignBallots - tests the assignment of ballots to parties is correct and is a secondary test to populating the ballots correctly from multiple files
 - testAssignBallots - tests the assignment of ballots is correct from a single file even with the multiple file feature
 - sortCandidates - test to see if ballots are assigned correctly to candidates via how many votes they have in a single file
 - sortCandidates - test to see if ballots are assigned correctly to candidates via how many votes they have in multiple files
 - testDuplicates - test to see if duplicates ballots are correctly found in a row for a single file
 - testDuplicates - test to see if duplicates ballots are correctly found in a row for multiple files
 - testGetNameandParty - test that the candidate's name and party are correctly stored for a single file
 - testGetNameandParty - test that the candidate's name and party are correctly stored for multiple files
 - testselectWinnerPool - test the fairness of pool select for candidates in a tie

<ul style="list-style-type: none"> election with a single file ○ testselectWinnerPool - test the fairness of pool select for candidates in a tie election with multiple files ○ testselectWinnerCoinToss - test the fairness of coin toss for candidates in a tie election with a single file ○ testselectWinnerCoinToss - test the fairness of coin toss for candidates in a tie election with multiple files ○ testAddVote - test the addVote() and getter for candidate's new vote for a single file ○ testAddVote - test the addVote() and getter for candidate's new vote for multiple files ○ selectWinner - test that the corect winner is selected in the election for a single file ○ selectWinner - test that the corect winner is selected in the election for multiple files
Outputs: 1. JUnit tests passed
Passed or Failed
April 28, 2023

The PBI, the Task Description (from Sprint Log) with Unique Testing Number: PBI 013: A GUI to select files Task Description: Implement the GUI window to select files for processing Testing Number: W2_13
Team Member(s) Responsible: Sidney Nguyen
Inputs: 1. An PO object containing single files (POfile1.csv) 2. An PO object containing multiple files (POfile1.csv, POfile2.csv, POfile3.csv)
Tests: <ul style="list-style-type: none"> • System test: <ul style="list-style-type: none"> ○ Check if the GUI window is showing ○ Check if the user can selected the file(s)
Outputs: 1. A pop-up (GUI) window is display to select files for processing
Passed or Failed
April 28, 2023

