Sprint Product Backlog

Committed Backlog Items	Not Started	In progress	Complete
PBI 05.) Process Multiple CSV files Into One Election	Task 1 (Write Code): Write code to handle multiple CSV files for different election types.	Task X (Task name) -> [Who pulled task]	
PBI 05 - Process Multiple CSV files Into One Election As an election official, I want the voting system to be able to take in multiple files rather than only one. This is so that we can brigin different files from different balloting locations. Acceptance Cottaria (conditions that have to be fulfilled to ensure the story is complete): • Voting System can handle multiple CSV files to compute election • Computes correct election results Definition of Done (what is required by the issum before sending out for review): • Multiple CSV files are read into voting system • System testing passes with multiple CSV files	Task 2 (Peer Review Implemented Code): Have a peer look over the implemented code that handles multiple CSV election files.		
Code is refacioned Code is peer reviewed Documentation completed Documentation completed Consistent coding style throughout code Meets all acceptance criteria conditions Effort: Large PBI Author(s): Ryan Mower, Hoal Bul, Emma Spindler	Task 3 (System Testing) : Run system tests and ensure they pass.		
	Task 4 (Documentation): Update documentation for refactored code, buglist, and new or modified functions.		
	Task 5 (Refactor Code): Go though code and refactor it, ensuring it is readable and understandable.		
	Task 6 (Ensure Consistent Coding Style): Go though code and make sure the same coding style is implemented throughout the application.		
	Task 7 (Write Unit Tests): Implement unit tests to ensure code is working as expected.		
	Task 8 (Unit Testing) : Run Unit tests and ensure they pass.		
	Task 9 (Testing Documentation): Update the system testing and unit testing documentation inside the testing logs.		
PBI 07.0.) Invalidate IRV Ballots	Task 1 (Write Code): Write code to invalidate ballots in IRV ballots when they are not of proper ballot form.		

PBI 07.0 - Invalidate RIV Ballots As an Excitor Official, I was the MIV salida to have at least half of the candidates narked on each beliot to be considered valid, so that invalidated ballots sent used in the efection. Acceptance Criteria (conditions that have to be fulfilled to ensure the story is complete): - Candidates must be ranked so that they are rounded up from 5 or above to the nest higher integer value, so that the ballot is value. - The ballots will not be invalidated at the port of collection but will need to be done when the election is run. - Definition of One field his required by the team before sending out for review): - Ballots are deemed valid or invalid - One is reflactioned in the control of the control	Task 2 (Peer Review Implemented Code): Have a peer look over the implemented code that invalidates IRV ballots.
	Task 3 (System Testing): Run system tests and ensure they pass.
	Task 4 (Documentation): Update documentation for refactored code, buglist, and new or modified functions.
	Task 5 (Refactor Code): Go though code and refactor it, ensuring it is readable and understandable.
	Task 6 (Ensure Consistent Coding Style): Go though code and make sure the same coding style is implemented throughout the application.
	Task 7 (Write Unit Tests) : Implement unit tests to ensure code is working as expected.
	Task 8 (Unit Testing) : Run Unit tests and ensure they pass.
	Task 9 (Testing Documentation): Update the system testing and unit testing documentation inside the testing logs.
PBI 03.) Load PO Ballots PBI 03. Load PO Ballots As an Electricon Official. I want the PO electron ballots to be brought in a file so that the ballots can be counted and electricon can be	Task 1 (Write Code): Write code to load in popularity ballots into memory.
rutt. Mast be a cave file Pomust be found from the whove to be fulfilled to ensure the story is complete): Next be a cave file Pomust be formatted as: 1 to Live PO for Popularity Chriy 1 to Live PO for Popularity Chriy 2 to Live The carefulders and that party in []. The name and party are separated by commas. Min Line: Number of Baltos	Task 2 (Peer Review Implemented Code): Have a peer look over the implemented code that loads PO ballots.
Definition of Done (what is required by the isam before searing out for review): • Reads in beliefs from CSV correctly rich memory • Bas is able to be worked an once in memory • Das is able to be worked an once in memory • Code is part reviewed • Code is per review	Task 3 (System Testing) : Run system tests and ensure

Task 4 (Documentation): Update documentation for refactored code, buglist, and new or modified functions.
Task 5 (Refactor Code): Go though code and refactor it, ensuring it is readable and understandable.
Task 6 (Ensure Consistent Coding Style): Go though code and make sure the same coding style is implemented throughout the application.
Task 7 (Write Unit Tests): Implement unit tests to ensure code is working as expected.
Task 8 (Unit Testing) : Run Unit tests and ensure they pass.
Task 9 (Testing Documentation): Update the system testing and unit testing documentation inside the testing logs.

Product Backlog Items Committed to Sprint

PBI 05 - Process Multiple CSV files Into One Election

As an election official, I want the voting system to be able to take in multiple files rather than only one. This is so that we can bring in different files from different balloting locations.

Acceptance Criteria (conditions that have to be fulfilled to ensure the story is complete):

- Voting System can handle multiple CSV files to compute election
- Computes correct election results

Definition of Done (what is required by the team before sending out for review):

- Multiple CSV files are read into voting system
- System testing passes with multiple CSV files
- Code is refactored
- Code is peer reviewed
- Documentation completed
- · Consistent coding style throughout code
- Meets all acceptance criteria conditions

Effort: Large

PBI Author(s): Ryan Mower, Hoai Bui, Emma Spindler

PBI 07.0 - Invalidate IRV Ballots

As an Election Official, I want the IRV ballots to have at least half of the candidates ranked on each ballot to be considered valid, so that invalidated ballots aren't used in the election.

Acceptance Criteria (conditions that have to be fulfilled to ensure the story is complete):

 Candidates must be ranked so that they are rounded up from .5 or above to the next higher integer value, so that the ballot is valid. • The ballots will not be invalidated at the point of collection but will need to be done when the election is run.

Definition of Done (what is required by the team before sending out for review):

- Ballots are deemed valid or invalid
- Only valid Ballots used in election
- Passes unit tests
- Code is refactored
- Code is peer reviewed
- Documentation completed
- Consistent coding style throughout code
- Meets all acceptance criteria conditions

Effort: Large

PBI Author(s): Ryan Mower, Hoai Bui, Emma Spindler

PBI 03 - Load PO Ballots

As an Election Official, I want the PO election ballots to be brought in a file so that the ballots can be counted and elections can be run.

Acceptance Criteria (conditions that have to be fulfilled to ensure the story is complete):

- Must be a .csv file
- PO must be formatted as:
 - o 1st Line: PO for Popularity Only
 - o 2nd Line: Number of Candidates
 - o 3rd Line: The candidates and their party in []. The name and party are separated by commas.
 - 4th Line: Number of Ballots

Definition of Done (what is required by the team before sending out for review):

- Reads in ballots from CSV correctly into memory
- Data is able to be worked on once in memory
- Passes unit tests

- Code is refactored
- Code is peer reviewed
- Documentation completed
 Consistent coding style throughout code
 Meets all acceptance criteria conditions

Effort: Medium

PBI Author(s)- Ryan Mower, Hoai Bui, Emma Spindler