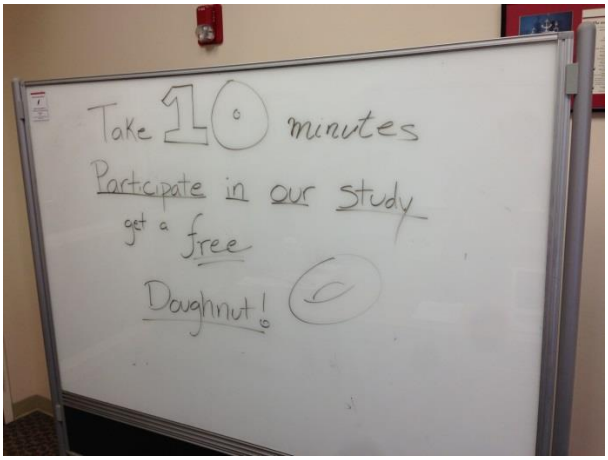
Brian Connerton
Dennis Law
Chris Selagea
Ethan Stewart

Group 8 Milestone 4

Part 1: Experimental Design and Procedure

Our study protocol began with us securing a mostly soundproof room in the Student Learning Center on campus. To advertise that we were conducting usability testing in the room we placed a large whiteboard near the room informing people that we were conducting testing and that participants would receive a free doughnut. Here is an image of the whiteboard advertisement:



When a participant came to the testing room they were then briefed about the project and what they would be doing from the following script:

> Hi, thank you so much for taking time out of your day to help us. You will be conducting usability testing for a program that we have designed. First, I'll give you some background on what we're doing here. We have identified a problem on this campus where many students are interested in finding internship and job opportunities, but they don't necessarily how to go about doing it. As it turns out, UGA offers a ton of great resources to help people out, but many students just don't know that they exist. Our solution to this problem was to make an online course for people to take where it lays out the ways UGA can help them and how they can help themselves. Afterwards, we have a quiz for them to take to make sure that they have retained the information presented to them. Now, people are probably not going to want to do something like this on their own, so we thought that a good way to compel people to take the course was to put a hold on their registration if it is not completed within a certain amount of time. Of course, the natural follow up to that is that we need some way to keep track of who has and has not done it and a way to manage the student database. That's where this comes in. You

will be using the administrator side of the system. It allows you to perform various operations on a student database to effectively manage the system.

After participants were briefed on the purpose of the system, we had 3 separate briefing where we explained which style of usability testing they would be completing. The 3 scripts used are as follows:

Benchmark Task Completion:

For this test we would like you to go through a list of common tasks that the system can perform. We will provide you with a list of tasks to perform and we just ask that you complete them to the best of your ability. Before we begin testing, you should know that we will be recording your onscreen actions while you participate in the test. All footage will only be used for our analysis later on and it will all be deleted when we have finished with our analysis. If you have any objections to this that is fine, however we can not proceed any further with testing. Otherwise, you will find the list of tasks next to the computer, and you may begin.

Thinkaloud Evaluation:

For this test we are going to do what we call a thinkaloud evaluation. What this means is that as you are using the system, we would like for you to speak aloud what you are doing and why you are doing it. We will provide you with a list of tasks to perform and we just ask that you complete them to the best of your ability. Before we begin testing, you should know that we will be recording your onscreen actions and audio while you participate in the test. All recordings will only be used for our analysis later on and it will all be deleted when we have finished with our analysis. If you have any objections to this that is fine, however we can not proceed any further with testing. Otherwise, you will find the list of tasks next to the computer, and you may begin.

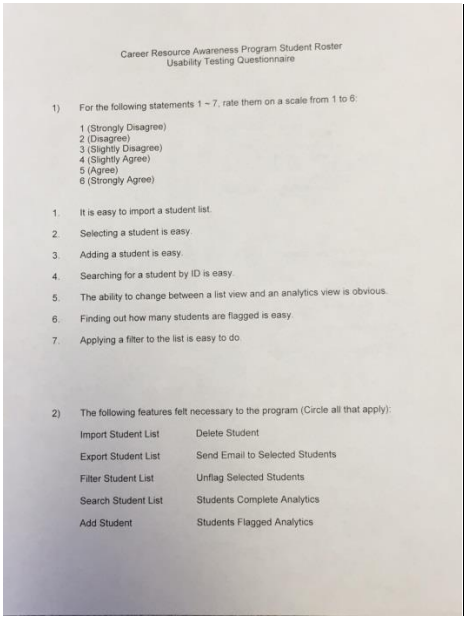
Retrospective Testing:

For this test we are going to do what we call a retrospective evaluation. What this means is that as you are using the system, we will be recording a video of your onscreen actions. We will provide you with a list of tasks to perform and we just ask that you complete them to the best of your ability. Once you have completed these tasks we will review the footage we have captured of your testing and ask you questions about what you did and why you did it. Before we begin testing, you should know that we will be recording your onscreen actions and audio while you participate in the test. All recordings will only be used for our analysis later on and it will all be deleted when we have finished with our analysis. If you have any objections to this that is fine, however we can not proceed any further with testing. Otherwise, you will find the list of tasks next to the computer, and you may begin.

After participants were briefed and gave consent, a tester began the recordings and presented the participant with a startup version of the program. The list of tasks that users were asked to perform is as follows:

1. Import “StudentRoster.txt”
2. Filter the list by “Flagged”
3. Unflag the first 5 students
4. Remove the filter
5. Search for a student with the ID number:
   810 924 674
6. Send an email to the student with that ID number
7. Add a new Student with the following information:
   Last: Doe
   First: John
   ID: 810 246 802
   Email: jdoe@uga.edu
   Hours: 56
8. Unflag the student you just added
9. Delete the student you just created
10. Find the number of students who are flagged
11. Export the updated list as “Updated List”
12. Close the program

Once a participant completed testing one of the testers ended the recording. We then asked the participant to complete a survey about their experience. The following is an image of the front and back of the sheet that the survey was on.



Career Resource Awareness Program Student Roster
Usability Testing Questionnaire

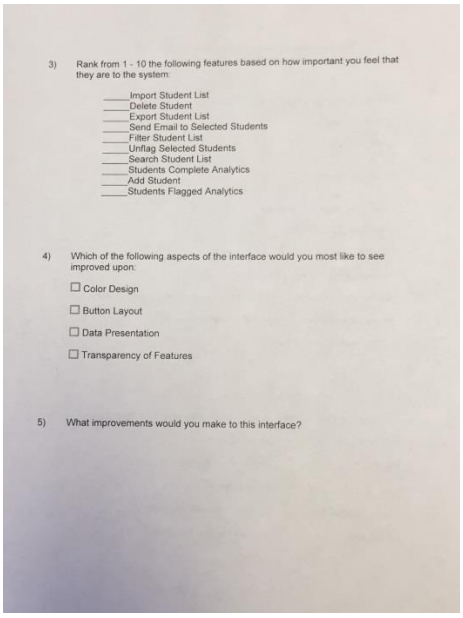
1) For the following statements 1 ~ 7, rate them on a scale from 1 to 6:

1 (Strongly Disagree)
2 (Disagree)
3 (Slightly Disagree)
4 (Slightly Agree)
5 (Agree)
6 (Strongly Agree)

1. It is easy to import a student list.
2. Selecting a student is easy.
3. Adding a student is easy.
4. Searching for a student by ID is easy.
5. The ability to change between a list view and an analytics view is obvious.
6. Finding out how many students are flagged is easy.
7. Applying a filter to the list is easy to do.

2) The following features felt necessary to the program (Circle all that apply):

Import Student List	Delete Student
Export Student List	Send Email to Selected Students
Filter Student List	Unflag Selected Students
Search Student List	Students Complete Analytics
Add Student	Students Flagged Analytics



3) Rank from 1 - 10 the following features based on how important you feel that they are to the system:

_____Import Student List
_____Delete Student
_____Export Student List
_____Send Email to Selected Students
_____Filter Student List
_____Unflag Selected Students
_____Search Student List
_____Students Complete Analytics
_____Add Student
_____Students Flagged Analytics

4) Which of the following aspects of the interface would you most like to see improved upon:

☐ Color Design
☐ Button Layout
☐ Data Presentation
☐ Transparency of Features

5) What improvements would you make to this interface?

Once the participant completed the survey they were thanked for their time and given a doughnut on the way out.

Testing was administered on a group member's laptop in a soundproof room in the SLC. The program was run with a cleared desktop background on screen. QuickTime was used to record the screen video and audio on the laptop. No additional equipment was used to record. There was only one participant in the testing room at a time so as to avoid any distractions. Participants were given a mouse to use while they completed the testing.

Our Heuristic Evaluation was conducted by having experts in design (other HCI students) explore the system without the guidelines we set forth for the other types of testing. The experts were provided with a list of the heuristics and a blank sheet of paper. We then asked them to record any usability issues they noted and assign a severity rating to each.

Part 2: User Demographics

The user demographics for our usability study may be a point of contention when analyzing our results. The target demographic for the system is UGA administrators and faculty members. With that being said, it would be very difficult for us to convince 15 different administrators to participate in usability testing for our project. Instead we used typical UGA students to conduct our testing. While we recognize that this is not a true representation of our target user base, we believe that the results found in our testing still reflect the good and bad elements of our project in terms of raw usability. It is worth noting that for our heuristic evaluations we exclusively used HCI students to conduct testing in an effort to reflect using "experts" in design.

Part 3: Experimental Design Justifications

There are two different sets of tasks that we had users complete. The first is the task list set forth for the benchmark/thinkaloud/retrospective testing sessions. The list is shown above in Part 1 of this document. We chose those tasks because we felt that they accurately encompassed every action that a user could take within the system. Even for functionality such as filtering, where the user can filter by different criteria, we felt that only one filter was necessary to accurately evaluate the effectiveness and usability of the feature. Since our system has a fairly limited amount of functionality, we approached the task list with a "no stone left unturned" mindset.

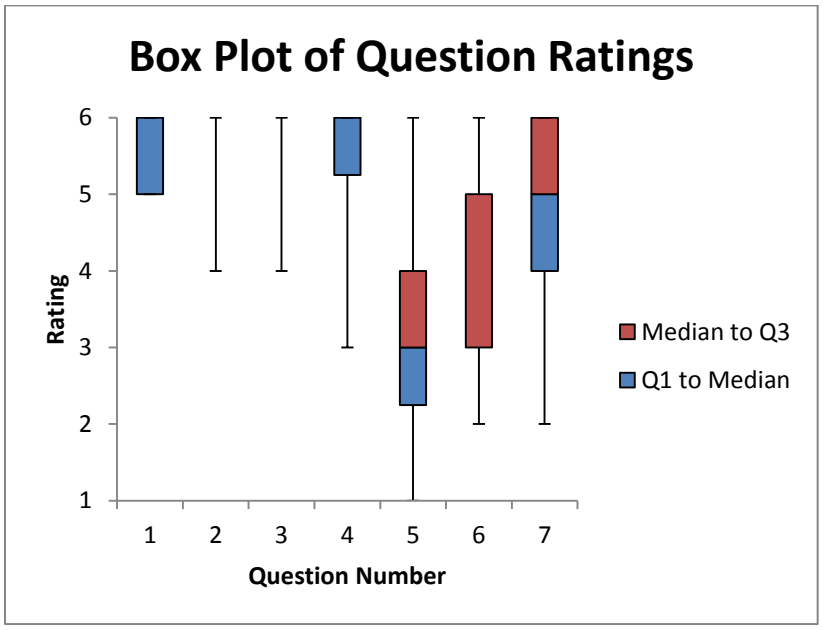
Our second task set was for the heuristic evaluation, however task set may not be the appropriate term. Instead, we explained to the participant what the purpose of the system is and the basic idea behind it. We then had them do an exploratory test of the system and make note of any usability errors they encountered during testing. All participants in the study were given the questionnaire when they finished.

Part 4: Results and Analysis

****Testing is currently incomplete and all analysis/results are subject to change**

After conducting usability testing, some very important issues became clear. One of the biggest problems that we identified was people not recognizing that certain functionality was provided to them, especially in the heuristic evaluations, which had less guidance than the others. Notably, many participants were not even aware that the analytics section existed, as the icons to switch to that view were note clearly marked as such. This was most clearly expressed to us in the first section of the post-testing questionnaires. In this section we provided participants with a series of statements and asked them to express their agreement with said statements with a number from 1 to 6. The statements we used are as follows:

1. It is easy to import a student list.
2. Selecting a student is easy.
3. Adding a student is easy.
4. Searching for a student by ID is easy.
5. The ability to change between a list view and an analytics view is obvious.
6. Finding out how many students are flagged is easy.
7. Applying a filter to the list is easy to do.

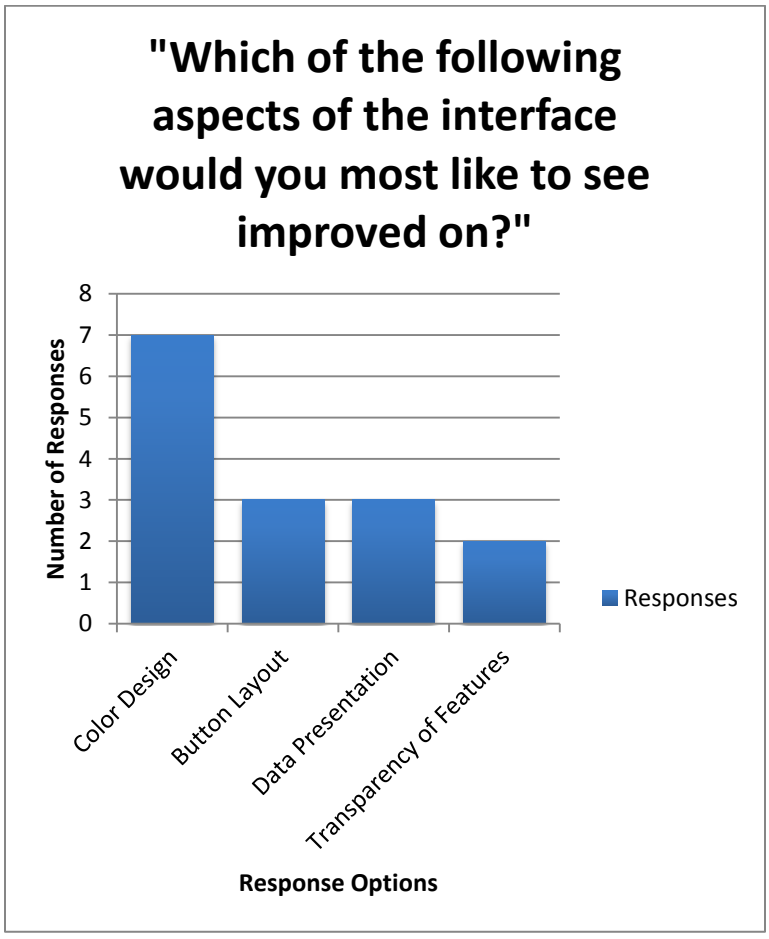


The above graph represents the rating that people gave I response to the aforementioned statements. Questions 5 and 6, the questions relating to the analytics of the list, clearly received much lower scores than the other questions. Between this and the free response section on the questionnaire, this was probably the complaint that we heard the most. On a related note, participants also noted that the filter option was not immediately apparent to them. Given a second cycle of development, the lack of transparency of functionality would be the first issue that we address. Some examples of fixes we

would need to implement are adding labels to the buttons that switch between list and analytics view. Another fix would be to give more indication that the icons are actually buttons. As it is now, there is no indication, whether it is a change in color of the buttons or a change in the cursor, that the icons can be pressed. This just adds another level of obfuscation of functionality.

Another response we frequently received was that the design of the program left much to be desired. On another section of our questionnaire we asked which element of the interface users would most like to see improved. We then provided them with a list of features that we identified as the weaker areas of our program. The four areas we defined were: Color design, Button Layout, Data Presentation, and Transparency of Features. The results from this section are shown below.

Improvement	Responses
Color Design	7
Button Layout	3
Data Presentation	3
Transparency of Features	2



This chart clearly shows that our design was a big problem for users. We did expect this to an extent before we began testing. Most of our effort was put into functionality and usability over aesthetic design. Given more time and another development cycle our efforts would turn to making the program better looking. We are not sure exactly how we would improve upon colors and design at this time, but it would become our main focus now that the program is working well and we have been able to identify specific usability issue to address.

The aforementioned problems were to biggest problems we identified during testing across all participants. Other than that, we uncovered a couple bugs in the program during testing such as integer overflow issues and incorrect file importing problems. These problems were quickly and easily fixed as it only required us to add in simple error prevention checks to the code. We also received some good feedback about possible fixes to make the program more user friendly, such as adding in global counts to the list and allowing for users to undo actions like delete.

In conclusion, the overall responses we received from testing were largely positive. People seemed to be very impressed with the functionality of the system and the general ease of use, barring the aforementioned issues with the analytics section. The main areas of issue that we identified were transparency of certain features and the aesthetic design of the program. Given another lifecycle of development we would devote the majority of our effort towards making filters and analytics more obvious, as well as giving the program a more pleasing skin and color design. We actually received on remark on our free response in the questionnaire that the design of the program reminded him of OASIS. If anything tells us that we need to reskin the program, it's that. As mentioned at the beginning of this section, testing is not yet complete, so current results are subject to change, and new, as yet unseen problems may arise. Any new issues will be reflected in subsequent testing reports.

****Included below are full sized images of the questionnaire given to participants after testing**

Career Resource Awareness Program Student Roster
Usability Testing Questionnaire

1) For the following statements 1 ~ 7, rate them on a scale from 1 to 6:

1 (Strongly Disagree)
2 (Disagree)
3 (Slightly Disagree)
4 (Slightly Agree)
5 (Agree)
6 (Strongly Agree)

1. It is easy to import a student list.
2. Selecting a student is easy.
3. Adding a student is easy.
4. Searching for a student by ID is easy.
5. The ability to change between a list view and an analytics view is obvious.
6. Finding out how many students are flagged is easy.
7. Applying a filter to the list is easy to do.

2) The following features felt necessary to the program (Circle all that apply):

Import Student List	Delete Student
Export Student List	Send Email to Selected Students
Filter Student List	Unflag Selected Students
Search Student List	Students Complete Analytics
Add Student	Students Flagged Analytics

3) Rank from 1 - 10 the following features based on how important you feel that they are to the system:

_____Import Student List
_____Delete Student
_____Export Student List
_____Send Email to Selected Students
_____Filter Student List
_____Unflag Selected Students
_____Search Student List
_____Students Complete Analytics
_____Add Student
_____Students Flagged Analytics

4) Which of the following aspects of the interface would you most like to see improved upon:

☐ Color Design

☐ Button Layout

☐ Data Presentation

☐ Transparency of Features

5) What improvements would you make to this interface?