Backlog

#1 Algorithms Fully Implemented
As a User
I want all algorithms that determine the results of an election (IRV, OPLV) work as described in the SRS document and SDD.
so that I can utilize the software confidently.

Acceptance Criteria:
- Tests are Written for all Algorithm paths
- Algorithm produces results for all algorithm expected path results including ties

Definition of Done:
- All IO Exception Handling is implemented
- Passes regression testing
- Meets all acceptance criteria
- In-line documentation and Javadocs have been updated.

Size: Small

Status: To-do

#2 Remove and Store Invalidated Ballots
As a user of the voting system,
I want IRV to only use valid ballots (ballots with at least half of the candidates ranked) and to store invalidated ballots into a separate audit file
so that the election is fair and only validated ballots are used

Acceptance Criteria:
- Audit file of invalidated ballot is produced
- Calculate election with validated ballots
- Invalidated ballots are only in the IRV election

Definition of Done:
- Passes Regression Testing
- Meeting all Acceptance Criteria
- In-line documentation and Javadocs have been updated.

Size: Medium

Status: To-do

#3 Election Table
As an official (of an IRV Election)
I would like to see a table showing each round of the IRV
so that I can see the number of votes that the candidate added/subtracted for each round

Acceptance Criteria:
- Calculate votes added for each candidate in each round
- Calculate votes subtracted for each candidate in each round
- Create GUI/Display of table

Definition of Done:
- Able to show feature in a demo
- Pass all tests (regression, acceptance, unit, etc)
- Approved by team
- Meets all Acceptance Criteria
- In-line documentation and Javadocs have been updated.

Size: Large

Status: To-do

 # 4 Command Line Argument Filename
As a user
I want to provide the filename as an argument in the commandline
so that I do not have to be prompted and to expedite the process of running the election

Acceptance Criteria:
- Take in the command line argument
- Save the argument as the file name
- No longer prompt the user, either in the command line or the GUI, for the file name

Definition of Done:
- Feature passes acceptance criteria
- Feature can be showed in a demo
- Approved by the team
- Passes all required tests
- In-line documentation and Javadocs have been updated.

Size: Small

Status: Done

#5 Filename Prompt
As a user
I want to only be prompted for the filename on the command line
so I can dictate the election file

Acceptance Criteria:
- OPLV & IRV is properly calculated
- Audit file can recreate the election
- election is fair
- Ensure it is thoroughly Tested - Forms of Regression Testing
- Documentation

Definition of Done:
- Functionality is able to be demo-ed
- All tests have been written and passed
- A log of potential conflicts with components in the system
- In-line documentation and Javadocs have been updated.

Size: Small

Status: Done

#6 GUI Filename prompt
As a electoral official
I want a GUI to use to prompt me for the file name instead of a text prompt
so I do not have to use a terminal

Acceptance Criteria:
- Ensure is is throughly tested
- Allow user to look for file on their local device
- GUI appears when program is run

Definition of Done:
- Passes all regression test
- Approved by team
- In-line documentation and Javadocs have been updated.
- Meets all Acceptance Criteria

Size: Large

Status: Done

#7 Handle Upwards of 100,000 Ballots
As a user
I want the program to be able to handle upwards of 100,000 ballots
so that I can run a large election or test the efficiency of the program

Acceptance Criteria:
- The program can run with an election file that contains 100,000 ballots
- The program will output the correct results
- The program still meets the efficiency requirements while running this large number of ballots
- Program will still operate and not break because of this

Definition of Done:
- All tests are passed
- Feature can be showed in a demo
- Acceptance criteria are all met
- Accepted by the team
- In-line documentation and Javadocs have been updated.

Size: Medium

Status: Done

#8 Print Summary Report
As an election official
I want the audit report and a short report containing date, type of election, candidates, the number of seats, and the winner(s) of the election
so that I can have a printable version to give to the election certification officials

Acceptance Criteria:
- Printable report
- Only contains date, type of election, candidates, the number of seats, and winner(s) of election
- Program produces both audit report and short report

Definition of Done:
- Passes all regression test
- Approved by team
- Able to show feature in demo
- Passes Acceptance Criteria

Size: Small

Status: To-do

#9 Add Votes Dynamically To Program Using GUI
As an election official using the GUI,
I want a way to add votes dynamically to the program
so that I can send the information directly to the system without the use of a file.

Acceptance Criteria:
- Run an election without a file
- A user can input details about the election that would normally be retrieved from the file (e.g. number of candidates).
- Ballots are dynamically added to the election
- Program will account for these and output the correct results
- Tests are written in JUNIT for all methods and classes created during the implementation of this feature

Definition of Done:
- Feature passes all tests
- Feature can be showed in a demo
- Accepted by the team
- Well documented in javadocs and inline
- Meets Acceptance Criteria

Size: Medium

Status: To-do

#10 Provide Immediate Results From Dynamically Input Votes
As electoral official using the GUI
I want to have immediate results after dynamically inputting votes
so that I can see the effect of the election

Acceptance Criteria:
-A user can run an election without a file
-A user can input details about the election that would normally be retrieved from the file (e.g. number of candidates).
-A user can see the results of the vote in the GUI (e.g. number of votes for a candidate)
-Tests are written in JUNIT

Definition of Done:
- Feature passes all tests
- Feature can be showed in a demo
- Accepted by the team
- Well documented in javadocs and inline
- Meets Acceptance Criteria

Size: Medium

Status: To-do

#11 Run Election Processor as a Server with Voting Machine Clients
As an election official
I want to securely collect ballot information directly from the voting machine clients
so that we can reduce the amount of work in processing election files and provide election information quickly.

Acceptance Criteria:
- A client/server network is established between voting machines and the election processor
- All previous functionality from files is still available
- Documentation regarding the communication (e.g. indicating the length of votes, maximum sizes for candidate names, etc) has been written and cross referenced.
-

Definition of Done:
- Feature passes all tests
- Feature can be shown in a demo
- Well documented in javadocs and inline
- Meets Acceptance Criteria

Size: Extra Large

Status: To-do

#12 Exception in Main Does Not Exit GUI
As a user
I want the GUI to exit with an alert when there is an exception in main
so that I know the program experienced a failure

Acceptance Criteria:
- The GUI exits when there is an exception in main
- The GUI continues to exit for other exceptions thrown
- The system will also flash an alert containing a message that the program failed
- This passes all manual and automated JUNIT tests

Definition of Done:
- The feature can be demo-ed in a presentation
- Accepted by the team
- Meets Acceptance Criteria
- Passes all tests
- Is well documented in line and in JavaDocs

Size: Small

Status: To-do

#13 Resolve Rare Failing Tests
As a Software Developer or Software Tester
I want the tests to not fail in rare cases
so that I can ensure the tests are accurate and that future dependencies can be placed on tests passing and that I can be sure that when a test fails it has a direct correlation to a bug in the code itself.

Acceptance Criteria:
- After 100,000 test runs there are no instances of a failed test.
- The removal of I/O in tests are considered by the team

Definition of Done:
- Team has agreed to the maintaining or removal of I/O in unit and system tests
- The passing of all tests in 100,000 runs of tests can be displayed in a demo

Size: Medium

Status: To-do