

Product Backlog:

PBI 01- Bug1
As a user, I want to compute a fair OPL election, including ties, during the course of computing the results so that bias is removed when selecting new candidates.
Acceptance Criteria (conditions that have to be fulfilled to ensure the story is complete): <ul style="list-style-type: none">• Computes fairly when fewer candidates voted for compared to the number of seats that need to be distributed• Randomly resolve winner of tie when parties tie• Randomly resolve winner of tie when candidates tie• Compute fair OPL election
Definition of Done (what is required by the team before sending out for review): <ul style="list-style-type: none">• Passes unit tests• Compiles and computes OPL election• Can determine the election winners for corresponding seats fairly
Effort: Medium
PBI Author(s)- Ryan Mower, Hoai Bui, Emma Spindler

PBI 02- Popularity Only Election
As an election official, I need to be able to run a popularity only (PO) election algorithm so that we can determine the winner if there are PO elections being held.
Acceptance Criteria (conditions that have to be fulfilled to ensure the story is complete): <ul style="list-style-type: none">• Election must be fair (ties are handled fairly)• Take in at least 10 candidates• Account for ballots correctly• Prove ballots were handled correctly
Definition of Done (what is required by the team before sending out for review): <ul style="list-style-type: none">• Passes unit tests• Passes system testing• Compiles and computes PO election• Can determine the winner for a PO election fairly
Effort: Extra Large
PBI Author(s)- Ryan Mower, Hoai Bui, Emma Spindler

PBI 03 - 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.
<p>Acceptance Criteria (conditions that have to be fulfilled to ensure the story is complete):</p> <ul style="list-style-type: none"> • Must be a .csv file • PO must be formatted as: <ul style="list-style-type: none"> ○ 1st Line: PO for Popularity Only ○ 2nd Line: Number of Candidates ○ 3rd Line: The candidates and their party in []. The name and party are separated by commas. ○ 4th Line: Number of Ballots
<p>Definition of Done (what is required by the team before sending out for review):</p> <ul style="list-style-type: none"> • Reads in ballots from CSV correctly into memory • Data is able to be worked on once in memory • Passes unit tests
Effort: Medium
PBI Author(s)- Ryan Mower, Hoai Bui, Emma Spindler

PBI 04 - PO Results Displayed
As an election official, I would like the popularity only results displayed to the screen so that they know who won and who lost the election.
<p>Acceptance Criteria (conditions that have to be fulfilled to ensure the story is complete):</p> <ul style="list-style-type: none"> • The results outputted to screen are percentages of votes each candidate received so that constituents who voted know how well their candidate did in the election • Winner is clearly displayed
<p>Definition of Done (what is required by the team before sending out for review):</p> <ul style="list-style-type: none"> • Losing candidates voting percentages clearly displayed on the screen • Winner is displayed with percentage of votes
Effort: Small
PBI Author(s): Ryan Mower, Hoai Bui, Emma Spindler

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): <ul style="list-style-type: none"> • 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): <ul style="list-style-type: none"> • Multiple CSV files are read into voting system • System testing passes with multiple CSV files
Effort: Large
PBI Author(s): Ryan Mower, Hoai Bui, Emma Spindler

PBI 06 - GUI interface
As a user, I want a GUI (Graphical User Interface) to appear instead of a text prompt when loading in the file name so that I can identify the file name to the computer in a much simpler manner.
Acceptance Criteria (conditions that have to be fulfilled to ensure the story is complete): <ul style="list-style-type: none"> • The GUI should allow the user to type out the file name • The GUI should allow the user to traverse the files on the computer with mouse or arrow keys to select the file for processing. • The GUI allows for multiple file selections or filename insertions
Definition of Done (what is required by the team before sending out for review): <ul style="list-style-type: none"> • Approved GUI visual inspection by development team • Approved GUI functionality inspection by development team
Effort: Extra Large
PBI Author(s): Ryan Mower, Hoai Bui, Emma Spindler

PBI 07.0 - Invalidate 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.
<p>Acceptance Criteria (conditions that have to be fulfilled to ensure the story is complete):</p> <ul style="list-style-type: none"> • 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.
<p>Definition of Done (what is required by the team before sending out for review):</p> <ul style="list-style-type: none"> • Ballots are deemed valid or invalid • Only valid Ballots used in election • Passes unit tests
Effort: Large
PBI Author(s): Ryan Mower, Hoai Bui, Emma Spindler

PBI 07.1 - Invalidated Ballot File
As an Election Official, I want the IRV invalidated ballots to be stored in a file for auditing purposes.
<p>Acceptance Criteria (conditions that have to be fulfilled to ensure the story is complete)</p> <ul style="list-style-type: none"> • A file needs to be created that stores the invalidated ballots for audit purposes. • The name of the file should be invalidated_dateofelection.xxx
<p>Definition of Done (what is required by the team before sending out for review):</p> <ul style="list-style-type: none"> • Passes system tests • Invalidated_dateofelection.xxx file created with invalidated ballots
Effort: Medium
PBI Author(s): Ryan Mower, Hoai Bui, Emma Spindler

PBI 08 - Election Summary Table
As an official, I want there to be a table at the end of the election showing each round of the IRV and the number of votes that the candidate added/subtracted for that round so that we can review how the winner was determined.
Acceptance Criteria (conditions that have to be fulfilled to ensure the story is complete): <ul style="list-style-type: none"> • A table is produced which shows the change of votes for each candidate in each round • The table is printed to the terminal • The table is saved in a file called election_table(\$DATE_AND_TIME).xxx
Definition of Done (what is required by the team before sending out for review): <ul style="list-style-type: none"> • Passes unit tests • Format looks good on visual inspection • Table contains all voting information about each round
Effort: Large
PBI Author(s): Eric Palmer

PBI 09 - Additional Short Report
As Election Official, I want a summary report so that it can be given to election officials in the case of election contesting.
Acceptance Criteria (conditions that have to be fulfilled to ensure the story is complete): <ul style="list-style-type: none"> • Report is produced that can be printed • The report should have the <ul style="list-style-type: none"> ◦ Date ◦ Type of election ◦ Candidates ◦ Number of seats ◦ Winner(s) of the election
Definition of Done (what is required by the team before sending out for review): <ul style="list-style-type: none"> • Report contains all of the needed information • Report is generated correctly regardless of election outcome • Report passes visual inspection of correctness • Passes system test
Effort: Large
PBI Author(s): Eric Palmer