Comitted Backlog Items	Tasks		
	Not Started	In Progress	Completed
Previous project PBIs PBI001		OPL audit Tanner	
			Seed rand() Fun Caleb
		Fixing Known Bugs Caleb	
			Documentation Haneesha/Charlie
		Unit tests Haneesha/Charlie	
		Test logs Haneesha/Charlie	
		OPL Seats Bug Tanner	
Multiple files PBI003		Code mechanism for taking in multiple files Caleb	
		Code loop for reading each file per iteration Caleb	
		Code the distribution of ballots to data structure Caleb	
		Unit tests Haneesha	
		Test logs Charlie	
		Doxygen Charlie/Haneesha	
IRV invalidated ballots PBI006	Code to determine if IRV ballot is valid		
	Code to remove invalid ballots		
	Code to store invalid ballots for audit purposes		
	Unit tests		
	Test logs		
	Doxygen		
CSV file handled for PO PBI008	CSV processing module loading		
	Parsing function		
	Unit tests		
	Test logs		
PO election PBI002	Update Audit File		
	Implement method to select to run the PO election		
	Code the main algorithm		
	Output results		
	Unit testing		
	Test logs		

Previous project PBIs PBI001

As an *election official*

I want the *IRV and OPL elections to run as expected*

So that *they don't have to be performed manually, and both output the expected results so I don't worry.*

Note: We are interested in functionality and not the individual tasks for the PBI. Do not create tasks at this time.

Acceptance Criteria: *(conditions that have to be fulfilled to ensure the story is complete)*
- *There is a valid input file*
  - *Valid ballots for given election*
  - *Valid header*
- *If invalid input file, handles the process correctly*
- *Unbiased tie breaker*
- *Majority wins*
- *Saves, but ignores invalid ballots*

Definition of Done: *new code is thoroughly tested, documented, and working via system/integration testing and unit testing*

Effort: *Small, Medium, Large, Extra Large (estimate of effort and time) OR number of hours needed*

Medium | I think they run correctly, but we need to test.

PBI Author(s): Tanner, Charlie, Haneesha, Caleb

Multiple files PBI003

As an *election official*

I want *to be able to bring in multiple files for the same election*

So that *different files can be brought in from different ballot locations*

Note: We are interested in functionality and not the individual tasks for the PBI. Do not create tasks at this time.

Acceptance Criteria: *(conditions that have to be fulfilled to ensure the story is complete)*
- *Valid input files (CSV files)*
  - *Valid ballots for given election*
  - *Valid header*
- *If invalid input file, handles correctly*

Definition of Done: *(what is required by the team before sending out for review)*
- Code fully functioning and documented
- Merges all files into an input for the given election

Effort: *Small, Medium, Large, Extra Large (estimate of effort and time) OR number of hours needed*

Small

PBI Author(s): Tanner, Charlie, Caleb, Haneesha

IRV invalidated ballots PBI006

As an *election official*

I want the *IRV ballots to have at least half the candidates ranked for the valid ballots. The Ballots will not be invalidated at the point of collection but will be invalidated when the election is run.*
So that *invalidated ballots are removed from the election and a file is created that stored the invalidated ballots for audit purposes.*

Note: We are interested in functionality and not the individual tasks for the PBI. Do not create tasks at this time.

Acceptance Criteria: *(conditions that have to be fulfilled to ensure the story is complete)*
- *IRV election runs properly*
- *0.5 or above is rounded to the next higher integer value*
- *Ballots are not invalided at the point of collection*

Definition of Done: *(what is required by the team before sending out for review)*
- Code fully functioning and documented
- Output displayed to the screen is as expected

Effort: *Small, Medium, Large, Extra Large (estimate of effort and time) OR number of hours needed*

MedLarge

PBI Author(s): Tanner, Charlie, Caleb, Haneesha

CSV file handled for PO PBI008

As an *election official*

I want to *be able to process a CSV file to be run for popularity vote*

So that *I do not have to manually input the data and the popularity vote can run as expected*

Note: We are interested in functionality and not the individual tasks for the PBI. Do not create tasks at this time.

Acceptance Criteria: *(conditions that have to be fulfilled to ensure the story is complete)*
- *There is a valid input file (csv format)*
  - *Valid ballots for given election*
  - *Valid header*
- *If invalid input file, handles correctly*

Definition of Done: *(what is required by the team before sending out for review)*
- Code fully functioning
- Thoroughly tested via unit testing and system/integration testing

Effort: *Small, Medium, Large, Extra Large (estimate of effort and time) OR number of hours needed*

Small

PBI Author(s): Tanner, Charlie, Caleb, Haneesha

PO election PBI002

As an *election official*

I want *to be able to run a popularity vote with the voting software*

So that *I do not have to manually run it, and it outputs the expected results*

Note: We are interested in functionality and not the individual tasks for the PBI. Do not create tasks at this time.

Acceptance Criteria: *(conditions that have to be fulfilled to ensure the story is complete)*
- *There is a valid input file (csv format)*
  - *Valid ballots for given election*
  - *Valid header*
- *If invalid input file, handles correctly*
- *Unbiased tie breaker*
- *Majority wins*

Definition of Done: *(what is required by the team before sending out for review)*
- Code fully functioning and outputs expected results
- Thoroughly tested via unit testing and system/integration testing

Effort: *Small, Medium, Large, Extra Large (estimate of effort and time) OR number of hours needed*

Medium

PBI Author(s): Tanner, Charlie, Caleb, Haneesha