Jack Connly

Dr. Pablo Rivas

Software Development I

November 8, 2016

Software Development I Phonebook: Final Writeup

**Abstract**

This paper is an analysis of the Computer Science student phonebook. I plan on making my final project a computer science student phonebook for students to use that is categorized by a list with the Name on the left side, and email on the right.

**Introduction**

Hopefully, with this program Marist college can help the community of Computer Science majors help each other using teamwork, a large portion of today’s computer science jobs. In addition, this may help blend the relationship between lower and upperclassmen, even though this current version only holds information of our Software Development I class. This application will need to access the Marist Computer Science database and the individuals per grade. Instead of gaining access to degree works to learn the classes taken, I plan on just having the normal classes taken for freshman and sophomores. The app will produce a chart of students in your class with the student names going down and the email on the right. The program also enables the user to enter a new names and emails to the bottom of the matrix.

**System Description**

The system will first ask the user if they would like to start with a chart view of individuals in their current class. The chart displays the student’s name and email. Once you have looked at the matrix, the user is prompted to enter new names into the phonebook. The original matrix is [999,2] to avoid any overflow errors when users enter new names to the phonebook. Each time the user answers they enter “Y” or “y” for yes and “N” or “n” for no. If they want to add a name and email to the list you can then reprint the matrix and see your new entries at the bottom. User is also prompted with an exit program question at each output. If a name is entered it reprints their name, and if the student has given approval on being contacted it will also print the email. I used mainly if else statements with a few complex counters, and one important while loop. My initial matrix was created by using “String list[][] = new String[999][2];” and I added the names to the list going position by position. For example

list[0][0] = "Name";

list[0][1] = "Email";

were the matrix headers in the top [0][0] position “Name” and [0][1] position “Email”. In my if statements, I used the “.equalsIgnoreCase(“”)” as a parameter for the if statement and nextLine for the input so the program is able to handle any entry.

**Requirements**

There are no physical requirements for this program to run besides a computer with any operating system.

**Literature survey**

In context to phonebook applications there are tons out there. The most used one I can think of is the “Contacts” application used on the Iphone IOS system. At the application launch the software displays a list of contacts to scroll through with a search bar at the top. My program will be similar but with no telephone information. This program is unique because it displays a list of our current class, along with everyone’s Marist email. My hope would be to have this feature added to the tools section of the ilearn page.

UML Diagram

|  |
| --- |
| **Stack** |
| +remove(index: int): Boolean  -question(index: string): Boolean  +running : Boolean  +FirstLast() : string  +a : String  +I : int  +j : int  +b : String  +email() : string  +name() : string  +count : int |
| +ArrayList()  -Add(index:int, o; Object): void  + StringList[][] : String |

**Literary Survey & Requirements**

**Conclusion**

In conclusion, this paper covers the logistics of the Marist Student Phonebook or Foxbook. It is a useful software because in many computer science jobs today, teams are initiated to make problem solving more effective. There should be more of this type of Student to Student teaching at Marist college, and through this app, that can be achieved. In addition, it is a useful resource for class mates throughout the year being able to contact each other regarding assignments. If you build good relationships in the classes you can see what each person has taken, and plan your schedules accordingly to line up future endeavors.

Bibliography

No scholarly resources were accessed in the making of this project.