

Fix Bug in Current CPL Election Code

| | |
|---------------------|--|
| UserStory | As an election official, I need the existing CPL (Closed Party List) election code to function correctly without errors, so that the voting process is accurate and reliable, reflecting the true intentions of voters. |
| Acceptance Criteria | Identify and resolve the specific bug(s) causing issues in the current CPL election code. Ensure that the fixed code accurately processes ballots according to the CPL format and rules without failing. Validate the compatibility and stability of the updated code in handling the data formats currently used in elections. Provide a clear report detailing what the bug was, how it was fixed, and the testing carried out to ensure the issue has been resolved. Update the system documentation to include any changes made to the code and any new procedures required to avoid similar issues in the future. |
| Definition of Done | The bug is clearly identified, and a detailed fix is implemented. Unit tests are updated or created to cover the bug scenario and pass successfully. Integration tests confirm that the system as a whole works seamlessly with the fixed code. Code review has been conducted to ensure the fix meets quality standards and does not introduce new issues. Documentation is updated to reflect the changes made. |
| Effort | Median |
| PBI Author | Naiqi Jiang |

CPL multiple files

| | |
|-----------|---|
| UserStory | As an election official, I would like the system to support the importing of multiple files for the CPL (Closed Party List) voting system, enabling us to bring in different files from various balloting locations to facilitate a streamlined voting process. |
|-----------|---|

| | |
|---------------------|--|
| Acceptance Criteria | <p>The system must allow the selection and simultaneous importing of multiple files specifically formatted for the CPL voting system.</p> <p>It must ensure the compatibility and proper integration of data from these files into the CPL voting process, processing all ballots without fail.</p> <p>The system must validate the integrity and format of the imported files to ensure accurate CPL voting results.</p> <p>A summary of the import process, including any issues encountered with specific files, should be displayed to the user, ensuring transparency and traceability.</p> |
| Definition of Done | <p>Unit and integration tests specifically designed for the CPL multiple file import feature pass successfully.</p> <p>Documentation has been updated to include instructions for preparing and importing multiple files into the CPL voting system.</p> <p>The feature has been validated through user acceptance testing by election officials, confirming its effectiveness and reliability.</p> |
| Effort | Large |
| PBI Author | Naiqi Jiang |

OPL multiple files

| | |
|---------------------|---|
| UserStory | <p>As an election official,</p> <p>I need the system to facilitate the importing of multiple files for the OPL (Open Party List) voting system,</p> <p>allowing for the inclusion of various files from different balloting locations to enhance the OPL voting process's comprehensiveness and accuracy.</p> |
| Acceptance Criteria | <p>The system must support the selection and simultaneous importing of multiple files, tailored for the OPL voting system. It should guarantee the seamless integration of data from these files into the OPL voting process, ensuring no ballot is overlooked.</p> <p>The system is required to check the integrity and format of the imported files for accurate OPL results.</p> <p>Users should be provided with a detailed summary of the import process, highlighting any encountered issues, to maintain</p> |

| | |
|--------------------|--|
| | transparency and accountability. |
| Definition of Done | Specific unit and integration tests for the OPL multiple file import feature are conducted and passed successfully. Updated documentation includes comprehensive guidelines on importing multiple files for the OPL system. Election officials have conducted and approved user acceptance testing for the feature, demonstrating its functionality and reliability. |
| Effort | Large |
| PBI Author | Naiqi Jiang |

MPO files(in general)

| | |
|---------------------|--|
| UserStory | As an election official, I need the system to have the capability to import MPO (Multiple Popularity Only) election ballots from a .csv file. |
| Acceptance Criteria | The system must allow the importation of ballot data from a .csv file format. The import function should be able to handle .csv files containing the names of candidates and the corresponding votes they have received. The system must validate the format and integrity of the .csv file to prevent errors during the election process. Upon successful import, the system should provide a summary of the imported data, including the total number of votes and any discrepancies found. |
| Definition of Done | Development of the .csv file import functionality is completed. The functionality has been tested for accuracy and reliability, passing all relevant unit and integration tests. Documentation for how to import ballot data from a .csv file has been updated, including guidelines for preparing the .csv file. The feature has been reviewed and accepted by election officials after successful demonstration and testing. |
| Effort | Median |
| PBI Author | Ruirui Xu |

MPO multiple files

| | |
|---------------------|---|
| UserStory | As an election official, I need the system to facilitate the importing of multiple files for the MPO (Multiple Popularity Only) voting system, allowing the inclusion of various files from different balloting locations to efficiently process and tally votes from a broader geographic area. |
| Acceptance Criteria | The system must support the selection and simultaneous importing of multiple files designed for the MPO voting system. It should ensure seamless integration of data from these files into the MPO voting process, guaranteeing that every ballot is processed. The system must validate the integrity and format of the imported files to ensure the accuracy of the MPO election results. A summary of the import process, including details of any issues encountered with specific files, should be provided to the user to maintain transparency and facilitate troubleshooting. |
| Definition of Done | Development of the multiple file import functionality for the MPO system is completed and integrated into the overall voting system. The feature passes all relevant unit and integration tests, affirming its functionality and reliability. Documentation is updated to include instructions for importing multiple files into the MPO system, along with guidelines for file format and preparation. The feature has undergone user acceptance testing by election officials, ensuring it meets the operational needs and standards. |
| Effort | Large |
| PBI Author | Ruirui Xu |

MPO election

| | |
|---------------------|--|
| UserStory | As the election official, I would like to run a MPO. So that I can get multiple winners from those candidates that receive the most ballots. |
| Acceptance Criteria | The system can determine the total number of winners needed |

| | |
|--------------------|---|
| | <p>based on the specified number of seats.</p> <p>In the event of a tie, the system must decide the winner(s) through a fair coin toss process.(fair and unbiased)</p> |
| Definition of Done | <p>Code has been completed and passes all unit tests.</p> <p>The system has passed functional testing, including vote counting, tie resolution, and seat allocation.</p> <p>Documentation has been completed, including user guides and system maintenance guides.</p> <p>Passed acceptance testing by the end-user (election officials).</p> |
| Effort | Median |
| PBI Author | Ruirui Xu |

MV files(in general)

| | |
|---------------------|---|
| UserStory | <p>As an election official,</p> <p>I need the system to support the importation of Municipal Voting (MV) election ballots from a .csv file.</p> |
| Acceptance Criteria | <p>The system must support the importation of MV ballot data from a .csv file format.</p> <p>The import function should successfully process .csv files containing the voters' choices, ensuring the data matches the expected format of candidate names and vote counts.</p> <p>The system must validate the integrity and format of the .csv file to ensure the data is accurately reflected in the voting results without errors.</p> <p>Upon successful import, the system should verify and summarize the ballot data, including a count of votes for each candidate and any discrepancies detected.</p> |
| Definition of Done | <p>Development of the .csv file import functionality for MV ballots is complete.</p> <p>The functionality is verified through comprehensive testing, including unit and integration tests, to ensure its accuracy and reliability.</p> <p>Documentation has been updated to include detailed instructions for preparing and importing .csv files for MV elections.</p> <p>The feature has undergone user acceptance testing by election officials, confirming its effectiveness and reliability in importing ballot data.</p> |

| | |
|------------|------------|
| Effort | Median |
| PBI Author | Jiahao Sun |

MV multiple files

| | |
|---------------------|---|
| UserStory | As an election official, I require the system to support the importing of multiple files for the MV (Municipal Voting) system, enabling the efficient management and processing of ballots from various polling locations. |
| Acceptance Criteria | The system must allow the selection and importing of multiple files simultaneously, catering specifically to the MV voting format. The system should ensure seamless integration of ballot data from these files into the MV voting process, accurately processing every ballot. It must validate the integrity and format of the imported files to prevent any discrepancies in the MV election results. The system should provide election officials with a detailed summary of the import process, highlighting any issues with specific files to promote transparency and ease of troubleshooting. |
| Definition of Done | The development and integration of the multiple file import functionality specific to the MV system are completed. The feature has passed all necessary unit and integration tests, proving its functionality and reliability. Documentation has been updated to include step-by-step instructions for importing multiple files for the MV system, along with file preparation guidelines. Election officials have conducted user acceptance testing of the feature, ensuring it meets the operational requirements and enhances the MV voting process. |
| Effort | Large |
| PBI Author | Jiahao Sun |

MV election

| | |
|---------------------|--|
| UserStory | As an election official, I need an algorithm, termed Municipal Voting (MV), that allows determining multiple winners based on the number of ballots received by each candidate. |
| Acceptance Criteria | The system must accurately count votes per candidate and identify if a tie occurs. In the event of a tie, the system performs a fair coin toss to determine the winner among the tied candidates. The system allows voters to select from 1 up to the total number of candidates. |
| Definition of Done | Unit and integration tests for this feature pass successfully. Documentation is updated to include operation instructions for the MV algorithm and guidelines on interpreting the results. Election officials have conducted acceptance testing and approved the functionality. |
| Effort | Large |
| PBI Author | Jiahao Sun |

MPO stats to screen

| | |
|---------------------|---|
| UserStory | As an election official, I want the MPO (Multiple Popularity Only) voting process to display detailed election stats at the end of the election, including the percentage of votes each candidate received. |
| Acceptance Criteria | At the end of the MPO voting process, the system displays the election results on the screen. The results include the percentage of votes each candidate received. The results clearly indicate which candidates won seats and which did not. The display of results is designed to be easily understandable by election officials and the public. |
| Definition of Done | All unit and integration tests pass. Documentation for how to interpret the displayed stats is completed. The feature has been reviewed and accepted by election officials. |

| | |
|------------|-----------|
| Effort | Median |
| PBI Author | Ruirui Xu |

MV stats to screen

| | |
|---------------------|--|
| UserStory | As an election official, I require the system to display Municipal Voting (MV) stats at the end of the election, showing clear information on who won and who lost, and detailing the number of votes received by all candidates. |
| Acceptance Criteria | The information should be presented in an easily understandable format, suitable for immediate interpretation by election officials and the public. The system should ensure the accuracy and integrity of the displayed stats, reflecting the true outcome of the election. |
| Definition of Done | The feature passes all relevant unit and integration tests, confirming its functionality and reliability. Documentation is updated to include details on how to use and interpret the end-of-election stats display feature. The feature has been tested and accepted by election officials, confirming its effectiveness in providing clear and accurate election outcomes. |
| Effort | median |
| PBI Author | Naiqi Jiang |