As an Election Official I would like to be able to run the system without any bugs.
I want every functionality to run perfectly without crashing in the middle of calculating the election file.

Acceptance Rate:

1) System should be able to run from the beginning until it finishes displaying the winner and generating both audit and summary file
2) It should be thoroughly tested
3) It should be documented with javadocs

Effort: Extra Large

As an Election Official I wanted to calculate the winner using a Largest Remainder.
I want the seat allocation to be calculated using the largest remainder calculation.

Acceptance Rate:

1) The result should be correct
2) It should not crashed in the middle of calculation
3) It should be thoroughly tested
4) It should be documented with javadocs

Effort: Big

As an Election Official I wanted to see a summary file.
I want the result of the summary file so that I can see who is the winner of the election.

Acceptance Rate:

1) The display of the result should be correct
2) It should create a new summary file
3) It should be documented with javadocs

Effort: Medium

As an Election Official I wanted to be able to choose the directory on where I would like to save the audit file.
I wanted to do it with a GUI and without entering the election file in the terminal. I wanted to use swing and awt as a java build in path.

Acceptance Rate:

1) It should be able to be save in a different directory
2) It should have all of the pertinent information about the election
3) It should be documented with javadocs

Effort: Medium

As an Election Official I wanted to be able to input the ballot information directly from the screen into the terminal and stored it into a CSV file as part of the audit file.
I wanted to do it with a GUI so it would be easier for me to look at.

Acceptance Rate:

1) It should be automated
2) It should output the correct result
3) It should be able to be run in a GUI
4) It should be documented with javadocs

Effort: Medium

As an Election Official I would like to search for a CSV file in a different directory through a GUI to select a file that is needed.
I would like the GUI to be created with JFileChooser so that it could prompt the user to pick a file.

Acceptance Rate:

1) It should be able to be search through a GUI
2) It should be thoroughly tested
3) It should be able to open a file in a different directory
4) It should be documented with javadocs

Effort: Medium

As an Election Official I would like to run more than 500,000 ballots in less than 10 minutes.
I want the system to use a good data structure like HashMap, LinkedList, and ArrayList to keep track of the ballots and assign the seats for each candidate.

Acceptance Rate:

1) It should process the correct winner
2) It should throw an error if the ballot is not correct
3) It should be documented with javadocs

Effort: Medium

As an Election Official I wanted to have a tie breaker. I want to use a SecureRandom class so that the result is completely random.

Acceptance Rate:

1) It should be pseudo random
2) It should be thoroughly tested
3) It should be able to determine a winner
4) It should be documented with javadocs

Effort: Small

As an Election Official I wanted a short report that can be printed in addition to the audit file that is being created in a new file. I wanted the system to prompt the user to name the report and create a new file with FileWriter.

Acceptance Rate:

1) The report should be correct
2) The report should have a pertinent information
3) It should be documented with javadocs

Effort: Small

As an Election Official I wanted to be able to name the audit file.
I would like to name the audit file as soon as the system stops running. So that I can keep track of which files I made for every election.

Acceptance Rate:

1) The audit file can be created by the Election Official
2) It should still have all of the pertinent information about the election
3) It should be documented with javadocs

Effort: Small