SELECTED BACKLOG ITEMS	DEV NAME	TASKS NOT STARTED	TASKS IN PROGRESS	TASKS COMPLETED
As an election official	Amelia Lunning			Fix bug where running STV with 1-2 seats results in an OutOfMemory error
I want to be able to process an STV-style election correctly So that when the Secretary of State announces an election of that type, winners can be	Amelia Lunning		Add additional unit and system tests for STV	
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 than or equal to the number of candidates, but greater than zero. 				
Ballot shuffling is toggleable and functions correctly 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				
- Baance Singh, Josh Subhan				
	Danner Olark		Editoration de the traillier anno all lafe form the bondon	
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		Edit methods that will parse all info from the headers.	Documentation in-line
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)	Documentation in-line
	Baanee Singh		Implement new main() logic	
Acceptance Criteria: - System should successfully parse the following from the file:	Baanee Singh		Implement new main() logic	Documentation javadocs
System should successfully parse the following from the file: Election type Number of seats	Josh Subhan		Refactoring Main	Documentation javadocs
- Number of candidates - Number of ballots	JUSTI GUDITATI		Relacioning Walli	
- 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: - Medium				
- 3-4 hours				
PBI Author(s): Josh Subhan				
	Zach Larsen			Create MV class
	Zach Larsen			Create MV testing class
	Zach Larsen Zach Larsen			Create MV testing class Code MV constructor, setDeterministic, startElection methods
	Zach Larsen Zach Larsen Zach Larsen			Create MV testing class Code MV constructor, setDeterministic, startElection methods Code tabulateVotes() method
As an election official I want to be able to process ballots that are designed for a municipal voting style election	Zach Larsen 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	Zach Larsen Zach Larsen Zach Larsen Zach Larsen Zach Larsen Zach Larsen			Create MV testing class Code MV constructor, setDeterministic, startElection methods Code tabulateVotes() 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	Zach Larsen		Unit test tabulateVotes()	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 amounces an election of that type, ballots can be appropriately counted and winners determined Acceptance Criteria: - Election winners are properly determined	Zach Larsen		Unit test determine_Winner_Loser()	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 amounces an election of that type, ballots can be appropriately counted and winners determined Acceptance Criteria: Election winners are properly determined Accuptate Criteria: Accuptate Criteria:	Zach Larsen		Unit test determine_Winner_Loser() Unit test settleTies()	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 amounces an election of that type, ballots can be appropriately counted and winners determined Acceptance Criteria: - Election winners are properly determined - Recurately counted 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	Zach Larsen Baanee Singh		Unit test determine_Winner_Loser() Unit test settleTies() Create JavaDocs for MV class	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 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 only voted for one candidate Accurately count ballots where a voter has voted for more than one candidate Accurately count ballots when a voter has voted for all candidates Settle a to between candidates with a fire root nots mechanism	Zach Larsen Banee Singh Banee Singh		Unit test determine_Winner_Loser() Unit test settleTies() Create JavaDocs for MV class Create JavaDocs for MV testing class	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 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 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 more than one candidate - Accurately count ballots were a voter has voted for more than one candidate - Accurate yount ballots were a voter has voted for more stan one candidate - Accurate Settle at the between candidates with a fair coin toss mechanism Definition of Dans:	Zach Larsen Baanee Singh	Create system tests for MV class	Unit test determine_Winner_Loser() Unit test settleTies() Create JavaDocs for MV class	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 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 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 more than one candidate - Accurately count ballots were a voter has voted for more than one candidate - Settle a tie between candidates with a fair coin toss mechanism Definition of Done: - Documentation (Isva Docs, In-line comments). - Unit and integration tests are written and pass	Zach Larsen Baanee Singh Baanee Singh Baanee Singh	Create system tests for MV class Write out logs for tests	Unit test determine_Winner_Loser() Unit test settleTies() Create JavaDocs for MV class Create JavaDocs for MV testing class	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 amounces an election of that type, ballots can be appropriately counted and winners determined Acceptance Criteria: - Election winners are properly determined - Accentacy count ballots 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 all candidates - Settle at its 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 landed gancefully	Zach Larsen Baanee Singh Baanee Singh Josh Subhan	Create system tests for MV class Write out logs for tests	Unit test determine_Winner_Loser() Unit test settleTies() Create JavaDocs for MV class Create JavaDocs for MV testing class	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 amounces an election of that type, ballots can be appropriately counted and winners determined **Cecplance 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 more than one candidate - Accurately count ballots were a voter has voted for fine and the second of	Zach Larsen Baanee Singh Baanee Singh Josh Subhan Josh Subhan		Unit test determine_Winner_Loser() Unit test settleTies() Create JavaDocs for MV class Create JavaDocs for MV testing class	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 ballots 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 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 as the between candidates with a fair cost 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 3-6 hours	Zach Larsen Bannee Singh Baanee Singh Josh Subhan Zosh Lursen		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 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 ballots 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 **Cecplance 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 more than one candidate - Accurately count ballots were a voter has voted for fine and the second of	Zach Larsen Baanee Singh Baanee Singh Baanee Singh Josh Subhan Josh Subhan Josh Subhan		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 MV testing class Code MV constructor, setDeterministic, startElection methods Code tabulateVotes() method Code determine_Winner_Loser() method Code settleTies() method Refactor plurality and MV inheritance structure
I want to be able to process ballots 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 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 as the between candidates with a fair cost 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 3-6 hours	Zach Larsen Baanee Singh Baanee Singh Baanee Singh Josh Subhan Josh Subhan Josh Subhan Josh Subhan Josh Subhan	Write out logs for tests	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 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 ballots 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 proyerly determined Accurately count ballots whose a voter has voted for more than one candidate Accurately count ballots whose a voter has voted for more than one candidate Accurately count ballots whose a voter has voted for more than one candidate 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 3-6-hours PBI Author(s): Josh Subhan	Zach Larsen Baanee Singh Baanee Singh Baanee Singh Josh Subhan Josh Subhan Josh Subhan Josh Subhan Josh Subhan	Write out logs for tests Create unit and integration tests	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 MV testing class Code MV constructor, setDeterministic, startElection methods Code tabulateVotes() method Code determine_Winner_Loser() method Code settleTies() method Refactor plurality and MV inheritance structure
I want to be able to process ballots 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 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 Settle at the between candidates with a fair coin to some mechanism Definition of Done: Documentation (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 An election official I want to be able to bring in multiple files from different balloting locations for both STV and	Zach Larsen Baanee Singh Baanee Singh Baanee Singh Josh Subhan Josh Subhan Josh Subhan Josh Subhan Josh Subhan Bashanee Singh	Write out logs for tests Create unit and integration tests Documentation in-line	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 MV testing class Code MV constructor, setDeterministic, startElection methods Code tabulateVotes() method Code determine_Winner_Loser() method Code settleTies() method Refactor plurality and MV inheritance structure
I want to be able to process ballots 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 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 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 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 more same candidate - Accurately count ballots were a voter has voted for more same candidate - Accurately count ballots were a voter has voted for more same candidate - Accurately count ballots were a voter has voted for more same candidate - Accurately count ballots were a voter has voted for more same candidate - Accurately count ballots were a voter has voted for more same candidate - Accurately count ballots were a voter has voted for more same candidate - Accurately count ballots were a voter has voted for more same candidate - Accurately count ballots were a voter has voter has voted for more same candidate - Accurately count ballots were a voter has voter h	Zach Larsen Baanee Singh Baanee Singh Baanee Singh Josh Subhan Josh Subhan Josh Subhan Josh Subhan Josh Subhan Bashee Singh	Write out logs for tests Create unit and integration tests	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 MV testing class Code MV constructor, setDeterministic, startElection methods Code tabulateVotes() method Code determine_Winner_Loser() method Code settleTies() method Refactor plurality and MV inheritance structure Create new DirectSelection abstract class
I want to be able to process ballots 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 - 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 Subban 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 Criteries:	Zach Larsen Baanee Singh Baanee Singh Josh Subhan Josh Subhan Josh Subhan Josh Subhan Josh Subhan Josh Subhan	Write out logs for tests Create unit and integration tests Documentation in-line	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 MV testing class Code MV constructor, setDeterministic, startElection methods Code tabulateVotes() method Code determine_Winner_Loser() method Code settleTies() method Refactor plurality and MV inheritance structure Create new DirectSelection abstract class Create a function to handle multiple file inputs
I want to be able to process ballots 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 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 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 more than one candidate - Accurately count ballots were a voter has voted for all candidates - Bettle a the between candidates with a fair coin toss mechanism - Definition of Done: - Documentation (Asa Docs, In-line comments). - Unit and integration tests are written and pass - All exceptions are handled gracefully - Large - 5-6 hours - Bl Author(s): Josh Subban - As an election official - I want to be able to bring in multiple files from different balloting locations for both STV and Plurality voting - Substantial conditions and the processed and managed efficiently from different locations. - Acceptance Criteria: - System must be able to accept multiple gay files when STV or PV are selected as the	Zach Larsen Baanee Singh Baanee Singh Baanee Singh Josh Subhan Josh Subhan Josh Subhan Josh Subhan Josh Subhan Bashee Singh	Write out logs for tests Create unit and integration tests Documentation in-line	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 MV testing class Code MV constructor, setDeterministic, startElection methods Code tabulateVotes() method Code determine_Winner_Loser() method Code settleTies() method Refactor plurality and MV inheritance structure Create new DirectSelection abstract class
I want to be able to process ballots 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 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 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 Detailor of a control of the 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 eyr files when STV or PV are selected as the - System must be able to accept multiple eyr files when STV or PV are selected as the	Zach Larsen Baanee Singh Baanee Singh Josh Subhan Josh Subhan Josh Subhan Josh Subhan Josh Subhan Josh Subhan	Write out logs for tests Create unit and integration tests Documentation in-line	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 MV testing class Code MV constructor, setDeterministic, startElection methods Code tabulateVotes() method Code determine_Winner_Loser() method Code settleTies() method Refactor plurality and MV inheritance structure Create new DirectSelection abstract class Create a function to handle multiple file inputs
I want to be able to process ballots 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 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 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 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 Doc: - Decumentation ((avaDocs, 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 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 cyr file Definition of Doc:	Zach Larsen Baanee Singh Baanee Singh Josh Subhan Josh Subhan Josh Subhan Josh Subhan Josh Subhan Josh Subhan	Write out logs for tests Create unit and integration tests Documentation in-line	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 MV testing class Code MV constructor, setDeterministic, startElection methods Code tabulateVotes() method Code determine_Winner_Loser() method Code settleTies() method Refactor plurality and MV inheritance structure Create new DirectSelection abstract class Create a function to handle multiple file inputs
I want to be able to process ballots 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 Aceptance 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 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 (IsvaDoos, In-line comments) - Unit and integration tests are written and pass - All exceptions are handled gracefully Effort: - Large - 3-6 hours PBI Author(s): Josh Subhan As an election official - Austral to be able to bring in multiple files from different balloting locations for both STV and Purality voting - So that the election data can be processed and managed efficiently from different locations. - Acceptance Criteria: - 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 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 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 must be able to stop accepting files when prompted by the user	Zach Larsen Baanee Singh Baanee Singh Josh Subhan Josh Subhan Josh Subhan Josh Subhan Josh Subhan Josh Subhan	Write out logs for tests Create unit and integration tests Documentation in-line	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 MV testing class Code MV constructor, setDeterministic, startElection methods Code tabulateVotes() method Code determine_Winner_Loser() method Code settleTies() method Refactor plurality and MV inheritance structure Create new DirectSelection abstract class Create a function to handle multiple file inputs
I want to be able to process ballots 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 Aceptance 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 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 (IsvaDoos, In-line comments) - Unit and integration tests are written and pass - All exceptions are handled gracefully Effort: - Large - 3-6 hours PBI Author(s): Josh Subhan As an election official - Austral to be able to bring in multiple files from different balloting locations for both STV and Purality voting - So that the election data can be processed and managed efficiently from different locations. - Acceptance Criteria: - 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 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 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 must be able to stop accepting files when prompted by the user	Zach Larsen Baanee Singh Baanee Singh Josh Subhan Josh Subhan Josh Subhan Josh Subhan Josh Subhan Josh Subhan	Write out logs for tests Create unit and integration tests Documentation in-line	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 MV testing class Code MV constructor, setDeterministic, startElection methods Code tabulateVotes() method Code determine_Winner_Loser() method Code settleTies() method Refactor plurality and MV inheritance structure Create new DirectSelection abstract class Create a function to handle multiple file inputs
I want to be able to process ballots 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 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 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 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 - Accurately count ballots were a voter has voted for all candidates - 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 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 eye file - Decumentation flows. In children the pass of the contents of th	Zach Larsen Baanee Singh Baanee Singh Josh Subhan Josh Subhan Josh Subhan Josh Subhan Josh Subhan Josh Subhan	Write out logs for tests Create unit and integration tests Documentation in-line	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 MV testing class Code MV constructor, setDeterministic, startElection methods Code tabulateVotes() method Code determine_Winner_Loser() method Code settleTies() method Refactor plurality and MV inheritance structure Create new DirectSelection abstract class Create a function to handle multiple file inputs
I want to be able to process ballots 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 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 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 (favaDoos, 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 - Lawat to be able to bring in multiple files from different balloting locations for both STV and Purality voting - So that the election data can be processed and managed efficiently from different locations. - Acceptance Criteria: - Selection type - System must be able to accept multiple gw files when STV or PV are selected as the election type - System must be able to stop accepting files when prompted by the user - System properly collects all the information from ballots in each say file Definition of Dace: - Reflectoring: - Unit and integration tests are written and pass - All exceptions are handled gracefully - Documentation (JavaDoos, In-line comments) Effort: - Medium	Zach Larsen Baanee Singh Baanee Singh Josh Subhan Josh Subhan Josh Subhan Josh Subhan Josh Subhan Josh Subhan	Write out logs for tests Create unit and integration tests Documentation in-line	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 MV testing class Code MV constructor, setDeterministic, startElection methods Code tabulateVotes() method Code determine_Winner_Loser() method Code settleTies() method Refactor plurality and MV inheritance structure Create new DirectSelection abstract class Create a function to handle multiple file inputs
I want to be able to process ballots 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 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 - 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 sendances - All exceptions are handled gracefully - As an election official I want to be able to bring in multiple files from different balloting locations for both STV and Purality voting - State the election data can be processed and managed efficiently from different locations. - System must be able to so accept multiple eavy files when STV or PV are selected as the election of Done: - System must be able to so accepting files when prompted by the user - System must be able to so accepting files when prompted by the user - System more benefit of the information from ballots in each exp file - Definition of Done: - Reflectoring - Unit and integration tests are written and pass - All exceptions are handled gracefully - Decland of the process of t	Zach Larsen Baanee Singh Baanee Singh Josh Subhan Josh Subhan Josh Subhan Josh Subhan Josh Subhan Josh Subhan	Write out logs for tests Create unit and integration tests Documentation in-line	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 MV testing class Code MV constructor, setDeterministic, startElection methods Code tabulateVotes() method Code determine_Winner_Loser() method Code settleTies() method Refactor plurality and MV inheritance structure Create new DirectSelection abstract class Create a function to handle multiple file inputs
I want to be able to process ballots 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 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 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 (favaDoos, 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 - Lawat to be able to bring in multiple files from different balloting locations for both STV and Purality voting - So that the election data can be processed and managed efficiently from different locations. - Acceptance Criteria: - Selection type - System must be able to accept multiple gw files when STV or PV are selected as the election type - System must be able to stop accepting files when prompted by the user - System properly collects all the information from ballots in each say file Definition of Dace: - Reflectoring: - Unit and integration tests are written and pass - All exceptions are handled gracefully - Documentation (JavaDoos, In-line comments) Effort: - Medium	Zach Larsen Baanee Singh Baanee Singh Josh Subhan Josh Subhan Josh Subhan Josh Subhan Josh Subhan Josh Subhan	Write out logs for tests Create unit and integration tests Documentation in-line	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 MV testing class Code MV constructor, set/Deterministic, startElection methods Code tabulateVotes() method Code determine_Winner_Loser() method Code settleTies() method Refactor plurality and MV inheritance structure Create new DirectSelection abstract class Create a function to handle multiple file inputs