Description	As a user of the voting system I want the IRV and OPL voting systems implemented in the first stage to work correctly So that I can effectively run elections according to the specifications of the first stage
Acceptance Criteria	- OPL and IRV programs appropriately calculate election winner(s). - The generated audit files can be used to recreate the election - The election is fair to all candidates.
Definition of Done	- Passes all written tests per acceptance criteria. - Documentation for the entire voting system is complete. - Audit files are generated in the correct directory. - Ability to demo the feature.
Effort	Small

Description	As a user of the voting system I want ballots with less than half of the candidates ranked to be discarded from any IRV election result calculations and be saved in a separate file named *invalidated_dateofelection.xxx* (where <dateofelection> is the date that the election was run) So that I can view which ballots were invalidated and can run elections without having the responsibility of removing invalid ballots myself and instead have the program resolve any issues to accurately determine the winner
Acceptance Criteria	- IRV program generates results for ballot files that contain ballots with less than half of the candidates ranked. - The generated audit files can be used to recreate the election. - invalidated_dateofelection.xxx file is appropriately generated with invalidated ballots
Definition of Done	- Passes all new written tests involving ballot files with invalid ballots per acceptance criteria. - Documentation for the updated IRV program and any pertinent program files is complete. - Audit files are generated in the correct directory and contain information regarding invalidated ballot. - invalidated_dateofelection.xxx file is generated in the current

	directory with correct invalidated ballots. - Passes all regression tests. - Ability to demo the feature.
Effort	Medium

Description	As a user of the voting system I want the progression of the votes for IRV elections to be displayed in a neatly formatted table that's easy to understand So that I can analyze the progression of the election and how votes were distributed throughout each election round until a winner emerged
Acceptance Criteria	- IRV appropriately calculates the winner and generates results. - The table is displayed to the screen with information neatly and correctly formatted so it's easy to follow the distribution of votes throughout the election.
Definition of Done	- Passes all new written tests per acceptance criteria. - Documentation for the updated IRV program and any supplementary program files is complete. - Outputted table is formatted correctly with accurate values in each field and adheres to the specifications of the example provided in the project write-up. - All table fields are populated when appropriate and contain accurate information. - Passes all regression tests. - Ability to demo the feature.
Effort	Large

Description	As a user of the voting system I want the option to either not provide any command line arguments to the voting system or provide one argument that is the name of the ballot file to be considered So that I can provide the name of the ballot file when I'm sure which one it is or can be prompted for the name and have the opportunity to

	search for the correct ballot file that's needed for the election to be run
Acceptance Criteria	- OPL and IRV programs generate results when ballot file filename is provided as a command line argument. - User is prompted for the ballot file if a command line argument isn't provided. - OPL and IRV programs generate results when ballot file filename isn't provided as a command line argument.
Definition of Done	- Passes all new written tests for both scenarios (command line argument is and isn't provided) per acceptance criteria. - Documentation for the updated program files is completed - Accurate results and correctly placed audit files are outputted for the correct ballot file. - Passes all regression tests. - Ability to demo the feature.
Effort	Small

Description	As a user of the voting system I want the voting system to be able to generate election results without prompting for any additional information besides the ballot filename (when indicated) and instead be able to obtain election information from the ballot file's header So that I don't have to bear the responsibility of having to provide minute details such as the number of candidates, the type of election, number of ballot files, etc. and can instead rely on the program to infer such information to generate correct results.
Acceptance Criteria	- OPL and IRV programs appropriately calculate election winner(s) and generate results without prompting the user for any information besides, possibly, the ballot filename.
Definition of Done	- Passes all new written tests for handling ballot header information per acceptance criteria. - Documentation for the updated program files is complete. - Audit files are generated in the correct directory. - Passes all regression tests. - Ability to demo the feature.
Effort	Small

Description	As a user of the voting system I want to be prompted for the ballot file when indicated (i.e. no command line argument) through an intuitive, presentable graphical user interface So that I can either type in the ballot filename in the window prompt or search for the ballot file in the current directory in case I don't remember the complete name of the ballot file
Acceptance Criteria	- OPL and IRV programs prompt me for the ballot file, when I don't provide the filename as a command line argument, through a GUI that let's me enter the ballot filename and/or search for the file within the current directory. - The election results are appropriately generated after I accepting user input.
Definition of Done	- Passes all new written tests for the GUI per acceptance criteria. - Documentation for the updated program files is complete. - Audit files are generated in the correct directory where the ballot file is also located. - Passes all regression tests. - Ability to demo the feature.
Effort	Large

Description	As an election official I want the voting system to be able to handle up to 100,000 ballots and return the results in a timely manner So that I can use this system for larger population and don't need to wait for too long.
Acceptance Criteria	- OPL and IRV programs appropriately calculate election winner(s) - The generated audit files can be used to recreate the election - The election is fair to all candidates
Definition of Done	- Passes all new written tests with up to 100,000 ballots per acceptance criteria. - Documentation for the updated program files is complete. - Accurate results and correctly placed audit files are outputted for the correct ballot file.

	- Passes all regression tests. - Ability to demo the feature.
Effort	Small

Description	As an election official I want the voting system to be able to print a short report including the date, type of election, the candidates, the number of seats and the winner(s) So that I can store and view these relevant information without finding them out in the audit file
Acceptance Criteria	- OPL and IRV programs appropriately calculate election winner(s), generate correct results and place the audit file in the correct place - A short report containing the date, type of election, the candidates, the number of seats and the winner(s) is printed to the election official.
Definition of Done	- Passes all new written tests for printing the short report per acceptance criteria. - Documentation for the updated program files is complete. - Audit files are generated in the correct directory. - Short report is printed with correct information. - Passes all regression tests. - Ability to demo the feature.
Effort	Medium

Description	As an election official, I want voters to be able to input their ballots on the screen directly into the program across multiple machines and sent directly to the system so that I can receive real-time voting updates. Configuration of the election as well as receipt of real-time updates should be through a neat user interface.
Acceptance Criteria	- Voters can input their ballots from a different machine than the system is running on. - The system can receive ballots from multiple machines in real-time - Real-time updates of the election should be viewable by users

	from a neat user interface - Election configuration can be entered through a neat user interface or provided through a file.
Definition of Done	- The new written tests per acceptance criteria pass - Passes all regression tests - Documentation regarding this new feature is added and old documentation is updated as needed. - Ability to demo feature using a number of machines
Effort	Extra Large