Igor Asipenka

Dr. Anewalt

Software Engineering

19 March 2019

Usability Study

1. Overview of System Functionality

The system our team is creating is a software that will allow Computer Science professors to randomly create groups with an optional set of various metrics that can be used to distribute students into certain groups (blacklist, gender, personality, etc.). The program will be built using Python 3. Furthermore, the software will take a CSV file as input, separate the values into individual student information, and ultimately group students together based on entered conditions. The client of the proposed Group Maker 3000 is Dr. Anewalt, who is a member of the University of Mary Washington's Computer Science department. She requested this project because she wanted a program to facilitate the creation of randomized groups for her classes. Additionally, she requested a system that could utilize specific parameters to create groups that could be personalized.

Due to the (assumed) high technical knowledge of the system's end users, the team decided to create a text-based user interface. All the UMW Computer Science professors would be able to correctly use an executable program, as long as the user interface is efficient and easy to use. As a result, the team decided on using a text-based menu that can take different paths depending on the options selected. Upon loading up the software, the user will be asked to enter a properly formatted CSV file from the folder the program is executed. Following this, they will be asked to choose if they would like their groups to be created based on the number of groups, or the number of students in each group. Subsequently, they will either pick the “Fully Randomized Option” which would randomly generate groups ignoring all parameters or select which conditions they would like to include in the generation process. These are all formatted as questions in order to minimize confusion with menu options. Following the end of the responses from the user, the software will display the groups to the user; if the groups are successfully validated by the user, an output CSV file with the finalized groups will be generated. Otherwise, the program will restart, and the user will need to resupply their conditions.

2. Imaginary Users

Dr. Frank Ferdinand is an Associate Professor at the University of Mary Washington. He has taught Computer Systems & Architecture every year for as long as he has been at UMW. For the end of the year project, his class is separated into groups to work together in creating a custom shell for Xv6, a simple Unix-like teaching operating system. He spends a lot of time creating truly random groups, but over the years he has also received numerous requests from students to implement a blacklist, where students pick a single student who they do not want to work with. This severely impacts his group making process, so he wants to use Group Maker 3000 to expedite this annoying task. He wants the software to be as easy as possible to use, with as much “hand-holding” as possible so that he doesn’t have to learn how to use it. Since he only wants one criterion for generating groups, the process is quick - which is exactly what Dr. Ferdinand is in search of.

Dr. Marco Gustav is an Adjunct Professor at the University of Mary Washington. He likes to emulate a real-life software development environment within his classroom, so for every major project he has allowed students to pick groups or has chosen them himself. However, Dr. Gustav has noticed that certain students always work with the same peers, and the students who do not pick their own groups also always end up working together. Dr. Gustav graduated Virginia Tech with a double major in Computer Science and Psychology, so he wishes to group students with a wide variety of personalities. The Group Maker 3000 software developed for CPSC professors has caught his eye because it would allow him to create randomized groups with students that have different personalities. Additionally, he wishes to use the blacklist functionality to prevent students from working with their lab partners. This software would allow him to efficiently create groups based on his personal settings, while also providing experimentation so he can apply his knowledge in human psychologies.

Dr. Pyotr Pistachio is a Tenured Associate Professor at the University of Mary Washington. He believes in individual learning and projects, so that students can properly establish beneficial study habits and knowledge. However, as the head of the Data Mavens (a data science club at UMW), he needs to create groups of club members for various competitions. The competition rules state that each group needs to have a maximum of three students, so Dr. Pistachio wants to group students in the following manner: beginner experience [1], intermediate experience [2], and expert experience [2]. This is evaluated by how many years they have been in the club, and how many competitions they have been in. The Group Maker 3000 software fits his needs, as he can select how many students should be in each group, and for each student to be different from the other students in the group.

3. User Model

The Group Maker 3000 is a software that will allow Computer Science professors to randomly create groups with an optional set of various conditions that can be used to distribute students into certain kinds of groups. Computer Science professors must regularly group students for group work and projects, so this software would allow them to be more efficient, while facilitating the tedious process of creating randomized groups. The goals of the GM-3000 is to streamline this process by allowing professors to select how many students will be in each group or how many groups should be created. Additionally, in the input file, professors will be able to include a blacklist, student personality, and other custom metrics that can be used to generate specially tailored groups. Groups can be specially created based on whether these metrics should be the same or different. Finally, the program will export the completed groups into an easy to read format.

Focus Functionality: Specific Grouping Metrics

Focus Group Individual 1:

1. **How will they interact with it?**
   a. Professors could make study groups for students or offer office hour sessions to groups that have poor skills.
2. **What should they be able to do?**
   a. Professors should be able to create groups for their classes easily and randomly.
3. **What should they not be able to do?**
   a. Users should not be able to change the information in the CSV file. They should not be able to change how the randomization works.
4. **What is important about the metrics?**
   a. Users should be able to customize the metrics to better suit their personal needs.

Focus Group Individual 2:

1. **How will they interact with the various metrics?**
    a. Users will assign students grouping metrics, like if they work well in a team or if they are leaders. Students could fill out a survey to decide these values.
2. **What should they be able to do?**
    a. Users should be able to adjust existing grouping metrics and add their own. An AI could be used to create a safety protocol to make sure the group members don't lean too far one way.
3. **What should they not be able to do?**
    a. Users should not be able to change the students' blacklists. Users should not be able to sort based on race.
4. **What is important about the metrics?**
    a. Users need to be able to omit metrics or have blank values in case the student doesn't want to answer

Focus Group Individual 3:

1. **How will they interact with the various metrics?**
    a. Users will be able to use the custom metrics to create more even groups throughout the class.
2. **What should they be able to do?**
    a. Users should be able to customize metrics.
3. **What should they not be able to do?**

   - **a.** Users should not be able to group all the students with beginner skills in one group.

4. **What is important about the metrics?**
   - **a.** The metrics allow professors to have more control about the group creation.

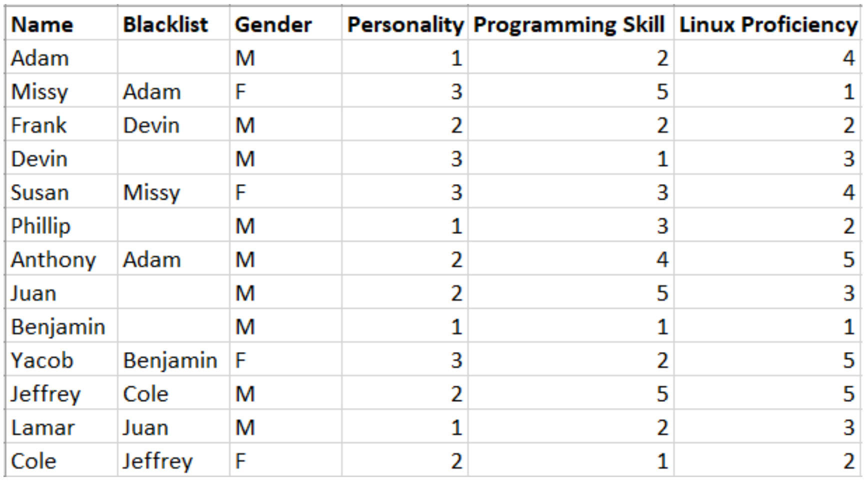
Focus Group Individual 4:

1. **How will they interact with it?**
   - **a.** Users would enter the types of metrics they want to focus on into the survey and students would then answer. I guess this would then be changed into a number.
2. **What should they be able to do?**
   - **a.** Users should be able to pick and choose which metrics they want to include. Maybe some groups can be made with one, while at another time a different value can be used.
3. **What should they not be able to do?**
   - **a.** Users should not be able to change the students blacklist.
4. **What is important about the metrics?**
   - **a.** Metrics seem really useful to create groups that seem to keep the student interests at heart.

The user of the software should be able to create custom metrics that they can then use to create specialty groups based on if these metrics should be the same or different within the groups. Professors should be professional in selecting what criteria should be included, and not edit the crucial student information. Also, the user should be responsible about which options they select so the groups are not imbalanced and implicit bias is limited. A survey will be used in order to properly create the input CSV file; this way students will be able to personalize their answers to the various metric questions.

4. Prototype Image/s

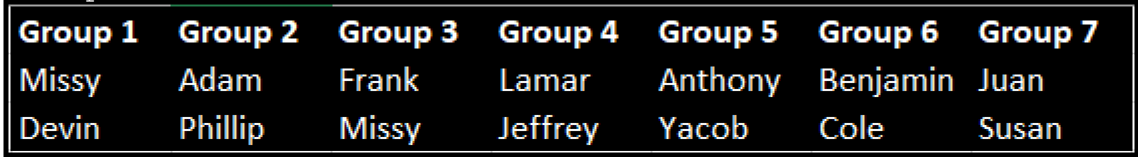
Mock Input File (CSV):



Name	Blacklist	Gender	Personality	Programming Skill	Linux Proficiency
Adam		M	1	2	4
Missy	Adam	F	3	5	1
Frank	Devin	M	2	2	2
Devin		M	3	1	3
Susan	Missy	F	3	3	4
Phillip		M	1	3	2
Anthony	Adam	M	2	4	5
Juan		M	2	5	3
Benjamin		M	1	1	1
Yacob	Benjamin	F	3	2	5
Jeffrey	Cole	M	2	5	5
Lamar	Juan	M	1	2	3
Cole	Jeffrey	F	2	1	2

Text-Based Menu:

```
Enter a filename:
______________
How would you like to have your groups created?
By number of students per group[N]?
Or
By how many groups will be made[G]
______________
How many groups will be made?
______________
Would you like the groups to be completely randomized? [Y/N]
______________
Would you like to use the blacklist? [Y/N]
______________
Would you like to create groups based on Gender? [Y/N]
______________
Should Gender be the same or different? [S/D]
______________
Would you like to create groups based on Personality? [Y/N]
______________
Should Personality be the same or different? [S/D]
______________
Would you like to create groups based on Programming Skill?
[Y/N]
______________
Should Programming Skill be the same or different? [S/D]
______________
Would you like to create groups based on Linux Proficiency? [Y/N]
______________
Should Linux Proficiency be the same or different? [S/D]
______________
Settings Established....
Generating Groups……………….
Generating Groups……………….
Groups have been randomly assigned.
Groups:
```



Group 1	Group 2	Group 3	Group 4	Group 5	Group 6	Group 7
Missy	Adam	Frank	Lamar	Anthony	Benjamin	Juan
Devin	Phillip	Missy	Jeffrey	Yacob	Cole	Susan

```
Are you satisfied with these groups? [Y/N]
______________
Exported Groups to GroupMakerOutput.csv
Exiting
```

5. Focus Group Feedback

Focus Group Individual 1 Comments:

- Very simple and convenient menu. Needs more color options. Easy to understand. I thought that sorting by gender could introduce some problems, what if people identify in a non-binary gender?

Focus Group Individual 2 Comments:

- The text-based menu is ugly; make it a graphical menu. Simple and easy to use. Very user-friendly language.

Focus Group Individual 3 Comments:

- Looks ugly, but fairly easy to understand. It used simple language and options are hard to mess up.

Focus Group Individual 4 Comments:

- It looks like those old 90's computers, but I think I understand how to work it. The questions made it really easy to understand.

There is a common theme that the design of the menu is not visually appealing. However, the overwhelming response about the content and word choice of the command line interface is that it is simple and easy to use/understand. All the focus group users knew exactly how to use the menu. This coincides with the assumption that CPSC professors will know how to use a command line interface. Additionally, the focus groups model matched the interface once the focus group was informed that it would be text-based. They were pleased that the menu was efficient and allowed them to select what metrics would be used in the randomization process. The individuals also liked how the menu was in question-answer form, and how it was obvious what letters to press.

6. Changes

I plan to discuss the priority of making a graphical user interface with the client since the majority of the replies in the focus group noted on the visual appearance of the software. Additionally, I believe that it would be a vital step to also discuss how professors will create the input CSV file due to the importance of the format of the various metrics. The overall opinions of the focus group were positive, and they liked the format of the command line interface.