

Rank	Title	User Story	Acceptance Criteria:	Definition of Doni Effort:	PBI Authors	Tasks: Red = Not Started, Yellow = In progress, Green = Completed	Add initial to track who is assigned to Sprint	Initial at start of task indicates who is working on it A-Anna, N-Navin, J-John, K-Kyle
1	PBI 5 - OPL algorithm runs correctly	As an Election Official, I want to be able to calculate the results of an OPL election so that I know the winners of an OPL election	Acceptance Criteria: The winners of an OPL election are successfully calculated for all cases including edge cases	Definition of Doni Effort: Large	PBI Author (s) Navin	A - Task 1: Change A - Task 2: Run OPL algorithm - Task 3: Identify N - Task 4: Run algorithm A - Task 5: Add code A - Task 6: Run algorithm A - Task 7: Update documentation including bug list and test log		
2	PBI 9 - IRV algorithm correctly identifies when a candidate has reached droop	As an Election Official, I want to be able to calculate the results of an IRV election so that I can know who won the IRV election	Acceptance Criteria: The IRV algorithm runs correctly for all cases including edge cases	Definition of Doni Effort: Large	PBI Author (s) Navin	A - Task 1: Review A - Task 2: Run algorithm A - Task 3: Update Documentation including test log and buglist		
3	PBI 6 - IRV algorithm audit file is produced correctly	As an Election Official, I want to be able to view an audit file so that I can verify that the IRV election results are correct	Acceptance Criteria: The audit file accurately reflects the IRV election algorithm's calculations	Definition of Doni Effort: Medium	PBI Author (s) Navin	J - Task 1: Change J - Task 2: Run algorithm J - Task 3: Make test J - Task 4: Run algorithm J - Task 5: Update documentation including buglist and testing log		
4	PBI 7 - Multiple election files are able to be processed	As an Election Official, I want to be able to input multiple csv files so that I can calculate the election results	Acceptance Criteria: Multiple csv files are able to be processed by the system	Definition of Doni Effort: Large	PBI Author (s) Navin	N - Task 1: Be able to process N - Task 2: Create N - Task 3: Design N - Task 4: change N - Task 5: create N - Task 6: Update N - Task 7: Run system integration test		
5	PBI 8 - IRV Remove Invalid Ballots	As an Election Official, I want to be able to remove all invalid ballots so that I can verify that the IRV election results are correct	Acceptance Criteria: The IRV removes invalid ballots The algorithm ensures that at least half of the candidates are ranked	Definition of Doni Effort: Medium	PBI Author (s) Navin	K - Task 1: Design K - Task 2: write code K - Task 3: write code K - Task 4: Write code K - Task 5: write code and Task 6: update documentation		
6	PBI 4 - Reading in the input file for PO	As an Election Official, I want the program to read in the data so that the rest of the algorithm can run	Acceptance Criteria: Able to read the first line of the CSV file to determine the voting type Able to determine the list of candidates based on the second line Able to determine the number of candidates based on third line Able to determine the ballot's choice of vote based on the structure	Definition of Doni Effort: Small	PBI Author (s) Navin	N - Task 1: write code N - Task 2: write code N - Task 3: write code N - Task 4: Update documentation		
7	PBI 3 - Processing ballots for PO	As an Election Official, I want the program to be able to process ballots so that I do not have to process them manually	Acceptance Criteria: The CSV file header and body is correctly processed by voting system Handle ballots without errors.	Definition of Doni Effort: Medium	PBI Author (s) Navin	Task 1: write code Task 2: write code Task 3: write code and Task 4: Update documentation		
7	PBI 1 - PO voting type election	As an Election Official, I want to be able to run a PO voting type election so that one winner could be chosen based on the most votes	Acceptance Criteria: The candidate who received the most votes is chosen	Definition of Doni Effort: Large	PBI Author(s) Navin	Task 1: Design algorithm Task 2: Write code Task 3: run and debug Task 4: Run and debug Task 5: Update documentation		