Criterion A: Planning

Defining the Problem

The client, Mrs.Conroy, is a mathematics teacher in charge of our Student Government (SG). When she has SG meetings, she needs to take attendance and notes whether or not they are late or excused on paper. If a member is either late/excused to over 70% of the meetings or two times consecutively, they are required to be on time to the next meeting or they will receive a warning or referral, depending on their reasons.

Lately, taking attendance has become a challenge due to her recent promotion as an IB teacher and an increase in SG members. This has caused her calculations to take longer and more likely to be error prone.

She has contacted me requesting an application that allows her to take attendance and see members who have pending disciplinary action, using an intuitive interface that streamlines the attendance process. This new system would be much faster and more efficient than her current paper notation method.

To help facilitate a user friendly, intuitive interface my father, a customer support manager, has agreed to be my advisor to assist in the development process.

Word Count: 186

Rationale for Solution

The idea behind creating a Student Government attendance manager is to minimize the time taken for Mrs.Conroy to enter data and see late students, while also increasing the accuracy of the calculations determining the need for disciplinary action. The GUI will need to be intuitive and user-friendly if the application is to be successful in streamlining the attendance process.

I am using Java as it is Objected-Orientated and I am familiar with the language, allowing me to take advantage of features providing robustness, organization, and security while also making the program easily expandable. With Java I am also able to create an effective GUI and database connection. Lastly the language is platform-independent, allowing any machine to run my program. This is especially useful as I am developing my program on Windows while my client uses a Mac.

I have chosen to use a database as my form of storing data as it is the easiest to keep track of multiple pieces of data in an organized, secure manner as well as being able to manage large amounts of data. As my client will be the only person accessing the application, my database will be offline and stored locally on the client's computer. Having the database offline is beneficial as a connection to the internet is not needed for the application to function.

I have used these tools to develop and design my product. I have also tested it thoroughly using my test plan and have implemented my product on my client's computer.

Word Count: 253

Criteria for Success

- Client will be able to modify Students, Dates, Excuses and Attendance
- Client will be able to take attendance through a GUI
- An easy to learn and use GUI
- Referral status will be calculated
- Client will be able to see all results in a spreadsheet
- Client will be able to see students categorized by their referrals (none, pending, on referral)
- Drop downs for dates and student names
- Remove students dynamically once attendance is taken
- Include a legend for attendance when you view the tables
- Allow client to see specific students only