PBI - description	Time Estimation	Tasks	Not Started	In Progress	Completed
1. Ballot Reassignment for STV : As an election official, I want ballots should be assigned in the correct order during STV elections, so that candidates are elected or eliminated based on fair ballot redistribution.	2 hrs	1. Debug STV - Implement STV class functions redistributeSurplus(), findLowestCandidate(); 2. Refactor redistributeEliminated(), runElection() of STV class; 3. Perform System Tests; 4. Add Unit Tests in stv_UT.cpp and stvballot.cpp; 5. Add Inline Documentation /Doxygen			1.Debugged STV - Implemented these functions in STV class - redistributeSurplus(), findLowestCandidate() - Anwesha; 2.Refactored the functions - redistributeEliminated() and runElection() of STV class - Anwesha; 3.System Tests with valid and invalid ballots- Anwesha;; 4.Added Unit Tests - Anwesha; 5.Documentation - Anwesha
2. One Candidate in STV: As an election official, I want the STV election algorithm to function when there is only one candidate, so that even rare election scenarios produce fair and accurate outcomes.	2 hrs	1.Debug STV → Refactored runElection() and findLowestCandidate() in STV class; 2.Complete System Tests with 1 candidate in STV CSV; 3.Add Unit Test TEST_F(STVTests, SingleCandidateTest) in stv_UT.cpp; 4.Inline Documentation /Doxygen			1.Debugged STV-> refactored the runElection() and findLowestCandidate() functions of STV class - Anwesha, 2.Completed System Tests with 1 candidate in STV csv - Anwesha 3.Unit Test added - TEST_F (STVTests, SingleCandidateTest) in stv_UT. cpp - Zoe/Anwesha, 4.Documentation - Anwesha
3. STV Results Output: As an election official, I want all candidate names to appear in STV election results, so that no candidate is omitted from the winners/losers list.	2 hrs	1.Debug Election class output functions – printToAudit(), generateResultsText() , displayResults() to fix STV result output issues; 2.log invalid ballots in display and audit file by updating stvballot error message; 3.complete System Tests; 4. Unit Tests; 5.Inline Documentation /Doxygen			1.Debugged Election class output functions - printToAudit(), generateResultsText(), displayResults() to fix issues with STV result outputs.:Anwesha, 2.Log invalid ballots while displaying results and audit file: updated the stvballot file's error message - Anwesha, 3.Completed System Tests - Anwesha, 4.Unit Tests for above Election class functions- Zoe/Anwesha, 5 .Documentation - Anwesha

4. Plurality Results Output: As an election official, I want Plurality's election results to print out every candidate's name, so that we can see each candidate and the number of votes they received.	2 hrs	1.Debug Election class output functions – printToAudit(), generateResultsText(), displayResults() for correct plurality result logging ; 2.update pluralityballot error message for invalid ballots; 3.add Unit Tests for regular and tied plurality cases; 4.System Tests; 5. Unit Tests ; 6.Inline Documentation /Doxygen			1.Debugged Election class output functions -> Anwesha : printToAudit(), generateResultsText(), displayResults() for appropriate logging and displaying plurality results. 2. updated pluralityballot for updated error message for invalid ballots entered. 3.Completed Unit Tests added: for regular ballots and tied plurality election - Annabelle, 4.System Tests, Documentation Unit Tests - Annabelle, 5.System Tests - Anwesha, 6.Doxygen Documentation - Zoe
5. Audit File Directory: As an election official, I want the audit file to be generated in the /src/ directory, so that the audit.txt file gets stored in the correct directory.	1 hr	1. Fix path compatibility in printToAudit() (Linux support); 2.complete System Tests; 3.Complete unit tests 4.Inline Documentation /Doxygen			1.Fixed path compability issues in printToAudit() function from Election class, made it Linux compatible - Zoe, 2.Completed System Tests - Anwesha, 3.Completed unit tests in Election_UT.cc: Anwesha 4.Documentation - Annabelle
6. Shuffle Functionality for STV : As an election official, I want full shuffle functionality in STV elections, so that the STV election output is fair and accurate.	2 hrs	1. Debug runElection() and setShuffle() in STV class; 2.complete System Testing (ballots before/after shuffle); 3.add Unit Test TEST_F(STVTests, BallotShufflingTest) in stv_UT.cpp; 4.complete Inline Documentation/Doxygen			1.Debugged runElection() and setShuffle() within STV class - Anwesha, 2.Completed System testing by printing ballots before and after adding shuffle - Anwesha, 3.Unit tests added - using test function TEST_F(STVTests, BallotShufflingTest) in stv_UT. cpp - Anwesha 4.Completed Documentation - Anwesha

7. Implementing MV (Municipal Voting) Algorithm: As an election official, I want to be able to run an MV election algorithm, so that multiple winners can be determined based on the candidates receiving the most ballots with a fair coin toss if there is a tie or ties between candidates.	6 hrs	1.Code MVlogic as child of Election class, add runElection(); 2.Refactor UserInterface (getInfo()) & Election (printToAudit(), generateResultsText(), displayResults(), setBallots()) for MV election logging/display 3. Add mvballot.cpp for ballot construction/validation 4.Inline Documentation /Doxygen			1.Completed coding MVlogic class as a child of Election class, add the runElection() function - Hilton, 2. Refactord UserInterface function - getInfo() and Election class functions - [printToAudit(), generateResultsText(), displayResults(), setBallots()] for appropriate logging and displaying MV election results, added mvballot.cpp for constructing MV election ballots and checking for invalid ballots - Anwesha, 3. Unit Tests for mvballot.cpp and MVlogic.cpp - Zoe, 4.System Tests - Zoe/Anwesha 5.Documentation - Hilton
8. Obtaining All Election Information from a CSV File (No User Input): As an election official, I want all pertinent election information to be obtained from a file with no user input, so that there can be no user error or possibility of fraud that could cause the election to output the incorrect people/seats.	2 hrs	Refactor getInfo() in UserInterface class & setBallots() in Election class; Refactor Unit Tests for Election class (getCSVFileName(), getNumSeats(), setBallots(), getAlgorithm()) ; System Tests for STV, MV, Plurality ; complete Inline Documentation/Doxygen			Refactored UserInterface class function getInfo() and Election class function setBallots() - Annabelle, Refactored Unit Tests for Election class :for testing getCSVFileName(), getNumSeats(), setBallots(), getAlgorithm() - Annabelle, System test for STV, MV, Plurality : Zoe / Anwesha, Documentation - Annabelle

User Story	Est.	Tasks			Completed
9. Taking Multiple CSV File Inputs: As an election official, I want to be able to bring in multiple CSV files to the system, so that we can bring in different files from different balloting locations.	4 hrs	1. Refactor UserInterface to handle multiple CSV inputs – added getCsvFileNames() 2. modify getInfo() for N inputs; 3. Add Unit Tests for getCSVFileNames, setBallots in electionunittests; 4.System tests for all election types 5.Inline Documentation/ Doxygen		4. MV System Test with multiple CSV inputs- Hilton, STV System test with multiple CSV inputs- Anwesha, Plurality System Test with multiple CSV inputs- Zoe,	1.Refactored UserInterface code to take multiple CSV inputs: added getter getCsvFileNames() to get a vector of multiple csv file names -Anwesha 2. modified getnfo() to prompt the user to take N no. of input files - Anwesha, 3.Unit Tests added in electionunittests class : getCSVFileNameTest- Annabelle, 5.Added inline Documentation - Anwesha , Completed system tests with valid, invalid and mixed ballots for MV -Anwesha 5.Doxygen -Annabelle
10. STV Ballot Validation & Reporting: As an election official, I want STV ballots with ranking errors to be removed and reported, so that only valid ballots are counted and errors are transparent.	5 hrs	1.Refactor STV and Election classes for ballot validation and logging – update generateResultsText() and setBallots() in Election, update stvballot.cpp to notify invalid ballots with IDs 2.Unit Tests, 3.System Tests, 4.complete Inline Documentation/Doxygen	2.Unit Tests, 3. System Tests, 4.Inline Documentation/Doxygen		1.Refactored STV and Election classes for Ballot validation and logging - updated functions generateResultsText() and setBallots() in Election, updated stvballot.cpp for notifying the user of invalid ballots with respective ballot IDs.- Anwesha
11. Small Election Report: As an election official. I want a small report summarizing election outcomes, so that I can maintain a record of relevant election information.	2 hrs	1.Refactor generateResultsText() in Election class to add election summaries with valid/invalid counts, vote percentages, winners and losers for MV, STV, and Plurality 2.Unit Tests, 3.System Tests, 2.complete Inline Documentation/Doxygen	2.Unit Tests, 3. System Tests, 4.Inline Documentation/Doxygen		1.Refactored Election class code generateResultsText() to add election summaries, with number of valid and invalid ballots present, percentage of votes received by each candidate and show winners and losers list for MV, STV and Plurality - Anwesha

12. Display MV Election Results: As an election official, I want MV election statistics displayed on-screen after voting, so that I can see winners, losers, and vote counts.	2 hrs	1.Refactor displayResults() and generateResultsText() in Election class to show MV output on UI 2.Unit Tests, 3.System Tests, 4.complete Inline Documentation/Doxygen	2.Unit Tests, 3. System Tests, 4.Inline Documentation/Doxygen		1.Refactored displayResult() function and generateResultsText() function of Election class to display MV election output on user interface-Anwesha
	30 hours total				