Table of Contents (in order of priority)

PBI 1: Fix droop and plurality algorithms

(1.Shana wants the droop and plurality algorithms as described in project #1.)

As a product owner,
I want the Voting System to run plurality election and STV election without major bugs,
So that I can run elections fairly.

Acceptance Criteria:

- Voting system takes user inputs
- Voting system runs plurality election if user chooses plurality election
- Voting system runs STV election if user chooses STV election
- Voting system provides user with election results
- Voting system saves an audit file
- Voting system turns off shuffle ballot function if program is run with command line argument '-t'

Done Criteria:

- All system components/classes/functions pass unit tests
- All system components/classes/functions pass integration tests
- Voting system passes system tests
- Approved by product owner
- Able to show feature in demo

Effort: Small, Medium, Large, Extra Large (estimate of effort and time)

<Bryan - I think this is Large. I know there is a lot of work, but I think that we can get it done with one or two people working on it in about two days. I think our issue before was that too many people were working on the same files and we were under a high time constraint.>
<Hailin - I think 2 days are extremely optimistic. Debugging and testing will take a lot longer time than putting down codes. Plus most of us are not going to be able to put in 10+hr days>
<Colin - Large>
<Josh - Large>

PBI 2: Deal with invalid ballots

The election officials have been told by their higher ups that when using droop, the ballots need to have at least half of the candidates ranked for the ballot to be valid. The ballots will no longer be invalidated at the point of collection but will need to be done when the election is run with the software system. The system must be able to account for all ballots that have been invalidated. A report should be run to disk that stores all of the ballots that are invalidated

2-1: Add ability to find invalid ballots

As an election official, I want STV ballots to be validated at election run time by the software, So that ballots need to have at least half of the candidates ranked for the ballot to be valid.
Acceptance Criteria: ● Only ballots with >= 50% candidates ranked are used in STV election Done Criteria: ● A function putting ballots with <50% candidates ranked in an invalid ballot list is implemented ● Passes all regression tests ● Passes testing per acceptance criteria items ● Approved by product owner ● Able to show feature in demo
Effort: Small, Medium, Large, Extra Large (estimate of effort and time) <Bryan - I think this is somewhere between small and medium. I think this would be a change to the ballot file processor and we could convert our logger object into the invalidator recorder.> <Hailin - Medium> <Colin - Medium> <Josh - Medium>

2-2: Save invalid ballots to a file

As an election official, I want invalidated STV ballots to be stored on the disk, So that all invalidated ballots can be accounted for.
Acceptance Criteria: ● Invalidated ballots are stored in a file on the disk Done Criteria: ● A feature is added to store invalidated ballots in a file ● Passes all regression tests ● Passes testing per acceptance criteria items ● Approved by product owner ● Able to show feature in demo
Effort: Small, Medium, Large, Extra Large (estimate of effort and time) <Bryan - I think this is somewhere between small and medium. I think this would be a change to the ballot file processor and we could convert our logger object into the invalidator recorder.> <Hailin - Medium> <Colin - Medium> <Josh - Medium>

PBI 3: Graphical User interface for selecting files from a directory

Users should be able to search for files on the disk and be able to select files from a directory structure. A graphical user interface is needed that is easy to use. (Josh)

As a election official, I want to have a graphical user interface to search for files on the disk and select files from a directory structure, So I can find and load ballot files into the system without knowing how to write file paths or use the command line.
Acceptance Criteria: ● When inputting ballot files, the user is presented with a GUI to select the file ● Accepts multiple ballot files ● Works on CSELabs machines Definition of Done: ● Passes all regression tests ● Passes testing per acceptance criteria items ● Approved by product owner ● Able to show feature in demo
Effort: Small, Medium, Large, Extra Large (estimate of effort and time) <Bryan - I think this is part of the item above. Large, but same as above. If we can get the one above we should be able to get this one.> <Josh - Large or medium just to implement searching for files? Unsure if there are prebuilt things we can use to search for files, maybe requires some research?> <Colin - either large or medium - There could be some tutorials available on how to do this, as I am sure selecting a file is a fairly common operateration.> <Hailin - Large>

PBI 4: Graphical user interface for user input

All user input should have a graphical user interface. (Josh)

As an election official, I want to have a graphical user interface for user input, So I know exactly where and how to select the type of election, input the number of seats, and specify ballot file(s).
Acceptance Criteria: ● The user is presented with a GUI for inputting requirements to run an election ● Accepts plurality and STV elections ● Accepts any positive integer of seats ● Accepts any valid ballot files specified ● Works on CSELabs machines Definition of Done: ● Passes all regression tests ● Passes testing per acceptance criteria items ● Approved by product owner ● Able to show feature in demo
Effort: Small, Medium, Large, Extra Large (estimate of effort and time) <Bryan - I agree that this is large. I know that Colin has done some work, but I am not sure how much would be needed to get this started.> <Colin - I agree, this is large. I have kind of a template that we could try to fit our code into, but it might be a lot of debug work to get it going, especially trying to get a makefile built up so it builds on CSE labs machines> <Hailin -Large> <Josh - Large>

PBI 5: Graphical user interface to name invalidated ballot file and save

The election officials should be able to name the invalidated file whatever they like and store it wherever they like. They would like to have a graphical user interface to name the file and save it

5-1: Give the system a file name to use to store invalid ballots

As an election official, I want to be able to name the invalidated file and specify location to store it, So that I can find the invalidated file.
Acceptance Criteria: ● The user is able to type in a file name and file path to store invalidated ballot file Done Criteria: ● Add an option to user interface to name and store invalidated ballot file ● Function passes unit test ● Passes all regression tests ● Passes testing per acceptance criteria items ● Approved by product owner ● Able to show feature in demo
Effort: Small, Medium, Large, Extra Large (estimate of effort and time) <Bryan - This is a medium. I can see how to pass the file name through the voting_info to file processor to the invalidator logger, but not sure how to add the file in other directories.> <Hailin - Medium> <Colin - Medium> <Josh - Medium>

5-2: Use graphical interface to specify invalidated ballot file names

As an election official, I want to have a graphical user interface to specify invalidated ballot file name and path, So that I can specify invalidated ballot filename and path with ease.
Acceptance Criteria: ● When specifying location and file name of invalidated ballot file, user is presented with a graphical interface (window) to input file name and select file location Done Criteria: ● Add graphical user interface option for specifying name and path of invalidated ballot file ● Function is implemented and tested ● Passes all regression tests ● Passes testing per acceptance criteria items ● Approved by product owner ● Able to show feature in demo
Effort: Small, Medium, Large, Extra Large (estimate of effort and time) <Bryan - I think anything with a graphical user interface is medium to large.> <Hailin - Medium> <Colin - Medium> <Josh - Medium>

PBI 6: Create a short election report and print

6-1: Save short election report to a file

As an election official, I want a short report with the date, type of election, candidates involved, number of seats, and winners of the election so that the election certification officials can view the results.
Acceptance Criteria: ● A file is created after the election has been run which has the date, type of election, candidates involved, the number of seats, and the winners of the election. Done Criteria: ● Function is implemented and tested ● Passes all regression tests ● Passes testing per acceptance criteria items ● Approved by product owner ● Able to show feature in demo
Effort: Small, Medium, Large, Extra Large (estimate of effort and time) <Bryan - We can discuss, but I didn't feel like this needed to be split up. To me it sounds like the summary that we print to the screen at the end of the election run just print the screen to a file. I marked medium because I am sure sure if there are other concerns I am missing.> <Josh - This seems really short to me - just redirecting the results display to and ostream object instead of std::out (or we could just have results display take an ostream object and pass it std::out when we want it to print to screen). The part I'm not sure about is actually physically(?) printing the file - can we just use any system application to print a .txt file it creates or we actually have to set up working with a printer?> <Hailin - Medium> <Colin - Medium>

6-2: Print the short election report

As an election official, I want the short election report to be printed, So that the election certification officials can view the results.
Acceptance Criteria: ● Users are presented with an option to print the election report. ● When user choose to print the election report, a report is printed on a supported device of user's choice Done Criteria: ● Function is implemented and tested ● Passes all regression tests ● Passes testing per acceptance criteria items ● Approved by product owner ● Able to show feature in demo
Effort: Small, Medium, Large, Extra Large (estimate of effort and time) <Bryan - unsure how to print.> <Josh - This seems really short to me - just redirecting the results display to and ostream object instead of std::out (or we could just have results display take an ostream object and pass it std::out when we want it to print to screen). The part I'm not sure about is actually physically(?) printing the file - can we just use any system application to print a .txt file it creates or we actually have to set up working with a printer?> <Hailin - Medium> <Colin - Medium>

PBI 7: Graphical system to cast ballots

7-1: Demo of GUI system to cast ballots

As an election official, I want a graphical user interface to cast a ballot, So that votes can be sent to the voting system directly without a ballot file.
Acceptance Criteria: ● The user is presented with a GUI for casting a ballot ● Ballot is sent to voting system to be processed Definition of Done: ● Function is implemented and tested ● Passes all regression tests ● Able to show feature in demo
Effort: Small, Medium, Large, Extra Large (estimate of effort and time) <Colin - Large> <Bryan - I agree that this is large. We need to update the ballot processor to have more items for error checking / handling and we need to add user displays and interfaces.> <Hailin - Large> <Josh - Large>

7-2: Remote casting of ballots

As an election official, I want the Voting System to work as a backend to the Vote casting GUI, So that ballots can be cast on remote machines and counted on a central server.
Acceptance Criteria: ● The voting system can add ballots in real time ● The voting system can be forced to stop accepting ballots ● An election can be run once the system is no longer accepting ballots Definition of Done: ● Function is implemented and tested ● Passes all regression tests ● Able to show feature in demo
Effort: Small, Medium, Large, Extra Large (estimate of effort and time) <Colin - Large> <Bryan - I agree that this is large. We need updated ballot processor, error handling, user GUI and methods to tell the system when we are done collecting ballots.> <Hailin - Large> <Josh - Large>