Criterion A: Planning

Scenario

My client is an accountant who works for Texas Electric Cooperatives (TEC) and would like an application that can calculate and search through membership dues for members of TEC. Currently, she uses excel for membership due calculations. However, she has cited various difficulties with calculating and searching for statistics across different sheets. Thus, she wants to aggregate all due calculations and membership information used for those calculations for the past 15 years into one easily accessible application.

The membership dues of each of the 75 members of TEC are assessed in equal proportions on three components: equal payment, the number of electric meters served, and revenue minus cost of power. As detailed in the initial interview notes, the client has provided the declining blocks and the relative weights of each member that are used to calculate each component of the membership dues per TEC's regulations.

I can obtain the data containing the number of meters, revenue, and cost of power for all 75 members of TEC for the past 15 years from my client. From this data, the client wants an application to calculate each member's yearly dues and allow users to search through the results. The application should have options for users to input membership data for the newest year and calculate the yearly membership dues once all 75 members have entered their information. She specified that this application is meant to be reused and updated annually.

Rationale

Based on the client's needs, a JavaFX application seems to be the most suitable. All functionalities requested by the client, which include calculating and searching through dues from past data files and from newly entered membership information, can be bundled into one JavaFX application. In addition, the client currently uses Excel for the majority of her work, and she specifically wanted an application for Desktop use that could be utilized simultaneously with Excel.

JavaFX allows for the creation of user-friendly interfaces. The integration process of back-end methods to the front-end user interface is also relatively straight-forward in a JavaFX application. A JavaFX application can allow for maximum functionality and understandability without being overly complex.

As the client has specified that the project must be able to be updated annually, the data file containing past membership information must be renewed as new membership information is added. With a JavaFX application, text files can be read in and written to. During the running of the application, the data file can be updated as new membership data is entered, and it will be saved for future launches of the application.

Java is platform-independent and can be run on nearly all operating systems, which will ensure that the application can retain its functionalities when launched on the client's operating system.

Success criteria

1. The product will calculate the dues based on equal assessment for each of the three components, and they should be allocated among the 75 members based on TEC's regulations.
2. The product will have an option for the user to either enter new membership information, specifically the number of meters, revenue, and cost of power for a TEC member within a given year.
3. The product will provide options for users to search up past due statistics based on a specified member and year as well as generate a full report for a selected year.
4. The product will provide options for users to search up the most recent due statistics based on a specified member as well as generate a full report for a selected year; the most recent due statistics will be updated annually once all 75 members have entered for new information.

5. The product will provide error and warning messages when the entered information does not match the expected data types or the entered information seems too unrealistic when compared to past data.
6. Newly entered membership information should be added to the data file and saved after the closing of the application
7. The product will consist of an interface with clearly labeled buttons, text fields, drop-down menus, and well-formatted text to optimize the user experience.

Word Count: 454