ESaaS
Our Process

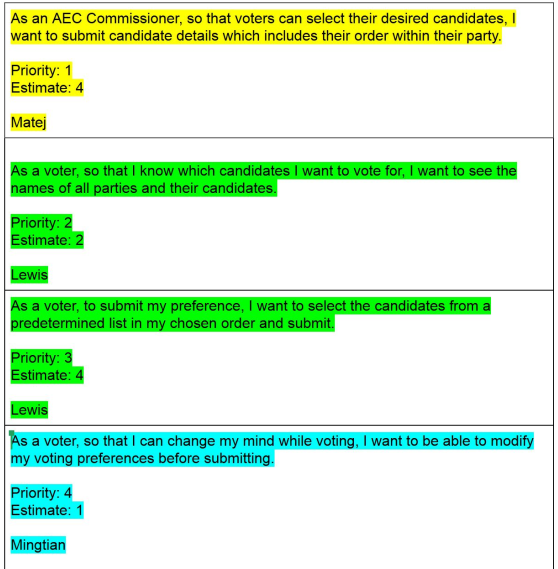
By Guy Farrelly, Matej Hosu, Lewis Cabban, Mingtian Tao and Jiarong Fan

Identifying Requirements

- Brainstorm basics
- Collaboratively merge

Creating User Stories

- Formalise the requirements into user story format
- Specify an entity, purpose and task
- Short and simple



As an AEC Commissioner, so that voters can select their desired candidates, I want to submit candidate details which includes their order within their party.

Priority: 1
Estimate: 4

Matej

As a voter, so that I know which candidates I want to vote for, I want to see the names of all parties and their candidates.

Priority: 2
Estimate: 2

Lewis

As a voter, to submit my preference, I want to select the candidates from a predetermined list in my chosen order and submit.

Priority: 3
Estimate: 4

Lewis

As a voter, so that I can change my mind while voting, I want to be able to modify my voting preferences before submitting.

Priority: 4
Estimate: 1

Mingtian

Verification

- Clarify any remaining questions we have with the ‘client’
- Remove unnecessary aspects

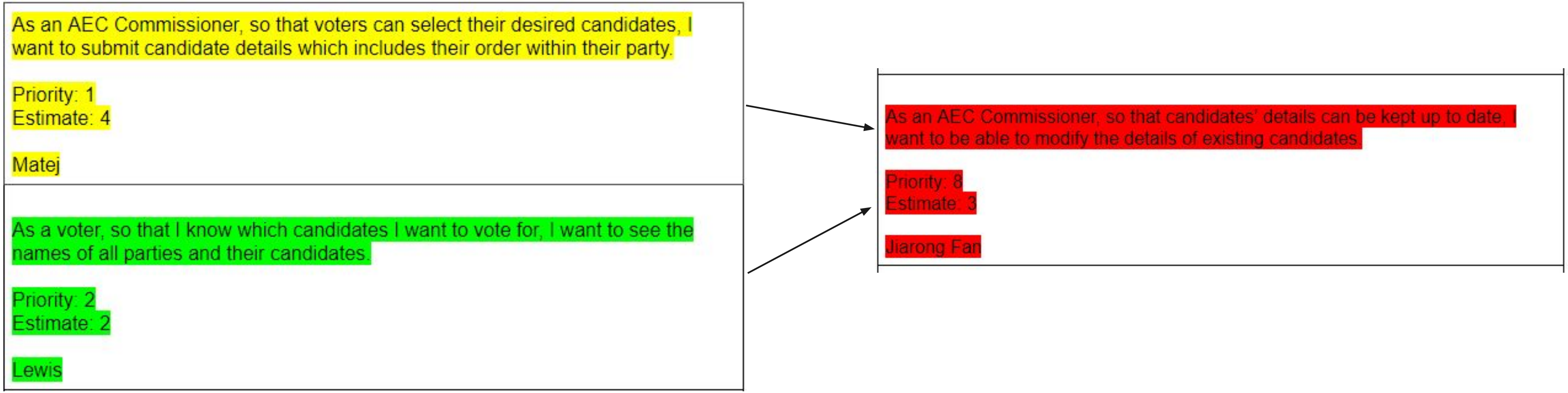
Assigning Priorities and Time Estimates

- Come to agreement on priority and estimated difficulty
- Requirements poker

User Story Allocation

- Began by assigning a user story to each member of the group.
- Based on priority of requirements
- And the logical order in which we thought they should be completed.

Priority and Logical Order



As an AEC Commissioner, so that voters can select their desired candidates, I want to submit candidate details which includes their order within their party.

Priority: 1
Estimate: 4

Matej

As a voter, so that I know which candidates I want to vote for, I want to see the names of all parties and their candidates.

Priority: 2
Estimate: 2

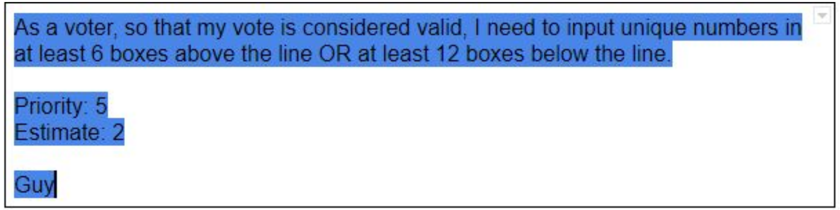
Lewis

As an AEC Commissioner, so that candidates' details can be kept up to date, I want to be able to modify the details of existing candidates.

Priority: 8
Estimate: 3

Jiarong Fan

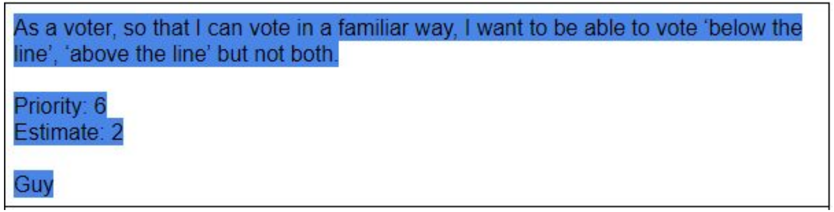
Completing User Stories



As a voter, so that my vote is considered valid, I need to input unique numbers in at least 6 boxes above the line OR at least 12 boxes below the line.

Priority: 5
Estimate: 2

Guy



As a voter, so that I can vote in a familiar way, I want to be able to vote 'below the line', 'above the line' but not both.

Priority: 6
Estimate: 2

Guy

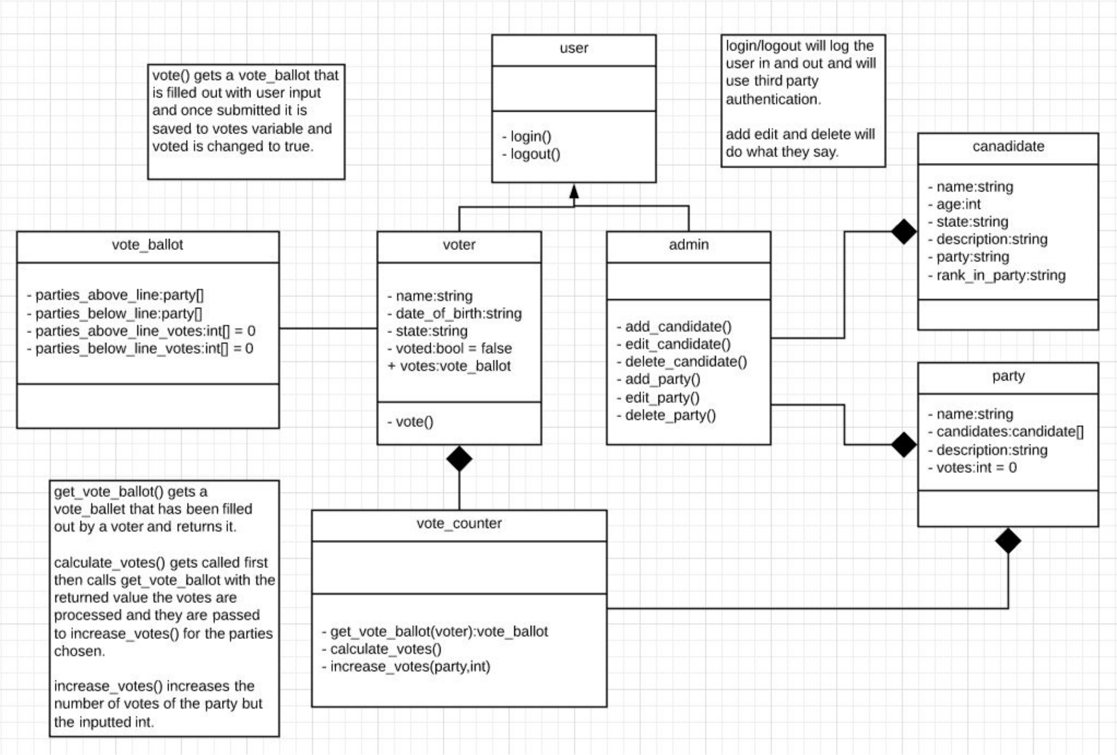
- Assign another user story to ourselves after completion of one.
- Tried to base next chosen story on a related story.
- Choose a story which was similar or related to a previous one you completed.

Maintaining Communication

- Weekly meetings to discuss progress and seek help.
- Solve issues we had collectively.
- Facebook group chat to discuss progress in between meetings.

Class Diagram

- One of the main starting points for the app
- Used to gain an understanding of how the overall app would work



vote() gets a vote_ballot that is filled out with user input and once submitted it is saved to votes variable and voted is changed to true.

user

- login()
- logout()

login/logout will log the user in and out and will use third party authentication.

add edit and delete will do what they say.

canadidate

- name:string
- age:int
- state:string
- description:string
- party:string
- rank_in_party:string

vote_ballot

- parties_above_line:party[]
- parties_below_line:party[]
- parties_above_line_votes:int[] = 0
- parties_below_line_votes:int[] = 0

voter

- name:string
- date_of_birth:string
- state:string
- voted:bool = false
+ votes:vote_ballot

- vote()

admin

- add_candidate()
- edit_candidate()
- delete_candidate()
- add_party()
- edit_party()
- delete_party()

party

- name:string
- candidates:candidate[]
- description:string
- votes:int = 0

get_vote_ballot() gets a vote_ballet that has been filled out by a voter and returns it.

calculate_votes() gets called first then calls get_vote_ballot with the returned value the votes are processed and they are passed to increase_votes() for the parties chosen.

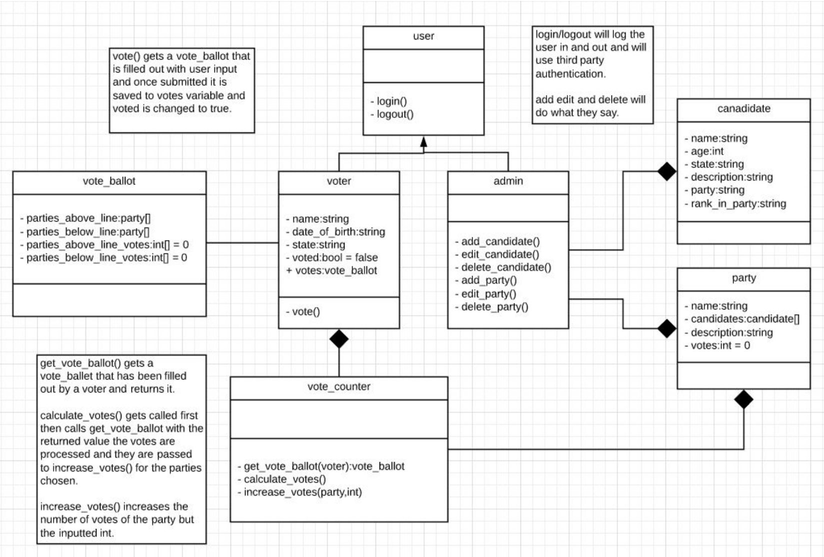
increase_votes() increases the number of votes of the party but the inputted int.

vote_counter

- get_vote_ballot(voter):vote_ballot
- calculate_votes()
- increase_votes(party,int)

Evolution of Class Diagram

- As time went on the class diagram was edited and improved to better represent the system.



vote() gets a vote_ballot that is filled out with user input and once submitted it is saved to votes variable and voted is changed to true.

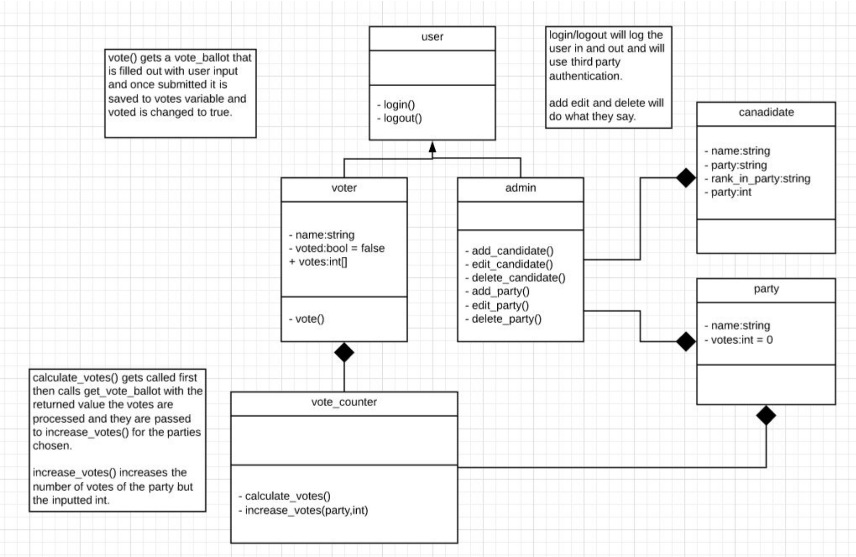
login/logout will log the user in and out and will use third party authentication.

add edit and delete will do what they say.

get_vote_ballot() gets a vote_ballet that has been filled out by a voter and returns it.

calculate_votes() gets called first then calls get_vote_ballot with the returned value the votes are processed and they are passed to increase_votes() for the parties chosen.

increase_votes() increases the number of votes of the party but the inputted int.



vote() gets a vote_ballot that is filled out with user input and once submitted it is saved to votes variable and voted is changed to true.

login/logout will log the user in and out and will use third party authentication.

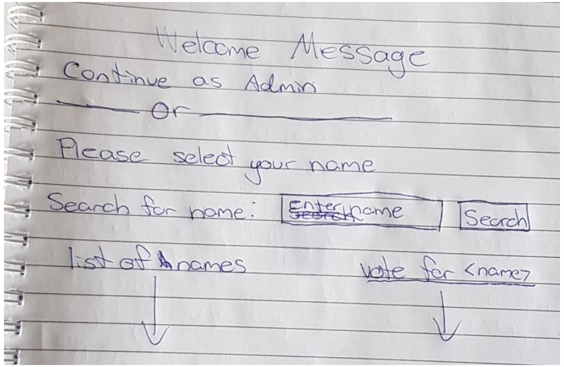
add edit and delete will do what they say.

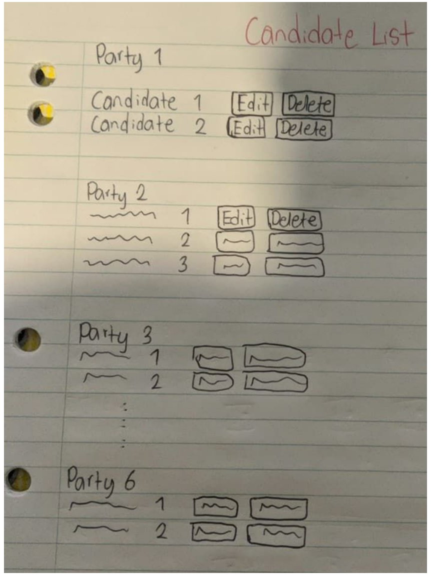
calculate_votes() gets called first then calls get_vote_ballot with the returned value the votes are processed and they are passed to increase_votes() for the parties chosen.

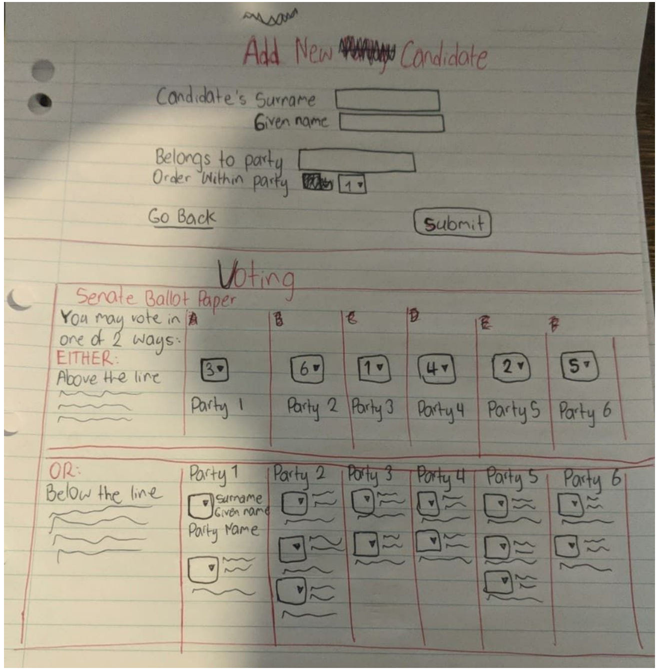
increase_votes() increases the number of votes of the party but the inputted int.

User Interface Design

- UI was designed before we started writing HTML and CSS







Process

1. Write cucumber test and Rspec base on design and requirement
2. Create a branch to implement the feature base on user stories and pass all test
3. Push the code to GitHub branch
4. Create new pull request
5. Group member does code review and discussion
6. Merge the branch to master by pull request

Project management - GitHub

Rule:

-Each new feature or code modify should create a new branch.

-Each commit linked to other artifacts such as issues or pull requests

-Code only merged by pull request with some discussion

-Avoid master rollback

Process that did not work

- Got duplicates requirements
- New features broke existing tests
- Don’t have enough model methods for rspec tests
- Not everyone reviewed all the code before a pull request was merged