

User story 1	As the product owner, I want the program to be able to handle 100,000 ballots so that I can use it for a local election.
Acceptance criteria	<ol style="list-style-type: none"> 1. Number of ballots should not exceed 100,000 2. The program gives the correct result with 100,000 ballots. 3. The program runs in acceptable amount of time.
Effort	medium

User story 2	As an election official, I want to be able to see ballots and inputs on the screen directly into the program so that I don't have to parse the ballots.
Acceptance criteria	<ol style="list-style-type: none"> 1. Accept ballots from Linux machines 2. Ballots are correctly printed onto the screen after reading input file 3. Generate a message of success/failure
Effort	medium

User story 3	As an election official, I want to see the progress of distributing votes on the screen as it's running so that I will be able to know the status of each candidate as well as each ballot
Acceptance criteria	<ol style="list-style-type: none"> 1. Votes distribution is printed on the screen 2. A list of ballots associated with each candidate is printed on the screen 3. A list of winners is printed on the screen 4. A list of losers is printed on the screen
Effort	large

User story 4	As an election official, I want the program to be able to handle errors from user so that the program will not malfunction if there is any mistake from user input.
Acceptance criteria	<ol style="list-style-type: none"> 1. Handle different errors in the input file 2. Generate an error message if the program encounters any error 3. Generate an useful message for user on how to potentially fix this error. 4. The program quit after encountering an error.
Effort	medium

