SELECTED BACKLOG ITEMS	DEV NAME	TASKS NOT STARTED	TASKS IN PROGRESS	TASKS COMPLETED
	Amelia Lunning	Fix bug where running STV with 1-2 seats results in an OutOfMemory error		
As an election official I want to be able to process an STV-style election correctly	Amelia Lunning Amelia Lunning	Add additional unit and system tests for STV		
So that when the Secretary of State announces an election of that type, winners can be determined	Amelia Lunning	Adjust unit and system tests for Main accordingly		
Acceptance Criteria:	Baanee Singh	Adjust documentation as necessary		
- STV can handle any number of seats and candidates where the number of seats is less	Baarice Origin	Adjust documentation as necessary		
than or equal to the number of candidates, but greater than zero. - Ballot shuffling is toggleable and functions correctly STV conversely reproduces the migracy and logger of the election				
- STV accurately produces the winners and losers of the election				
Definition of Done: - Code refactoring				
- Exceptions are handled gracefully				
In-line documentation is updated Unit and integration tests are written and pass				
Effort:				
- Small				
- 2-3 Hours				
PBI Author(s): - Baanee Singh, Josh Subhan				
- Damice Singil, 90sti Suomai				
	Danna Cirah	Oneste a service different illinormore illinos forme the bandons		
As an election official I want to be able to run elections without having to manually input any parameters into the	Baanee Singh Baanee Singh	Create a new method that will parse all info from the headers. Documentation in-line		
system except for the election csv file So that erroneous or fraudulent information is not given, and elections are secure.				
	Baanee Singh	Create new unit and system tests (Scanner object as a parameter)		
Acceptance Criteria: - System should successfully parse the following from the file:	Josh Subhan	Implement new main() logic		
System should successfully parse the following from the file: Election type Number of seats	Baanee Singh	Documentation javadocs		
- Number of candidates	Josh Subhan	Refactoring Main		
- Number of ballots - Candidate names				
- CSV file cannot be modified				
- CSV file cannot undergo any preprocessing				
Definition of Done: - Documentation (JavaDocs, In-line comments)				
- Code refactoring				
Unit and integration tests are written and pass All exceptions are handled gracefully				
Effort:				
Madian				
- Medium - 3-4 hours				
PBI Author(s): Josh Subhan				
	Zach Larsen	Create MV class		
	Zach Larsen Josh Subhan	Create MV class Create MV testing class		
As an election official	Josh Subhan	Create MV testing class		
I want to be able to process ballots that are designed for a municipal voting style election	Josh Subhan Zach Larsen	Create MV testing class Code MV constructor, setDeterministic, startElection methods		
As an election official I want to be able to process ballots that are designed for a municipal voting style election So that when the Secretary of State announces an election of that type, ballots can be appropriately counted and winners determined	Josh Subhan Zach Larsen Zach Larsen Zach Larsen	Create MV testing class Code MV constructor, setDeterministic, startElection methods Code tabulateVotes() method Code determine_Winner_Loser() method		
I want to be able to process ballots that are designed for a municipal voting style election So that when the Secretary of State announces an election of that type, ballots can be appropriately counted and winners determined	Josh Subhan Zach Larsen Zach Larsen Zach Larsen Zach Larsen Zach Larsern	Create MV testing class Code MV constructor, setDeterministic, startElection methods Code tabulateVotes() method Code determine_Winner_Loser() method Code settleTies() method		
I want to be able to process hallots that are designed for a municipal voting style election So that when the Secretary of State announces an election of that type, ballots can be appropriately counted and winners determined Acceptance Criteria: - Election winners are properly determined	Josh Subhan Zach Larsen Zach Larsen Zach Larsen Zach Larsen Amelia Lunning	Create MV testing class Code MV constructor, setDeterministic, startElection methods Code tabulateVotes() method Code determine_Winner_Loser() method Code settleTies() method Unit test tabulateVotes()		
I want to be able to process hallots that are designed for a municipal voting style election So that when the Secretary of State amounces an election of that type, ballots can be appropriately counted and winners determined Acceptance Criefrais Election winners are properly determined Accupated youth ballots where a voter has only voted for one candidate	Josh Subhan Zach Larsen Zach Larsen Zach Larsen Zach Larsern Amelia Lunning Amelia Lunning	Create MV testing class Code MV constructor, setDeterministic, startElection methods Code tabulateVotes() method Code determine_Winner_Loser() method Code settleTies() method Unit test tabulateVotes() Unit test determine_Winner_Loser()		
I want to be able to process hallots that are designed for a municipal voting style election So that when the Secretary of State announces an election of that type, ballots can be appropriately counted and winners determined Acceptance Criteria: - Election winners are properly determined - Election winners are properly determined - Accurately count hallots where a voter has voted for more than one candidate - Accurately count hallots where a voter has voted for more than one candidate - Accurately count hallots were a voter has voted for more than one candidate	Josh Subhan Zach Larsen Zach Larsen Zach Larsen Zach Larsen Anelia Lunning Amelia Lunning Baanee Singh	Create MV testing class Code MV constructor, setDeterministic, startElection methods Code tabulateVotes() method Code determine_Winner_Loser() method Code settleTies() method Unit test tabulateVotes() Unit test determine_Winner_Loser() Unit test settleTies()		
I want to be able to process hallots that are designed for a municipal voting style election So that when the Secretary of State announces an election of that type, ballots can be appropriately counted and winners determined Acceptance Criteria: Election winners are properly determined Accurately count ballots where a voter has only voted for one candidate Accurately count ballots where a voter has voted for more than one candidate Accurately count ballots were a voter has voted for all candidates Settle at the between candidates with a fire con toss mechanism	Josh Subhan Zach Larsen Zach Larsen Zach Larsen Zach Larsen Zach Larsen Amelia Lunning Amelia Lunning Baanee Singh Baanee Singh	Create MV testing class Code MV constructor, setDeterministic, startElection methods Code tabulateVotes() method Code determine_Winner_Loser() method Code settleTies() method Unit test tabulateVotes() Unit test determine_Winner_Loser() Unit test settleTies() Create JavaDocs for MV class		
I want to be able to process hallots that are designed for a municipal voting style election So that when the Secretary of State amounces an election of that type, ballots can be appropriately counted and winners determined Acceptance Criteria: - Election winners are properly determined - Accurately count hallots where a voter has only voted for one candidate - Accurately count hallots where a voter has voted for more than one candidate - Accurately count hallots were a voter has voted for all candidates - Settle a tie between candidates with a fair coin toss mechanism Definition of Done:	Josh Subhan Zach Larsen Zach Larsen Zach Larsen Zach Larsern Amelia Lunning Amelia Lunning Baanee Singh Baanee Singh Baanee Singh	Create MV testing class Code MV constructor, setDeterministic, startElection methods Code tabulateVotes() method Code determine_Winner_Loser() method Code settleTies() method Unit test tabulateVotes() Unit test determine_Winner_Loser() Unit test settleTies() Create_JavaDocs for MV class Create_JavaDocs for MV testing class		
I want to be able to process hallots that are designed for a municipal voting style election So that when the Secretary of State announces an election of that type, ballots can be appropriately counted and winners determined Acceptance Criteria: Election winners are properly determined Accurately count ballots where a voter has only voted for one candidate Accurately count ballots where a voter has voted for more than one candidate Accurately count ballots were a voter has voted for all candidates Settle a to between candidates with a fair coin toss mechanism Definition of Done: Decemberation (JavaDocs, In-line comments).	Josh Subhan Zach Larsen Zach Larsen Zach Larsen Zach Larsen Zach Larsen Amelia Lunning Amelia Lunning Baanee Singh Baanee Singh Baanee Singh Baanee Singh	Create MV testing class Code MV constructor, setDeterministic, startElection methods Code tabulateVotes() method Code determine_Winner_Loser() method Code settleTies() method Unit test tabulateVotes() Unit test determine_Winner_Loser() Unit test settleTies() Create JavaDocs for MV class Create JavaDocs for MV testing class Create in-line comments for MV class		
I want to be able to process hallots that are designed for a municipal voting style election So that when the Secretary of State amounces an election of that type, ballots can be appropriately counted and winners determined Acceptance Criteria: - Election winners are properly determined - Election winners are properly determined - Accurately count hallots where a voter has voted for more than one candidate - Accurately count hallots where a voter has voted for more than one candidate - Accurately count hallots were a voter has voted for more than one candidate - Accurately count hallots were a voter has voted for more than one candidate - Comment of the control of	Josh Subhan Zach Larsen Zach Larsen Zach Larsen Zach Larsen Amelia Lunning Amelia Lunning Baanee Singh Baanee Singh Baanee Singh Josh Subhan	Create MV testing class Code MV constructor, setDeterministic, startElection methods Code tabulateVotes() method Code determine_Winner_Loser() method Code settleTies() method Unit test tabulateVotes() Unit test determine_Winner_Loser() Unit test settleTies() Create JavaDocs for MV class Create Jinie comments for MV class Create system tests for MV class Create system tests for MV class		
I want to be able to process hallots that are designed for a municipal voting style election So that when the Secretary of State announces an election of that type, ballots can be appropriately counted and winners determined Acceptance Criteria: Election winners are properly determined Accurately count ballots where a voter has only voted for one candidate Accurately count ballots where a voter has voted for more than one candidate Accurately count ballots where a voter has voted for all candidates Settle a in the between candidates with a fair coin toss mechanism Definition of Done: Documentation (JavaDocs, In-line comments). Unit and integration tests are written and pass All exceptions are handled gracefully Effort:	Josh Subhan Zach Larsen Zach Larsen Zach Larsen Zach Larsen Zach Larsen Amelia Lunning Amelia Lunning Baanee Singh Baanee Singh Baanee Singh Baanee Singh	Create MV testing class Code MV constructor, setDeterministic, startElection methods Code tabulateVotes() method Code determine_Winner_Loser() method Code settleTies() method Unit test tabulateVotes() Unit test determine_Winner_Loser() Unit test settleTies() Create JavaDocs for MV class Create JavaDocs for MV testing class Create in-line comments for MV class		
I want to be able to process hallots that are designed for a municipal voting style election So that when the Secretary of State amounces an election of that type, ballots can be appropriately counted and winners determined Acceptance Criteria: - Election winners are properly determined - Election winners are properly determined - Accurately count hallots where a voter has voted for more than one candidate - Accurately count hallots where a voter has voted for more than one candidate - Accurately count hallots were a voter has voted for more than one candidate - Accurately count hallots were a voter has voted for more than one candidate - Comment of the control of	Josh Subhan Zach Larsen Zach Larsen Zach Larsen Zach Larsen Amelia Lunning Amelia Lunning Baanee Singh Baanee Singh Baanee Singh Josh Subhan	Create MV testing class Code MV constructor, setDeterministic, startElection methods Code tabulateVotes() method Code determine_Winner_Loser() method Code settleTies() method Unit test tabulateVotes() Unit test determine_Winner_Loser() Unit test settleTies() Create JavaDocs for MV class Create Jinie comments for MV class Create system tests for MV class Create system tests for MV class		
I want to be able to process hallots that are designed for a municipal voting style election So that when the Secretary of State announces an election of that type, ballots can be appropriately counted and winners determined Acceptance Criteria: Election winners are properly determined Accurately count ballots where a voter has only voted for one candidate Accurately count ballots where a voter has voted for more than one candidate Accurately count ballots where a voter has voted for all candidates Settle a in the between candidates with a fair coin toss mechanism Definition of Done: Documentation (JavaDocs, In-line comments). Unit and integration tests are written and pass All exceptions are handled gracefully Effort:	Josh Subhan Zach Larsen Zach Larsen Zach Larsen Zach Larsen Amelia Lunning Amelia Lunning Baanee Singh Baanee Singh Baanee Singh Josh Subhan	Create MV testing class Code MV constructor, setDeterministic, startElection methods Code tabulateVotes() method Code determine_Winner_Loser() method Code settleTies() method Unit test tabulateVotes() Unit test determine_Winner_Loser() Unit test settleTies() Create JavaDocs for MV class Create Jinie comments for MV class Create system tests for MV class Create system tests for MV class		
I want to be able to process hallots that are designed for a municipal voting style election So that when the Secretary of State announces an election of that type, ballots can be appropriately counted and winners determined Acceptance Criteria: Election winners are properly determined Accurately count ballots where a voter has only voted for one candidate Accurately count ballots where a voter has voted for more than one candidate Accurately count ballots were a voter has voted for all candidates Settle a to between candidates with a flir coin toss mechanism Definition of Done: Decumentation (JavaDocs, In-line comments). Unit and integration tests are written and pass All exceptions are handled gracefully Effort: Large 5-6 hours	Josh Subhan Zach Larsen Zach Larsen Zach Larsen Zach Larsen Amelia Lunning Amelia Lunning Baanee Singh Baanee Singh Baanee Singh Josh Subhan	Create MV testing class Code MV constructor, setDeterministic, startElection methods Code tabulateVotes() method Code determine_Winner_Loser() method Code settleTies() method Unit test tabulateVotes() Unit test determine_Winner_Loser() Unit test settleTies() Create JavaDocs for MV class Create Jinie comments for MV class Create system tests for MV class Create system tests for MV class		
I want to be able to process hallots that are designed for a municipal voting style election So that when the Secretary of State announces an election of that type, ballots can be appropriately counted and winners determined Acceptance Criteria: Election winners are properly determined Accurately count ballots where a voter has only voted for one candidate Accurately count ballots where a voter has voted for more than one candidate Accurately count ballots were a voter has voted for all candidates Settle a to between candidates with a fair coin toss mechanism Definition of Done: Decumentation (JavaDocs, In-line comments). Unit and integration tests are written and pass All exceptions are handled gracefully Effort: Large 5-6 hours PBI Author(s): Josh Subhan	Josh Subhan Zach Larsen Zach Larsen Zach Larsen Zach Larsen Zach Larsen Amelia Lunning Amelia Lunning Baanee Singh Baanee Singh Baanee Singh Baanee Singh Josh Subhan Josh Subhan	Create MV testing class Code MV constructor, setDeterministic, startElection methods Code tabulateVotes() method Code determine_Winner_Loser() method Code settleTies() method Unit test tabulateVotes() Unit test determine_Winner_Loser() Unit test settleTies() Create JavaDocs for MV class Create JavaDocs for MV testing class Create in-line comments for MV class Create system tests for MV class Write out logs for tests		
I want to be able to process hallots that are designed for a municipal voting style election So that when the Secretary of State announces an election of that type, ballots can be appropriately counted and winners determined - Receptance Criteria: - Election winners are properly determined - Receptance Criteria: - Counted and the state of the season of the state of the season	Josh Subhan Zach Larsen Zach Larsen Zach Larsen Zach Larsen Zach Larsen Amelia Lunning Amelia Lunning Baanee Singh Baanee Singh Baanee Singh Baanee Singh Josh Subhan Josh Subhan	Create MV testing class Code MV constructor, setDeterministic, startElection methods Code tabulateVotes() method Code determine_Winner_Loser() method Code settleTies() method Unit test tabulateVotes() Unit test determine_Winner_Loser() Unit test settleTies() Create JavaDocs for MV class Create JavaDocs for MV testing class Create in-line comments for MV class Create system tests for MV class Write out logs for tests Create unit and integration tests		
I want to be able to process hallots that are designed for a municipal voting style election So that when the Secretary of State announces an election of that type, ballots can be appropriately counted and winners determined Acceptance Criteria: Election winners are properly determined Accurately count ballots where a voter has only voted for one candidate Accurately count ballots where a voter has vote for more than one candidate Accurately count ballots where a voter has vote for all candidates Settle at the between candidates with a fair coin toss mechanism Definition of Done: Documentation (JavaDocs, In-line comments). Unit and integration tests are written and pass All exceptions are handled gracefully Effort: Large Set hours PBI Author(s): Josh Subhan As an election official I want to be able to bring in multiple files from different balloting locations for both STV and Plurality voting	Josh Subhan Zach Larsen Zach Larsen Zach Larsen Zach Larsen Zach Larsen Zach Larsen Amelia Lunning Amelia Lunning Baanee Singh Baanee Singh Baanee Singh Josh Subhan Josh Subhan Josh Subhan Josh Subhan Baanee Singh	Create MV testing class Code MV constructor, setDeterministic, startElection methods Code tabulateVotes() method Code determine_Winner_Loser() method Code settleTies() method Unit test tabulateVotes() Unit test determine_Winner_Loser() Unit test settleTies() Create JavaDocs for MV class Create JavaDocs for MV testing class Create in-line comments for MV class Create system tests for MV class Write out logs for tests Create unit and integration tests Documentation in-line		
I want to be able to process hallots that are designed for a municipal voting style election So that when the Secretary of State announces an election of that type, ballots can be appropriately counted and winners determined Acceptance Criteria: Election winners are properly determined Accurately count ballots where a voter has only voted for one candidate Accurately count ballots where a voter has voted for more than one candidate Accurately count ballots where a voter has voted for all candidates Accurately count ballots were a voter has voted for all candidates Settle a tel between candidates with a flir coin toss mechanism Definition of Done: Denies to between candidates with a flir coin toss mechanism Definition of Done: Decumentation (JavaDocs, In-line comments). Unit and integration tests are written and pass All exceptions are handled gracefully Effort: Large 5-6 hours PBI Author(s): Josh Subhan As an election official I want to be able to bring in multiple files from different balloting locations for both STV and Pluratily voting St that the election data can be processed and managed efficiently from different locations.	Josh Subhan Zach Larsen Zach Larsen Zach Larsen Zach Larsen Zach Larsen Amelia Lunning Amelia Lunning Baanee Singh Baanee Singh Baanee Singh Baanee Singh Josh Subhan Josh Subhan Josh Subhan	Create MV testing class Code MV constructor, setDeterministic, startElection methods Code tabulateVotes() method Code determine_Winner_Loser() method Code settleTies() method Unit test tabulateVotes() Unit test determine_Winner_Loser() Unit test settleTies() Create JavaDocs for MV class Create JavaDocs for MV testing class Create in-line comments for MV class Create system tests for MV class Write out logs for tests Create unit and integration tests Documentation in-line Documentation javadocs		
I want to be able to process hallots that are designed for a municipal voting style election So that when the Secretary of State announces an election of that type, ballots can be appropriately counted and winners determined Acceptance Criteria: Election winners are properly determined Accurately count ballots where a voter has only voted for one candidate Accurately count ballots where a voter has voted for more than one candidate Accurately count ballots were a voter has voted for all candidates Settle as the between candidates with a flar coin toss mechanism Definition of Done: Decumentation (JavaDocs, In-line comments). Unit and integration tests are written and pass All exceptions are handled gracefully Effort: Large 5-6 hours PBI Author(s): Josh Subhan As an election official Lawant to be able to bring in multiple files from different balloting locations for both STV and Plurality voting So that the election data can be processed and managed efficiently from different locations. Acceptance Criteria: Sestern must be able to accept multiple set files when STV or PV are selected as the	Josh Subhan Zach Larsen Zach Larsen Zach Larsen Zach Larsen Zach Larsen Amelia Lunning Baanee Singh Baanee Singh Baanee Singh Josh Subhan Josh Subhan Josh Subhan Josh Subhan Baanee Singh	Create MV testing class Code MV constructor, setDeterministic, startElection methods Code tabulateVotes() method Code determine_Winner_Loser() method Unit test tabulateVotes() Unit test determine_Winner_Loser() Unit test settleTies() Create JavaDocs for MV class Create JavaDocs for MV testing class Create in-line comments for MV class Write out logs for tests Create unit and integration tests Documentation in-line Documentation javadocs Create a function to handle multiple file inputs		
I want to be able to process hallots that are designed for a municipal voting style election So that when the Secretary of State amounces an election of that type, ballots can be appropriately counted and winners determined Acceptance Criteria: Election winners are properly determined Accurately count ballots where a voter has vote for more than one candidate Accurately count ballots were a voter has vote for more than one candidate Accurately count ballots were a voter has vote for more than one candidate Accurately count ballots were a voter has vote for all candidates Settle at the between candidates with a fair coin toss mechanism Definition of Done: Documentation (JavaDocs, In-line comments). Unit and integration tests are written and pass All exceptions are handled gracefully Effort: Large Set hours PBI Author(s): Josh Subhan As an election official I want to be able to bring in multiple files from different balloting locations for both STV and Plurality voting. So that the election data can be processed and managed efficiently from different locations. Acceptance Criteria: Syntem must be able to accept multiple ey files when STV or PV are selected as the	Josh Subhan Zach Larsen Zach Larsen Zach Larsen Zach Larsen Zach Larsen Amelia Lunning Amelia Lunning Baanee Singh Baanee Singh Baanee Singh Baanee Singh Josh Subhan Josh Subhan Josh Subhan	Create MV testing class Code MV constructor, setDeterministic, startElection methods Code tabulateVotes() method Code determine_Winner_Loser() method Code settleTies() method Unit test tabulateVotes() Unit test determine_Winner_Loser() Unit test settleTies() Create JavaDocs for MV class Create JavaDocs for MV testing class Create in-line comments for MV class Create system tests for MV class Write out logs for tests Create unit and integration tests Documentation in-line Documentation javadocs		
I want to be able to process hallots that are designed for a municipal voting style election So that when the Secretary of State amounces an election of that type, ballots can be appropriately counted and winners determined Acceptance Criteria: Election winners are properly determined Accurately count ballots where a voter has vote for more than one candidate Accurately count ballots were a voter has vote for more than one candidate Accurately count ballots were a voter has vote for more than one candidate Accurately count ballots were a voter has vote for all candidates Settle at the between candidates with a fair coin toss mechanism Definition of Done: Documentation (JavaDocs, In-line comments). Unit and integration tests are written and pass All exceptions are handled gracefully Effort: Large Set hours PBI Author(s): Josh Subhan As an election official I want to be able to bring in multiple files from different balloting locations for both STV and Plurality voting. So that the election data can be processed and managed efficiently from different locations. Acceptance Criteria: Syntem must be able to accept multiple ey files when STV or PV are selected as the	Josh Subhan Zach Larsen Zach Larsen Zach Larsen Zach Larsen Zach Larsen Amelia Lunning Baanee Singh Baanee Singh Baanee Singh Josh Subhan Josh Subhan Josh Subhan Josh Subhan Baanee Singh	Create MV testing class Code MV constructor, setDeterministic, startElection methods Code tabulateVotes() method Code determine_Winner_Loser() method Unit test tabulateVotes() Unit test determine_Winner_Loser() Unit test settleTies() Create JavaDocs for MV class Create JavaDocs for MV testing class Create in-line comments for MV class Write out logs for tests Create unit and integration tests Documentation in-line Documentation javadocs Create a function to handle multiple file inputs		
I want to be able to process hallots that are designed for a municipal voting style election So that when the Secretary of State amounces an election of that type, ballots can be appropriately counted and winners determined Acceptance Criteria: - Election winces are properly determined - Recurately count hallots where a voter has voted for more than one candidate - Accurately count hallots where a voter has voted for more than one candidate - Accurately count hallots where a voter has voted for more than one candidate - Accurately count hallots were a voter has voted for more than one candidate - Accurately count hallots were a voter has voted for more than one candidate - Accurately count hallots were a voter has voted for more than one candidate - Accurately count hallots were a voter has voted for more than one candidate - Accurately count hallots were a voter has voted for more than one candidate - Manual to the second of the candidate of the can	Josh Subhan Zach Larsen Zach Larsen Zach Larsen Zach Larsen Zach Larsen Amelia Lunning Baanee Singh Baanee Singh Baanee Singh Josh Subhan Josh Subhan Josh Subhan Josh Subhan Baanee Singh	Create MV testing class Code MV constructor, setDeterministic, startElection methods Code tabulateVotes() method Code determine_Winner_Loser() method Unit test tabulateVotes() Unit test determine_Winner_Loser() Unit test settleTies() Create JavaDocs for MV class Create JavaDocs for MV testing class Create in-line comments for MV class Write out logs for tests Create unit and integration tests Documentation in-line Documentation javadocs Create a function to handle multiple file inputs		
I want to be able to process hallots that are designed for a municipal voting style election So that when the Secretary of State announces an election of that type, ballots can be appropriately counted and winners determined Acceptance Criteria: - Election winners are properly determined - Recurately count hallots where a voter has voted for more than one candidate - Accurately count hallots where a voter has voted for more than one candidate - Accurately count hallots were a voter has voted for more than one candidate - Accurately count hallots were a voter has voted for more than one candidate - Accurately count hallots were a voter has voted for more than one candidate - Accurately count hallots were a voter has voted for more than one candidate - Accurately count hallots were a voter has voted for all candidates - Settle a tie between candidates with a fair coin toss mechanism Definition of Done: - Decumentation (JavaDocs, In-line comments). - Unit and integration tests are written and pass - All exceptions are handled gracefully Effort: - Large - 5-6 hours PBI Author(s): Josh Subhan As an election official I want to be able to bring in multiple files from different balloting locations for both STV and Plurality voting So that the election data can be processed and managed efficiently from different locations. - Acceptance Criteria: - System must be able to accept multiple eavy files when STV or PV are selected as the election to pee - System must be able to scop accepting files when prompted by the user - System properly collects all the information from ballots in each eavy file - Definition of Done: - Reflactorine.	Josh Subhan Zach Larsen Zach Larsen Zach Larsen Zach Larsen Zach Larsen Amelia Lunning Baanee Singh Baanee Singh Baanee Singh Josh Subhan Josh Subhan Josh Subhan Josh Subhan Baanee Singh	Create MV testing class Code MV constructor, setDeterministic, startElection methods Code tabulateVotes() method Code determine_Winner_Loser() method Unit test tabulateVotes() Unit test determine_Winner_Loser() Unit test settleTies() Create JavaDocs for MV class Create JavaDocs for MV testing class Create in-line comments for MV class Write out logs for tests Create unit and integration tests Documentation in-line Documentation javadocs Create a function to handle multiple file inputs		
I want to be able to process hallots that are designed for a municipal voting style election So that when the Secretary of State announces an election of that type, ballots can be appropriately counted and winners determined Acceptance Criteria: - Election winners are properly determined - Accurately count ballots where a voter has voted for more than one candidate - Accurately count ballots we are a voter has voted for more than one candidate - Accurately count ballots we are a voter has voted for more than one candidate - Accurately count ballots were a voter has voted for more than one candidate - Accurately count ballots were a voter has voted for more than one candidate - Accurately count ballots were a voter has voted for all candidates - Settle a tie between candidates with a fair coin toss mechanism Definition of Done: - Documentation (JavaDoes, In-line comments). - Unit and integration tests are written and pass - All exceptions are handled gracefully Effort: - Large - 5-6 hours PBI Author(s): Josh Subhan As an election official I want to be able to bring in multiple files from different balloting locations for both STV and Plurality voting So that the election data can be processed and managed efficiently from different locations. Acceptance Criteria: - System must be able to accept multiple exv files when STV or PV are selected as the election to pee - System must be able to stop accepting files when prompted by the user - System must be able to stop accepting files when prompted by the user - System properly collects all the information from ballots in each exv file Definition of Done: - Reflections: - All exceptions are handled arracefully	Josh Subhan Zach Larsen Zach Larsen Zach Larsen Zach Larsen Zach Larsen Amelia Lunning Baanee Singh Baanee Singh Baanee Singh Josh Subhan Josh Subhan Josh Subhan Josh Subhan Baanee Singh	Create MV testing class Code MV constructor, setDeterministic, startElection methods Code tabulateVotes() method Code determine_Winner_Loser() method Unit test tabulateVotes() Unit test determine_Winner_Loser() Unit test settleTies() Create JavaDocs for MV class Create JavaDocs for MV testing class Create in-line comments for MV class Write out logs for tests Create unit and integration tests Documentation in-line Documentation javadocs Create a function to handle multiple file inputs		
I want to be able to process hallots that are designed for a municipal voting style election So that when the Secretary of State announces an election of that type, ballots can be appropriately counted and winners determined Acceptance Criteria: - Election winners are properly determined - Recurately count hallots where a voter has voted for more than one candidate - Accurately count hallots where a voter has voted for more than one candidate - Accurately count hallots were a voter has voted for more than one candidate - Accurately count hallots were a voter has voted for more than one candidate - Accurately count hallots were a voter has voted for more than one candidate - Accurately count hallots were a voter has voted for more than one candidate - Accurately count hallots were a voter has voted for all candidates - Settle a tie between candidates with a fair coin toss mechanism Definition of Done: - Decumentation (JavaDocs, In-line comments). - Unit and integration tests are written and pass - All exceptions are handled gracefully Effort: - Large - 5-6 hours PBI Author(s): Josh Subhan As an election official I want to be able to bring in multiple files from different balloting locations for both STV and Plurality voting So that the election data can be processed and managed efficiently from different locations. - Acceptance Criteria: - System must be able to accept multiple eavy files when STV or PV are selected as the election to pee - System must be able to scop accepting files when prompted by the user - System properly collects all the information from ballots in each eavy file - Definition of Done: - Reflactorine.	Josh Subhan Zach Larsen Zach Larsen Zach Larsen Zach Larsen Zach Larsen Amelia Lunning Baanee Singh Baanee Singh Baanee Singh Josh Subhan Josh Subhan Josh Subhan Josh Subhan Baanee Singh	Create MV testing class Code MV constructor, setDeterministic, startElection methods Code tabulateVotes() method Code determine_Winner_Loser() method Unit test tabulateVotes() Unit test determine_Winner_Loser() Unit test settleTies() Create JavaDocs for MV class Create JavaDocs for MV testing class Create in-line comments for MV class Write out logs for tests Create unit and integration tests Documentation in-line Documentation javadocs Create a function to handle multiple file inputs		
I want to be able to process hallots that are designed for a municipal voting style election So that when the Secretary of State amounces an election of that type, ballots can be appropriately counted and winners determined Acceptance Criteria: - Election winners are properly determined - Recurately count hallots where a voter has voted for more than one candidate - Accurately count hallots where a voter has voted for more than one candidate - Accurately count hallots where a voter has voted for more than one candidate - Accurately count hallots were a voter has voted for more than one candidate - Accurately count hallots were a voter has voted for more than one candidate - Accurately count hallots were a voter has voted for more than one candidate - Accurately count hallots were a voter has voted for more than one candidate - Accurately count hallots were a voter has voted for more than one candidate - Accurately count hallots were a voter has voted for more than one candidate - Definition of Done: - Demittion of Done: - Large - 5-6 hours - So hours - So hours - Bla Author(s): Josh Subhan - As an election official I want to be able to bring in multiple files from different balloting locations for both STV and Plurality voting - So that the election data can be processed and managed efficiently from different locations. - Acceptance Criteria: - System must be able to accept multiple cay files when STV or PV are selected as the election of pone: - System must be able to stop accepting files when prompted by the user - System must be able to stop accepting files when prompted by the user - System must be able to stop accepting files when prompted by the user - System properly collects all the information from ballots in each ear file - Refactoring. - Unit and integration tests are written and pass - All exceptions are handled gracefully - Declanding of the comments of the state of the comments of the com	Josh Subhan Zach Larsen Zach Larsen Zach Larsen Zach Larsen Zach Larsen Amelia Lunning Baanee Singh Baanee Singh Baanee Singh Josh Subhan Josh Subhan Josh Subhan Josh Subhan Baanee Singh	Create MV testing class Code MV constructor, setDeterministic, startElection methods Code tabulateVotes() method Code determine_Winner_Loser() method Unit test tabulateVotes() Unit test determine_Winner_Loser() Unit test settleTies() Create JavaDocs for MV class Create JavaDocs for MV testing class Create in-line comments for MV class Write out logs for tests Create unit and integration tests Documentation in-line Documentation javadocs Create a function to handle multiple file inputs		
I want to be able to process hallots that are designed for a municipal voting style election So that when the Secretary of State announces an election of that type, ballots can be appropriately counted and winners determined Acceptance Criteria: Election winners are properly determined Accurately count ballots where a voter has voted for one candidate Accurately count ballots where a voter has voted for more than one candidate Accurately count ballots where a voter has voted for more than one candidate Bellection of the counter of the counter of the counter of the candidate of the counter of t	Josh Subhan Zach Larsen Zach Larsen Zach Larsen Zach Larsen Zach Larsen Amelia Lunning Baanee Singh Baanee Singh Baanee Singh Josh Subhan Josh Subhan Josh Subhan Josh Subhan Baanee Singh	Create MV testing class Code MV constructor, setDeterministic, startElection methods Code tabulateVotes() method Code determine_Winner_Loser() method Unit test tabulateVotes() Unit test determine_Winner_Loser() Unit test settleTies() Create JavaDocs for MV class Create JavaDocs for MV testing class Create in-line comments for MV class Write out logs for tests Create unit and integration tests Documentation in-line Documentation javadocs Create a function to handle multiple file inputs		
I want to be able to process hallots that are designed for a municipal voting style election So that when the Secretary of State amounces an election of that type, ballots can be appropriately counted and winners determined Acceptance Criteria: - Bleetion winners are properly determined - Accurately count hallots where a voter has voted for more than one candidate - Accurately count hallots where a voter has voted for more than one candidate - Accurately count hallots where a voter has voted for more than one candidate - Accurately count hallots were a voter has voted for more than one candidate - Accurately count hallots were a voter has voted for more than one candidate - Accurately count hallots were a voter has voted for more than one candidate - Accurately count hallots were a voter has voted for more than one candidate - Accurately count hallots were a voter has vote for all candidates - Settle a tile between candidates with a fair coin toss mechanism Definition of Done: - Decumentation (Ivan Does, In-line comments). - All exceptions are handled gracefully Effort: - Large - 3-6 hours - 3-6 hours - 3-6 hours - 3-6 hours - 4-8 hours - 4-8 hours - 5-6 hours - 5-7 hours - 5-7 hours - 5-7 hours - 5-8 hours - 5-8 hours - 5-8 hours - 6-8 hours - 6-8 hours - 7-8 hours - 7-9 hours - 8-1 hours	Josh Subhan Zach Larsen Zach Larsen Zach Larsen Zach Larsen Zach Larsen Amelia Lunning Baanee Singh Baanee Singh Baanee Singh Josh Subhan Josh Subhan Josh Subhan Josh Subhan Baanee Singh	Create MV testing class Code MV constructor, setDeterministic, startElection methods Code tabulateVotes() method Code determine_Winner_Loser() method Unit test tabulateVotes() Unit test determine_Winner_Loser() Unit test settleTies() Create JavaDocs for MV class Create JavaDocs for MV testing class Create in-line comments for MV class Write out logs for tests Create unit and integration tests Documentation in-line Documentation javadocs Create a function to handle multiple file inputs		
I want to be able to process hallots that are designed for a municipal voting style election So that when the Secretary of State announces an election of that type, ballots can be appropriately counted and winners determined Acceptance Criteria: Election winners are properly determined Accurately count ballots where a voter has only voted for one candidate Accurately count ballots where a voter has voted for more than one candidate Accurately count ballots where a voter has voted for more than one candidate Accurately count ballots where a voter has voted for more than one candidate Accurately count ballots where a voter has voted for more than one candidate Accurately count ballots where a voter has voted for more than one candidate Better than the state of the sta	Josh Subhan Zach Larsen Zach Larsen Zach Larsen Zach Larsen Zach Larsen Amelia Lunning Baanee Singh Baanee Singh Baanee Singh Josh Subhan Josh Subhan Josh Subhan Josh Subhan Baanee Singh	Create MV testing class Code MV constructor, setDeterministic, startElection methods Code tabulateVotes() method Code determine_Winner_Loser() method Unit test tabulateVotes() Unit test determine_Winner_Loser() Unit test settleTies() Create JavaDocs for MV class Create JavaDocs for MV testing class Create in-line comments for MV class Write out logs for tests Create unit and integration tests Documentation in-line Documentation javadocs Create a function to handle multiple file inputs		