Defining the Problem

      Mr. Cohen, a director of the Alumni Association at the Village School, was in need of an infrastructure to help kick start his new system to contact previous alumni who went to the Village School. With this system he would be able store useful Alumni information and leverage that information to boost initiatives such as the internship program and the scholarship fund. The system would allow him to better alert groups of Alumni about school events and solicit potential cooperation between the school and Alumni employers for potential internships for Village students.

      Mr. Cohen initially had some alumni information dispersed across multiple databases and multiple systems. In order to create the new system, we decided that it was going to need a single, central database that would hold alumni information. The database would need to hold contact information such as name, email, and telephone number, as well as other information such as employment, college, and whether or not their employer offered internships.

      Aswell, an easy to use method to insert new and legacy entries from a comma separated values (CSV) file into the database would be essential, as manually entering entries would be tedious and slow.

      The application would need to be able to query the database for usable information. An example of a query would be to find all students who graduated between 2010 and 2012.

      All of these functions would need to be available from an intuitive and organized graphical user interface, while also being designed to look in keeping with the design of the Village website.

Word Count: 259

Rationale for the Proposed Solution

The application's function is to streamline the access to alumni information, as such it will need a streamlined graphic user interface (GUI), that will be intuitive and easy to use. The application must also be able to facilitate bulk insertion of data into the database, as there are many alumni and it would take a prohibitive amount of time to enter each entry manually through the application.

I will be using Java, as it is the programing language I am the most familiar with. Java can also be used to query databases which is a primary purpose of the application. A database is a form of abstraction, where complex underlying processes are controlled through simplified interface. Because of this, using a database facilitates complex searching of data through queries that simplifies my job in storing and searching for alumni data. The language is versatile, and will be able to run on either mac or PC, which ever the administration decides to use. As Java is an object orient programming language, I will be allowed to structure the application to facilitate the use of objects, making my application easily expandable. Using an object oriented programming language allows me to use features such as inheritance. The usage of inheritance allows classes to inherit methods and datafields of the parent, allowing for faster development and reducing errors through reusability. Doing this allows the administration to build onto the product at a later date as the alumni, scholarship, and internship programs expand.

I have chosen to use a database structure to store alumni information. This allows for easier manipulation of data through SQL code than having to iterate through a flat file.

Word Count: 280