| | Due Date | 4/20/2022 | 4/20/2022 | 4/20/2022 | 4/20/2022 | 4/20/2022 | 4/20/2022 | 4/20/2022 | 4/20/2022 | 4/20/2022 | 4/20/2022 | 4/22/2022 | 4/22/2022 | 4/22/2022 | 4/22/2022 | 4/22/2022 | 4/22/2022 | 4/22/2022 | 4/22/2022 | | | |
|--------------------------|--|--|---|---|--|--|--|--|---|---|--|---|---|---|--|---|---|--|--|--|--|--|
| Sprint Backlog 4/27/2022 | Definition of Done | | All previous regression tests pass Regression tests for new functionality run automatically | every build 3) Code written in pairs/ viewed by other members | 4) Duplicate code is removed through refactoring 5) Messy code cleaned up through refactoring 6) manual explorationy testing has been conducted 7) | 7) Uneokout and build fully reproducible (1-2 commands) | 1) All previous regression tests pass 2) Regression tests for new fundionality run automatically 3) Code written in parts/ viewed by other members 4) Duplicate code is removed through reflacting 5) Messy code cleaned up through reflacting 6) manual, exploratory testing has been conducted 7) Checkout and build fully reproducible (1-2 commands) | | | | | 2) Regression tests pass and unctionality run automatically 3) Code written in pairs' (wiewed by other members 4) Dualicate code is emroved through refacting 5) Nessy code cleaned up through refacting 6) manual a cyptoratory testing has been conducted 7) Checkout and build fully reproducible (1-2 commands) | | | | | | | | | | |
| | Acceptance Criteria Definition of Done | The user can submit a file for input | command can be run to calculate correct IRV election results | receives correct results of the election | all tests pass and system is able to test runtime of 100000 ballots | Functionality of calcResults is maintained, but all functionality that involves reading from files is moved to readinput() | The user can submit a file for input | command can be run to calculate correct OPL election results | receives correct results of the election | all tests pass and system is able to test runtime of 100000 ballots | Functionality of calcResults is maintained, but all functionality that involves reading from files is moved to readinput() | "flename" command does not cause previous input flesnames to be overwritten, and instead adds the supplied flename to a list of flenames | FileHandler no longer keeps track of a single filename. Instead, FileHandler maintains a list of filenames. | FileHandler.setinputfile() no longer exists, and FileHandler.addinputfile() is created with full functionality, New function checks the supplied filename for if texists, If it axists, then it is added to the list of filenames. Otherwise it is ginored. | System creates a Filethander object when the program starts up, this object is chosed when the program is chosed with the exit" command. The "sxit" command. The "sxit" command a Fielthander object when the first valid filename is passed in. | FileHandler has a getter function which returns the FileHandler's list of files. | an OPL election can be ran which draws its inputs from more than one input file | an IRV election can be ran which draws its inputs from more than one input file | the user can remove a filename from the list of filenames via a command. | | | |
| | Who's working on it | | Ann, Sean | | | | | Ann, Sean | | | | | | | | | | | | | | |
| | Status | Done | Done | Done | Started | Done | Done | Done | Done | Started | Done | Done | Done | Done | Done | Done | Done | Done | Done | | | |
| | Est Time (hours) | 0.5 | 0.5 | 0.5 | 2 | 3.0 | 0.5 | 1.5 | 0.5 | n | 0.5 | 0.25 | 0.25 | 0.25 | 0.25 | 0.25 | 0.5 | 0.5 | 0.5 | | | |
| | | Fix file parsing to allow multiple winners/ensure that file parsing works properly | Fix calcResults() function to allow multiple winners | Ensure that correct winners can be retrieved by user | Unit testing | bonus: split calcResults() into readinput() and calcResults() (low priority) | Ensure that file parsing works properly | Fix calcResults() function to calculate the winners correctly | Ensure that correct winners can be retrieved by user | Unit Testing | bonus: split calcResults() into readinput() and calcResults() (low priority) | Change input loop so that "Henane" always adds the given filename to a list of files, rather than overwriting previous filenames | give the filehandler a list of filenames rather than just one filename | replace FleHandler.sethputFile() with something like FleHandler. addinputFile(), this new method should behave similarly to sethputFile but should add valid files to the last of filenames rather man just overwriting the previous | possibly change from creating a FileHandler when the first file is speased in, to creating one when the program starts? | create a getter for FIIeHandler. filenames size (probably useful for adding multiple file support to electionOPL/electionIRV) | update electionOPL to support reading input from multiple files | update electionIRV to support reading input from multiple files | possibly create function which makes it possible to remove files from the list of input files (low priority) | | | |
| | Owner T | Official | | | | | | | | | 256 | Flection Official | | | | | | | | | | |
| | User Story C | n official I I umber of umber of when to mumber of V so that I vyself in official I I umber of official I I umber of official I I umber official I umber officia | | | | | | | | | , | | As an election official I want to be able to bring in multiple files instead of a single file | | | | | | | | | |
| | PBIID | # # | | | | | | | | | | | | | £# | | | | | | | |

| 22 | 55 | 22 | 22 | 22 | 22 | 22 | 22 | | 22 | 122 | 122 | 22 | 22 | | |
|--|--|--|---|--|---|--|---|--|---|---|--|---|--|--|--|
| 4/22/2022 | 4/25/2022 y | 4/25/2022 | 4/25/2022 | 4/27/2022 | y 4/27/2022 | 4/27/2022 | 4/27/2022 | | 4/27/2022 | y 4/27/2022 | 4/27/2022 |) 4/27/2022 | 4/27/2022 | | |
| | 1) All previor 2) Regression tests for | 3) Code written in pairs/ viewed by other members 4) Duplicate code is removed through refactoring 5) Messy code deamed up frunding refactoring 6) Messy code deamed up frunding refactoring 6) manual, exploratory testing has been conducted 7) Checkout and build fully reproducible (1-2 commands) | | | All previous regression tests pass Regression tests for new functionality run automatically every build | J. Ocea withen in plass viewed us of onter interests. 4) Duplicate code is removed through reflactoring. 5) Messay code cleared up intrough reflactoring. 6) Messay code cleared up intrough reflactoring. 6) manual. exploratory testing has been conducted (6) manual. 7) Checkout and build fully reproducible (1-2 commands). | | | 1) All previous regression lests pass 2) Regression tests for new functionality un automatically 3) Code written in pairs viewed by other members 4) budgidate gode is removed through relacioning 5) Messy code cleaned up through relacioning 6) manual. exploratory testing has been conducted 7) Checkout and builk fully reproducible (1-2 commands) | | | | | | |
| All unit tests for each of the tasks for this PBI pass, along with all previous unit tests. | An IRV election can be fed invalid ballots via input file(s) and the election will automatically ignore and mark the invalid ballots in the relevant audit file | Can log a string to the audit file via some FileHandler method. This method takes a string and logs that string to the audit file. | all tests pass and system is able to test runtime of 100000 ballots | ElectionPO class exists and extends the Election class | ElectionPO readInputs() exists and can read all candidates/votes in from all supplied input files. | CandidatePO class exists and has all needed functionality which the Candidate class did not have. | all tests pass and system is able to test runtime of 100000 ballots | | command can be run to calculate correct PO election results | receives correct results of the election | existing input command can read a PO election option | all tests pass and system is able to test runtime of 100000 ballots | Coin filp in case of tie is fair and randomized | | |
| Ann, Sean | | | Ann, Sean | | | | Ann, Sean | | | | | Shey, Jasper | | | |
| Started | Done | Done | Started | Done | Done | Done | Started | | Done | 0.5 Done | Done | Started | Done | | |
| 2 | | 5.0 | , | 0.25 | 1.5 | . s ₀ 0 | 5 | | 1.5 | 0.5 | 0.5 | 2 | 0.5 | | |
| Unit testing | change the code which reads the ballots in so that it checks each ballots in so that it checks each ballot for if it is valid. Invalid ballots will be disregarded/added to audif file | update FileHandler to support this new audit file (add a function) | Unit testing | create ElectionPO dass | write ElectionPO.readInputs() which reads all supplied files for inputs. | If the default Candidate class is not enough to complete this, then write a CandidatePO class | unit testing | | create ElectionPO.calcResults() which behaves similarly to ElectionIRV.calcResults() and ElectionOPL.calcResults() | ensure that correct winners can be retrieved by user | create input command to request PO election | unit testing | write a fair random coin flip (possibly use the one which was already written) | | |
| | | | | | | | | | | | | | | | |
| | As an election official, I | want to be able to automatically filter out the ballots which have less than half the candidates ranked in an IRV election | | | As an election official, I want to be able to bring in PO election ballots via a. | As an election official, i went to be able in the beather of the candidate who received the most ballots) | | | | | | | | | |
| | | 6# # | | 4# | | | | | | ç. | | | | | |
| | | | | | | | | | | | | | | | |