Epic voting system report

Kane Wardle

Requirements completed and modifications made

(Step by step explanation for scenario of functionalities implemented/completed )

* Main screen has help option added, with input validation. If invalid system puts the user puts the user back to the start of the main menu so they can try to add a valid input again. (main menu/screen now has options to vote, go to the help screen, and login as admin)
* Help option is added, the reading of the help file is on the voting controller, the displaying is on the voting interface. Data is NOT HARD CODED but on a text file name “help.txt”.
  + Data on the text file is the title/ heading of the topic and the body, the numbering is done by the code.
  + The amount of array “lines” is written by the code itself, which means you can add or remove help data (headings and corresponding body of text) without any worry of breaking the help file reader (as long as there are no extra “!” added (what I used to separate my values, as I wanted to use commas in my body of text).
  + Appropriate try and catch statements exist.
    - If the amount of lines of data cannot be found, the program prints that they can be found.
    - If the help data cannot be set into its multi-dimensional array the program prints that it cannot.
  + User can only enter numbers that correlate to the help options they wish to view and “q” to quite back to the main menu. Any invalid inputs cause the program to notify the user and gets them to try again and to input a valid input until a valid input is entered.
* Staff class has been modified to store the staff’s password and their voting date timestamp.
  + Timestamp writes the date and the time done to milliseconds, as I was unsure whether I was to add a timestamp or a date stamp, so I choose the more accurate of the two.
  + Staff who haven’t voted yet their timestamp is written as “null” and stored in the program as a string
  + The default passwords for the staff have been set to the staffs first name and “123” e.g. Staff ID: 100 (Mike Long) his password is set to “mike123”.
* Candidate class has been modified to store information on the candidates department
  + Sample data has been added using the current six different departments.
* Before any staff can vote the administrator has to ADD a start date to the program. After the start date is add, staff can vote (\*as long as the current date is in the date range set by the admin)
  + If a start date is not added before staff try to vote, the program informs them that the admin needs to add the start date (doesn’t crash)
  + If the current date is out of the set date range it informs the voter what the current date is and date range they can vote in.
  + The voter cannot proceed to login in with their staff ID and password unless it is int the set time range
* Before the staff can vote the program checks to see if that staff ID has already voted if it has, the user is notified that they have already voted and put back to the main screen
  + Can quit out of the login process her and go back to the main screen, at this point.
  + If users enter the staff ID incorrectly more than three times they are notified, that they have entered the ID wrong to many times, then they are put back to the main screen
* Before staff can vote they are required to enter their personal password
  + If users enter the password incorrectly more than three times they are notified, that they have entered the password wrong to many times, then put back to the main screen
  + Can quit out of the login process her and go back to the main screen, at this point.
* When the ID is correct, the password is correct (both entered correctly within 3 attempts )and the current time is within the time/date range, which is NOT LIMITED to seven days, as the admin can change the date range, then the user can see all the candidates to vote for including what departments the candidates belong to. The user is then prompted to enter the code of the candidate they want to vote for
  + Can quit out of the login process her and go back to the main screen, at this point.
  + Before the voting is finalised the staff member has to confirm their vote by entering “y” any other input puts the staff back to the candidate list to select a different candidate or quit (resulting in voting for no-one allowing them to come back later and vote).
  + After staff finalise their vote their voting status is changed from not voted (0) to voted (1), the timestamp is created and the data is then saved to the staff.txt file. The candidate they voted for gets one vote added to their telly of votes on the candidates.txt file and the text file is saved. The Staff member is displayed a confirmation message and the program goes back to the main screen.
* At any point if the data cannot be save to the text files a message is displayed notify the user of which text file fails to save. This is handled by the voting controller.
* After the program starts and the user chooses to login as an administrator
  + The admin username is no longer case sensitive, but the password still is.
  + If the user fails to login after the third attempt in a row, they are sent back to the main screen
* After successfully logging in as and Admin the admin will see the NEW admin menu, with the options to:
  + To end voting
  + To view the voting results and stats (including the option to see which staff has/has not voted)
  + To view/ delete / add staff
  + To view/ delete / add candidates
  + To view/ change the days where users can vote (both the start day can change and the amount of days open can also be changed, end date is calculated from the two)
  + To quit, back to the main menu
* When the admin chooses to stop the voting the start date string is set back to null
  + If the staff try to vote after this they are informed that the admin needs to set a start date
* When the admin chooses to see the voting results the following is displayed
  + The voting stats Code, Name, Votes and (Vote%)
  + Numbers on voting list
  + Numbers voted
  + Then the option to see a list with the staff who have voted and a separate list of staff who haven’t voted (I have decided to split the list into two so it is faster for the admin to see who hasn’t voted and vis versa, this way the admin doesn’t have to comb through one big list to see who has and hasn’t voted when looking for one of the two options.
* When the admin chooses to view/add/ delete staff
  + They are displayed the list of staff (staff ID, name and password). I have chosen to show the admin the passwords for all staff for the purpose of reminding the potentially forgetful staff what there password is. (program has brief notice built in so that admins know why they are shown the passwords of the staff)
  + At the bottom of the list the admin is given the options to ADD or DELETE staff to/from the list
  + Adding staff has built in error checking to help maintain the validity of the staff text file.
    - ID can only be entered as a number, which is then checked to make sure the same ID isn’t already on the staff list
    - The staff name is not allowed any numbers or symbols in it (except “–“s so staff can have hyphened names. There has to be a space somewhere on the inside of the inputted string (cannot have the space before the start of the text or after the end of the text). The name also needs to be three or more characters long (one character for the first name, one for the space and one for the last name)
    - The pass word has to be grater or equal to three characters in length (including not being empty) and cannot contain any commas, as that would break the text file reader.
    - The voting status and the time stamp are set without input from the admin (set to 0 and null)
  + Deleting staff has built in error checking to help ensure that the correct staff is deleted
    - Input has to be a number and then the program checks to make sure that that ID actually exists.
    - If the ID exists then the program displays the name and ID of the Staff (chosen by the admin from the staff members ID) and askes the admin to confirm that they want to delete the staff member
    - Admin can quit out of the deletion process at any time
* When the admin chooses to view/add/ delete staff
  + They are displayed the list of candidates (candidate code, name and department)
  + At the bottom of the list the admin is given the options to ADD or DELETE candidates to/from the list
  + Adding staff has built in error checking to help maintain the validity of the staff text file.
    - Candidate code input has to be a number, then it is checked to see that it doesn’t already exist on the candidate list
    - The candidate name is not allowed any numbers or symbols in it (except “–“s so staff can have hyphened names. There has to be a space somewhere on the inside of the inputted string (cannot have the space before the start of the text or after the end of the text). The name also needs to be three or more characters long (one character for the first name, one for the space and one for the last name)
    - The candidates department cannot be empty or contain commas, so the file reader doesn’t come up with an error trying to read/ save the file
    - The amount of votes the candidate has is set to 0 by the program
  + Deleting candidates has built in error checking to help ensure that the correct candidate is deleted
    - Input has to be a number and then the program checks to make sure that that candidate code actually exists.
    - If the candidates code exists then the program displays the name, candidate code and the department of the candidate (chosen by the admin from the candidates code ) and askes the admin to confirm that they want to delete the staff member
    - Admin can quit out of the deletion process at any time
* When the admin chooses to view/ change the days where users can vote
  + The current time range is shown (start date, date range and the end date is calculated). If there isn’t a start date set, the program states that here isn’t a start date set
  + The admin is then asked if they want to change the current dates
    - Yes, allows the admin to change the start date of the days the ballot is open for
    - No, puts the admin back to the admin menu
  + Adding/changing the start day. This is done by getting the admin to enter the year, month and day separately to minimise errors when the program checks to make sure that it’s a valid date
    - Year has to be a number between 2013-3014
    - The month has to be in the correct format that matches a switch statement
    - The date/date has to be a number between 1-31
    - After all of these has been entered correctly the program checks to see if it’s a real date (cannot have dates line 2014-feb-31)
    - If it is valid the users is informed that it is saved, if the start date is the current date or a passed date the admin is informed, but the date is still saved
    - If the date turns out to be invalid, the user is put back to the start (asked for the year, month and day again)
  + Changing the amount of days the ballot can be open, this also has built in error checking
    - User can quit without changing the amount of days
    - The input has to be a number that is one or greater than one (cannot be zero)
* If the admin chooses to quit, the program reverts back to the main screen
* All inputs have error checking if the users input was not recognised/ incorrect the program asks for the input again (program as far as I know cannot crash due to the users input, if it does I would like to know how/ where it does)
* Fixed most of the spelling in the program
* Test plan is complete at row 380
* Code has been given a large amounts of comments so hopefully anyone can understand what it is doing in each part
* J-Unit testing has been started, but DOES require more work

# Conclusion

All requirements are met but j-unit testing requires some more work and the status report should have been smaller to match the brief’s recommended size.

* Staff class has been modified to store the staff’s password and their voting date timestamp.
* Candidate class has been modified to store information on the candidates department

The extra fields have been added to the ends, not in the middle of the already existing values, to make the modifications to the voting interface to read and save the data easier to write.

* Added help option to the main screen
* Remade the admin menu to suit the new needs of the admin menu
* Edited/ added the processes to login to both the admin menu and to vote, to match the new requirements