USABILITY SIZE N

Except where reference is made to the work of others, the work described in this thesis is my own or was done in collaboration with my advisory committee. This thesis does not include proprietary or classified information.

Andrea E. Williams

Certificate of Approval:

Cheryl Seals
Associate Professor
Computer Science and Software Engineering

Juan Gilbert, Chair
Associate Professor
Computer Science and Software Engineering

Peter Grandjean
Associate Professor
Health & Human Performance

George T. Flowers
Interim Dean
Graduate School

USABILITY SIZE N

Andrea E. Williams

A Thesis

Submitted to

the Graduate Faculty of

Auburn University

in Partial Fulfillment of the

Requirements for the

Degree of

Master of Science

Auburn, Alabama
August 4, 2007

USABILITY SIZE N

Andrea E. Williams

____
Signature of Author

____
Date of Graduation

VITA

Andrea E. Williams, daughter of James and Erma Williams was born on February 24, 1983 in Ft. Meyers, FL. She graduated from Columbus High School with honors in 2001. She attended Spelman College in the fall of 2001 and received a Bachelor of Science in Computer Science in May 2005. The following fall she attended Auburn University where she is currently working towards her Ph.D.

THESIS ABSTRACT

USABILITY SIZE N

Andrea E. Williams

Master of Science, August 4, 2007
(B.S., Spelman College, May 2005)

41Typed Pages

Directed by Juan E. Gilbert

In today's software development environment building a usable, customer satisfactory product is key to the success of a business. User satisfaction and usefulness are measured using usability studies which involve potential customers. During the politics of software development and delivery however, having to conduct usability studies can become a costly expense in the overall budget. This can cause problems because some managers would simply fall back on heuristic evaluations which are significantly cheaper using the developer as a tester and leaving out the real user, the customer. By using Applications Quest, a data mining clustering tool, we would like to see if given a population of size N is there a subset of N that would yield the same results as the larger population. If a company could use a smaller subset of N and get the same results, they could possibly stay on budget, on schedule, and save money.

ACKNOWLEDGEMENTS

First and foremost, I would like to thank the Lord above because it is through him I am able to do all things. He has guided me through life’s winding roads and I am still here. I would like to thank Dr. Juan Gilbert for all his support, encouragement, and patience. At times when I did not know how I was going to make it, he was a guide to the light. I would also like to thank my committee members for their support and words of wisdom. Their efforts in advising me and reviewing my work are forever appreciated. Thank you to my family for their love, support, and encouragement. They have always told me to follow my dreams and every time I have they have been right there behind me believing and pushing. Last but not least I would like to say thank you to my fiancé Justin. There were times when homework had to come first and he was right there with me googling for answers. At times when I felt alone, he showed up with treats in hand from Conyers, GA. Thank you so much!

Style manual or journal used: Journal of SAMPE

Computer software used: Microsoft Word 2003

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

1. INTRODUCTION

In the software development cycle, it is often the practice that developers will hold usability studies to test the accuracy and effectiveness of the software and to retrieve user response as to the satisfaction or usability of that software. In practice, usability studies can give developers insight into the mind of the user as well as unveil errors, major and minor, within the system. Of course, as with anything that involves users and studies, planning and budgeting to assess the cost of usability testing and users in the study must be done. Planning studies can be time consuming because activities such as designing studies, enlisting participants, and possibly implementing several runs of a study must take place. In planning, developers must consider different methods of usability, heuristic evaluation, and observation of tasks done in the study; these activities can become burdensome and intimidating to companies not familiar with this practice or not sure which practices will benefit their company most. Budgeting for studies within the development cycle is often a tug of war because although several tests might prove beneficial in the long run of the project scheme, in the short term the budget might not allow for testing at all or it might allow for a single test with a select number of participants. Often there are numerous problems with planning and budgeting that ultimately cause studies to either be drastically cut down in size or eliminated altogether.

Determining the best factors for a study can be problematic because studies should be designed to fit the particular company, its size, and its goal for the study. Some companies are not familiar enough with design specifications and often have to hire someone to implement their study or they neglect it altogether. In some cases they even implement their own study. In all cases, the outcomes can become costly if proper judgment is not used in selecting the type of study, the number of participants, the type of participants, or even the number of runs (trials) needed for that particular study.

1.1 PROBLEM DEFINITION

The aim of this study is to show that using software, ApplicationsQuest, one aspect of designing a study, namely the selection of participants can be done effectively to 1) reduce costs and 2) maintain or improve result quality. The experiment conducted compares the results for two groups, those randomly selected and those chosen by ApplicationsQuest, and evaluates the significant difference of one group over the other. The rest of this thesis is organized as follows: Chapter 1 introduces the problem definition while Chapter 2 examines past and current methods for selecting the number of participants as well as which participants are chosen from the population of users. Chapter 3 gives a thorough description of ApplicationsQuest and how it is used in the context of this experiment. Chapter 4 discusses the design and results analysis and Chapter 5 presents conclusions and ideas for future work.

2. LITERATURE REVIEW

In usability design when choosing the number of users for a study there is a debate about the number of participants to use; there is the five-user assumption, that says five users is all you need in a study, and then there is the idea that five users is not nearly enough because five users will not provide enough feedback about a product. Which idea is correct? In actuality there are a number of theories that claim to know the number of users for a study each saying that that number of users will provide a large percentage of accuracy and return most of the defects in the product.

2.1 FIVE-USER ASSUMPTION

As discussed previously, usability studies can become expensive when it comes to designing and selecting users. Nielsen says, “The best users come from testing no more than 5 users and running as many small tests as you can afford.” [2] According to the formula, Problems found (i) = $N(1-(1-?)^{i})$ represented graphically below, one user should be able to uncover a third of the findings and as more users are added, redundancy occurs in the information.

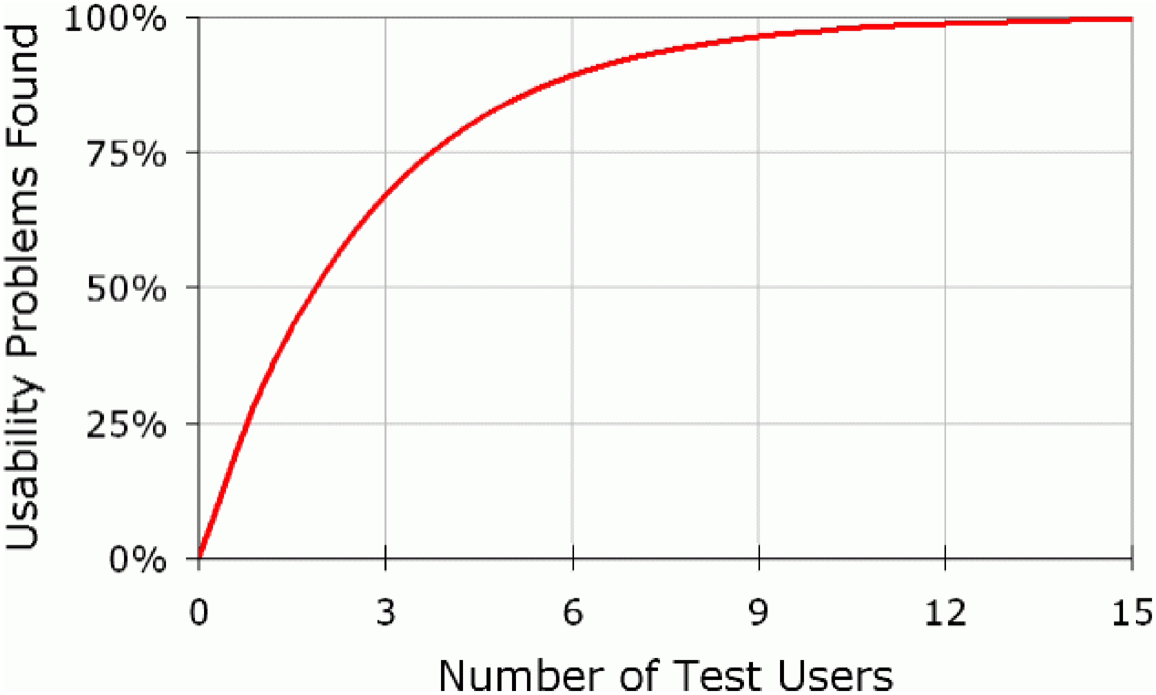

Figure 1: Curve showing relationship between problems found and number of users

Nielsen's study showed that a group of five users were able to find about 80% of the findings in a system and as more users were added, there was less information to be found but more and more money was being spent to run tests and compensate additional users. The idea behind the assumption is that you can learn more from a group of five completing multiple tests than you would on fifteen participants completing one test. The study would yield more results and cost the same or less than the study with fifteen participants. Many usability professionals, because of this study done by Nielsen, only use four to five users in their study. [2]

2.2 FIVE USERS AND BEYOND

Upon using the five-user assumption many usability professionals have found that five users are not enough. One study was done where five users were randomly chose and only uncovered 35% of the findings while the 13th and 18th user uncovered results that the original missed. This shows that if the study had been discontinued at five those results would have been overlooked. In this study, users 6-18 were able to find other new results that the original five were unable to find which goes to show that if the right users are not chosen pertinent results can be left out. [1] In attempts to describe the confines of the five-user assumption, many professionals neglected the rest of the assumption that recommends running the subjects until the findings meet an “acceptable level,” and instead adopted the most minimal number particularly five. To further examine the theory of five not being enough, Nielsen conducted another more structured study that took a population of sixty and randomly selected multiple groups of five or more. Each group’s findings were then compared against the findings of the entire population to measure how each group’s size affected data reliability, confidence and usability issues. The average percentage of findings by 100 trials of groups of five was 85% while the average percentage for any random group of five was 55-100%. Adding users increased the percentages, but the most important result showed that 55% was the minimum percentage for a group of five while a group of twenty produced a minimum percentage of 95%. [1]

2.3 RANDOM SAMPLING

Random sampling is another method often used in usability studies. When properly done, random sampling contains no bias and can be relatively representative of the targeted population. [8] This method is also used because it requires no prior research or skill in selecting participants and is less expensive. Random sampling allows researchers to make generalizations about the majority of the population and those claims can be justified by a certain level of certainty. [9] Of course as with any choice made surrounding a usability study and selecting users, companies must choose methods that most benefit their budget and the goal of their study. Samples are chosen in different ways such as simple random, systematic, weighted (quota), or convenience selection. In the case of simple random selection, participants are chosen from the entire group by the random selection of a unique identifier which can be drawn by hand like a tag drawn from a hat or mathematically selected by a computer program. Systematic selection is used by dividing the population into partitions and from each partition randomly selecting a participant. In some studies, particularly web based studies where companies are trying to target a specific user group; usability professionals give weight to that particular group so that they ensure their presence in the sample group. A convenience selection is simply as it sounds, the researcher randomly chooses participants that can be conveniently found. The participants may or may not be representative of the targeted group at all. Study results have demonstrated that random sampling can be problematic because you can never be 100% certain that the results from the selected sample are representative of the entire population. [8] Random sampling can also give you a false sense of security because in some usability studies the goal is not to find significant difference but more so to find insight into the usefulness of a particular product.

2.4 HOMOGENEOUS VS. HETEROGENEOUS POPULATIONS

In the article “Eight Users Is Not Enough,” authors Perfetti and Landesman found after trying to complete testing on an e-commerce site that the recommendations of four to five users with no more than eight was not enough. The first five users alone only yielded 35% of the problems in the system; at that rate it would take them 90 tests to uncover the 600 problems in their system. The problem found with this study was that the usability professionals tried to apply a concept that did not quite fit their needs. E-commerce websites contain much more complexity in content versus software and simple websites, continuously and incrementally, change whereas software only changes in between a version which does not occur as frequently. With that discovery they also found that their users varied just as much as the complexity in their system. Their results showed that a sample group could not be used as a representation of the whole because each user that interacted with the system used the system differently. Understanding the type of product they were testing and their users, the authors were able to successfully learn what worked for their system. [5]

2.5 CONCLUSION

When choosing the "right" participants it is imperative that the users be representative of the population your product is trying to solicit. [3] As a sample of the entire group, gathering the relevant demographic information can prove to be helpful in differentiating between the results of individuals in the group. [4] Recruiting these representative participants is yet another timely and costly activity that creates an intimidation factor for potential usability professionals. Most professionals agree that testing should be done but some companies just do not have the capability or experience necessary to pull off small tests let alone multiple tests involving users within deadlines set for the project. On average, it is said to cost $107/user in a study depending on location and profession and that's without a recruiting agency's help. Companies who use recruiting agencies must add additional fees while other companies must spend approximately 1.15 hours per person recruited man hours recruiting. [6] Even after choosing the “right” participants it is important for practitioners to understand that there are variables within a study that they have varied control over. The types of participants a usability professional can find, the mission criticality of a system, or usability issues found posing a problem to a system have a deep impact on the number of users a study needs to still obtain accurate results. All things considered, there is still no dry cut way to select the number of participants to use in studies nor is there a way to select which user should be used or is most representative. A method that could help usability professionals minimize costs and test group sizes as well as maximize results would have a significant impact in usability design.

3. USING APPLICATIONS QUEST AS AN APPROACH TO FINDING N SIZE

3.1 WHAT IS APPLICATIONS QUEST?

"Applications Quest is data mining software that clusters admission applications based on holistic comparisons." [7] The idea behind this software came from two landmark court cases, Grutter vs. Bollinger and Gratz vs. Bollinger, where two students challenged the University of Michigan's admissions policies. Because of these cases the Supreme Court ruled that diversity could be used in admissions policies, but race could not be the determining factor for admission. It was determined that applicants' applications should be reviewed holistically and not based on a single attribute such as race or ethnicity. [7] The notion of holistically reviewing an application means considering each and every attribute of the application such that no single attribute weighs heavier than another. For admissions committees the action of holistically reviewing an application is time consuming and difficult because humans do not possess the ability to effectively compare attributes subjectively and with reproducibility of results. Applications Quest achieves the goal of holistically comparing applications and recommending applicants that represent diversity with diversity not being defined by race or ethnicity. Because the algorithm compares each application with the same rigor, the results are reproducible and justifiable. [7]

3.2 HOW IT WORKS?

Incorporating computer science and information retrieval clustering algorithms, Applications Quest holistically compares thousands of applications one to another and places them in groups or clusters based on their holistic similarity. [7] The algorithm uses attribute-value pairs to compare each application, the more values each application has in common determines its placement in a cluster, this means that similar applications appear in the same cluster. With diversity in mind, each cluster is designed to hold similar applications but from each cluster the most different applicant is chosen.

3.3 DIVISIVE CLUSTERING: A POSSIBLE SOLUTION TO OUR PROBLEM

Employing a divisive clustering algorithm, Applications Quest recommends applicants that are representative of diversity within an admissions applicant pool. Using this same software but modifying the context in which it is used, namely for participant selection in usability studies, could possibly help usability professionals select the most representative users of their targeted population. With the most representative test users selected by Applications Quest, usability professionals can save money and time on recruiting and weeding out studies completed by outliers in the group. The idea is that users selected by Applications Quest will yield the same, if not better, results as the entire population of potential test users. Applications Quest would pose a solution that has a minimal cost, reproducible recommendations, and quality results.

4. EXPERIMENT

An experiment was conducted to determine if given a usability group size N, Applications Quest could select a subset of users whose study results would be representative of the population. To determine if the group was representative, it was necessary that their results prove insignificantly different than those of the majority population.

4.1 EXPERIMENT DESIGN

4.1.1 DATA

The data for this experiment was selected from a previous study done in the Human Centered Computing Lab at Auburn University. The seventy-two users in the study represented students from Science, Technology, Engineering and Mathematics (STEM) majors. Their ages ranged from 19-30 years old. Of the seventy-two participants seventy spoke English as their native language while the other two spoke English as their second language. There were twenty-one females and fifty-one males. In the study where this information was collected users' demographic data was collected in pre-surveys and their answers to the questionnaire about the software they used were in post-surveys.

4.1.2 MATERIALS

This experiment's results were stored and manipulated in Microsoft Access and Microsoft Excel.

4.1.3 PROCEDURE

The data discussed above was imported into an Access database. To clean the data, participants' were filtered to make sure that pre- surveys had an accompanying post- survey. This experiment was conducted in two parts with the second part being done two different ways: 1) randoml y selected participants were chosen from a group of seventy-two users. Each participant was given a unique identifier from the original study, that identifier was used in this study as well to maintain their anonymity. To select users in this approach, a program was written to randomly select groups of participants using the time divided by the total population size as the seed. For each group size chosen, five trials were run. The group sizes selected was 5, 7, 13, 15, and 20. For each group size, random trials were run five times meaning that for each trial new participants were randomly chosen. Each random participants' answers to questions selected from the questionnaire were queried and placed in an excel spreadsheet where they could be tested for any significant difference from the entire population. The attributes chosen, wonderful---terrible, frustrating---satisfying, usable---not usable, and this medium was easy for me to use, from the questionnaire were based on a 5 point likert scale. Significant difference was tested on each group using Microsoft Excel's formula for the t-test. The t-test is an analysis tool that tests for equality of a population for each underlying sample. The t-test can employ three different assumptions but for this experiment the two-sample unequal variance assumption will be utilized. A two-sample unequal variance means it is assumed that the two samples used have come from distributions with unequal variance and is used to determine whether the distributions have equal population means. Once calculated, the results of the t-test were analyzed to see if the randomly selected group could be considered representative of the population. 2a) All pre-survey demographic data was loaded into a database and run by Applications Quest. Applications Quest was given a specified number of clusters to return and from those clusters it chose the most representative person of each cluster. Once the participants were chosen their answers to questions selected from the questionnaire were queried and placed in an excel spreadsheet where they could be tested for any significant difference from the entire population. The same attributes for the first part of this experiment were employed here as well. Significant difference was again tested with the t-test and the results analyzed for comparison. 2b) The same data loaded into the database for part 2a was used to run Applications Quest again. The algorithm however for part 2b was changed to select the most different person from each cluster. In the case where a cluster contained only two participants, Applications Quest would select the participant most different from the entire population. Again the same attributes were used for querying and results were tested and analyzed using the t-test to determine significant difference.

4.2 RESULTS

Once the statistical analysis tools had been applied as described in both approaches to each group size chosen for this experiment, the results were as follows:

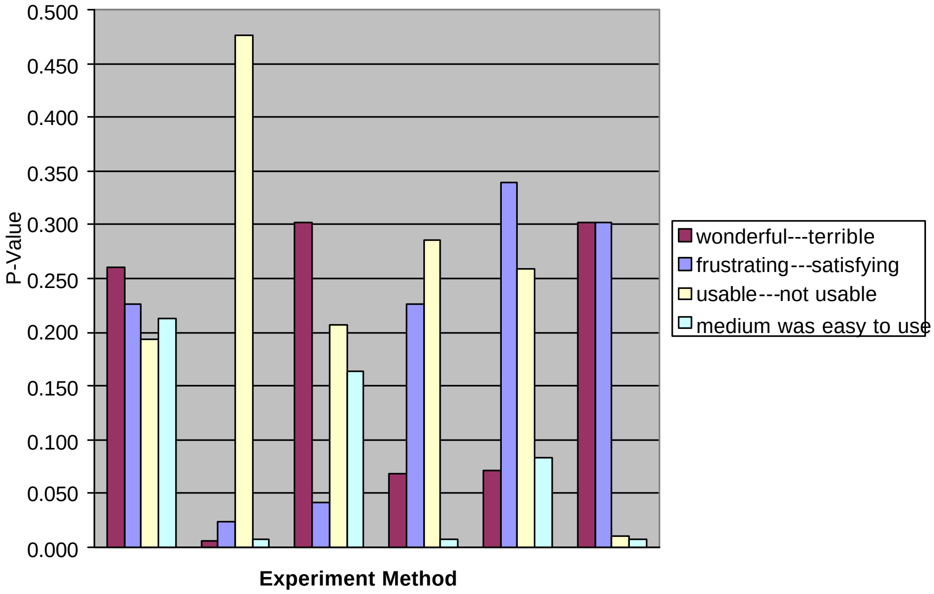

Figure 2: Comparison of statistical differences for group size 5: Random Trial vs. Applications Quest

As seen in the graph above, the random trials for group size five were very inconsistent; through each trial the results varied considerably from the trial before. In Figure 2, 60 percent of the random trials for group size five were found significantly different for the attribute wonderful---terrible, meaning that their p-value was below .10 or did not meet the 90 percent confidence level set as acceptable for this experiment. This suggests that a usability professional has a 40 percent chance of randomly selecting five participants representative of the targeted population. For the attributes wonderful---terrible and frustrating---satisfying in this table, Applications performed better than four of the five random trials. On the fifth random trial it was equal to wonderful---terrible and it was slightly behind frustrating---satisfying. As you'll continue to see Applications Quest maintains its accuracy and confidence.

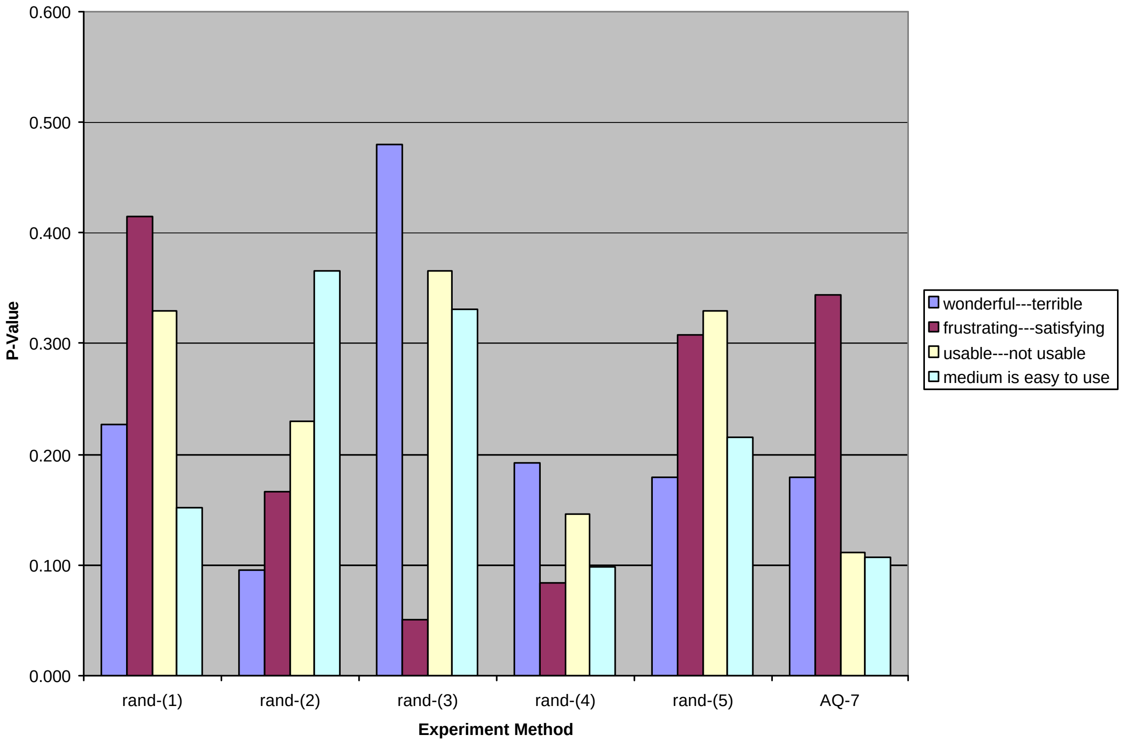

Figure 3: Comparison of statistical differences for group size 7: Random Trial vs. Applications Quest

Here in Figure 3, trials rand-(1), rand (5), and Applications Quest each produced all insignificantly different attributes. The p-values for the random trials were able to surpass those of Applications Quest, but the probability of those trials being selected was only 40 percent. The other random trials were able to generate attributes with insignificant difference but they still maintained a level of inconsistency.

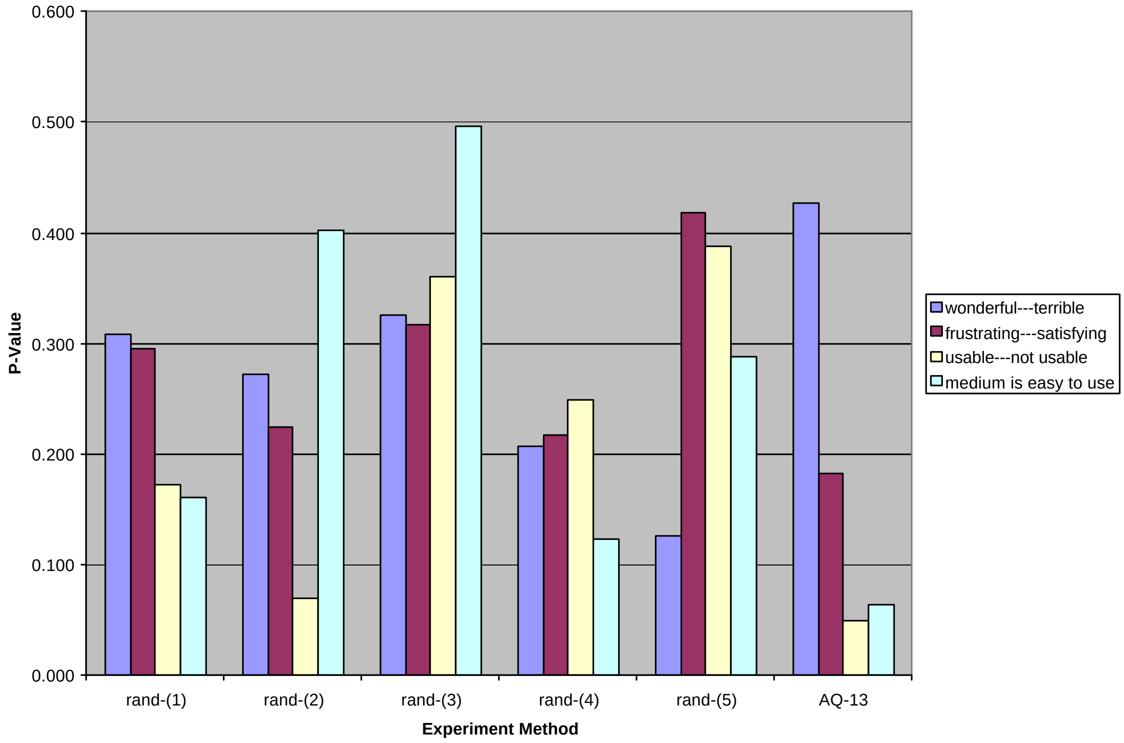

Figure 4: Comparison of statistical differences for group size 13: Random Trial vs. Applications Quest

In figure 4, 80 percent of the random trials produced insignificantly different results while Applications Quest was only able to produce two attributes that were insignificantly different. The random trials clearly outperformed Applications Quest but the size of the group makes the results questionable. The group size represents approximately 20 percent of the total populace and from the previous trials it has been shown that there exists a smaller subset of participants that can yield similar results.

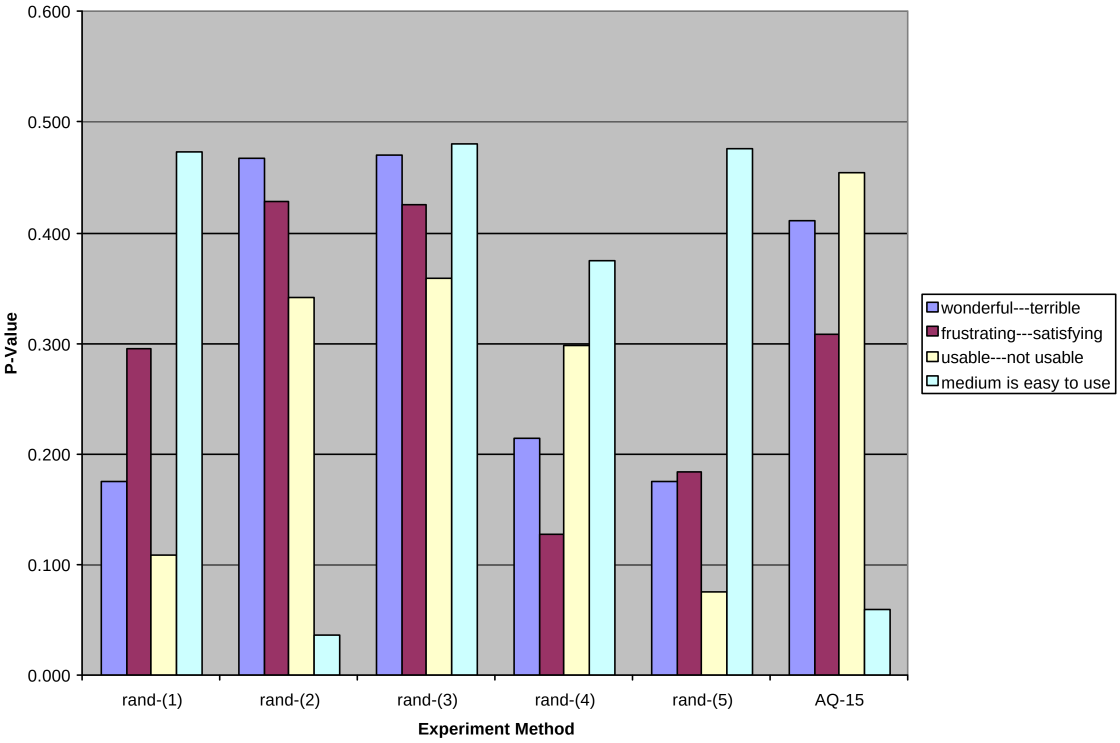

Figure 5: Comparison of statistical differences for group size 15: Random Trial vs. Applications Quest

Figure 5 shows that as more participants have been randomly selected, the results for the random trials got better across the board. Applications Quest throughout each of the trials has maintained a steady level of consistency by matching the results or performing better. Although the random trials present a high level of confidence, the number of participants is steadily increasing as would the price for user testing.

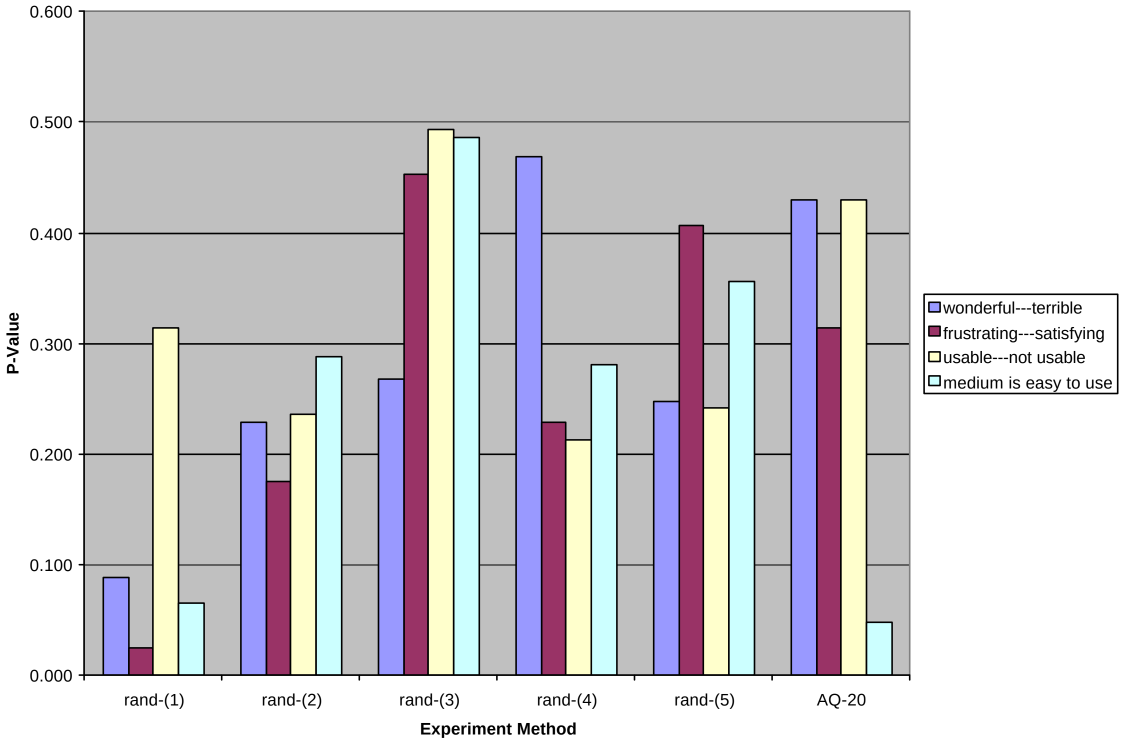

Figure 6: Comparison of statistical differences for group size 20: Random Trial vs. Applications Quest

Again in Figure 6, the results improved as more participants were selected randomly. From the graphs above it can be seen that Applications Quest although predicted to outperform the random trials was only able to match the results and in some cases perform less than expected across the board. These facts at the onset seemed disheartening but with further investigation of the data it can also be seen that when selecting individual attributes from the study, Applications Quest was able to demonstrate that it could select participants whose post-survey results were insignificantly different from those of the population. For example, in all of the above figures Applications Quest was able to select participants in every trial that were representative of the population for the attributes wonderful---terrible and frustrating--- satisfying. Because the results of Applications Quest in comparison with the random results initially seemed unaligned with this experiment's hypothesis, approach 2b (Applications Quest with a revised algorithm) was designed and the results are as follows:

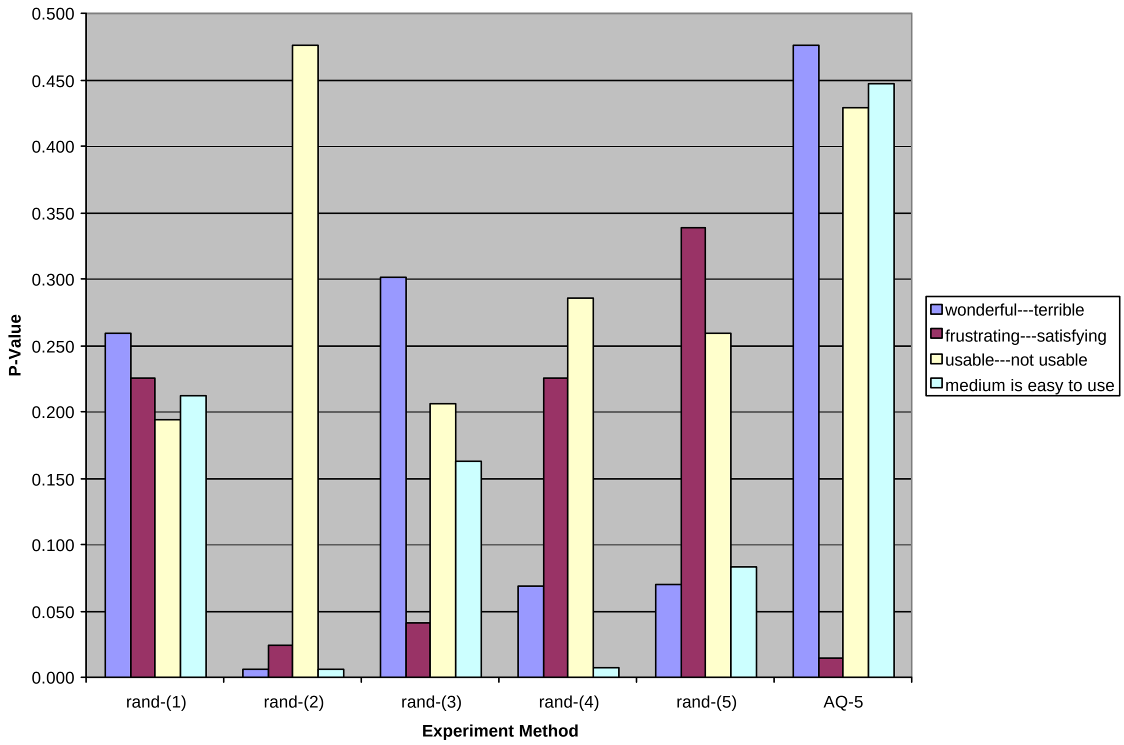

Figure 7: Comparison of statistical difference with revised algorithm for group size 5: Random Trial vs. Applications Quest

Figure 7 illustrates that even with a change in the Applications Quest algorithm, it still was able to produce exemplary results in comparison to the random trials. Random trial one was able to generate all attributes that were insignificantly different but the subsequent trials still demonstrated highly random results. When broken down into individual attributes, the attributes usable---not usable and frustrating---satisfying were consistently above the 90 percent confidence threshold for all trials, random and Applications Quest. The Applications Quest trial was able to produce three attributes that were insignificantly different and whose p-values were large enough in value to support a high confidence level in the participants selected.

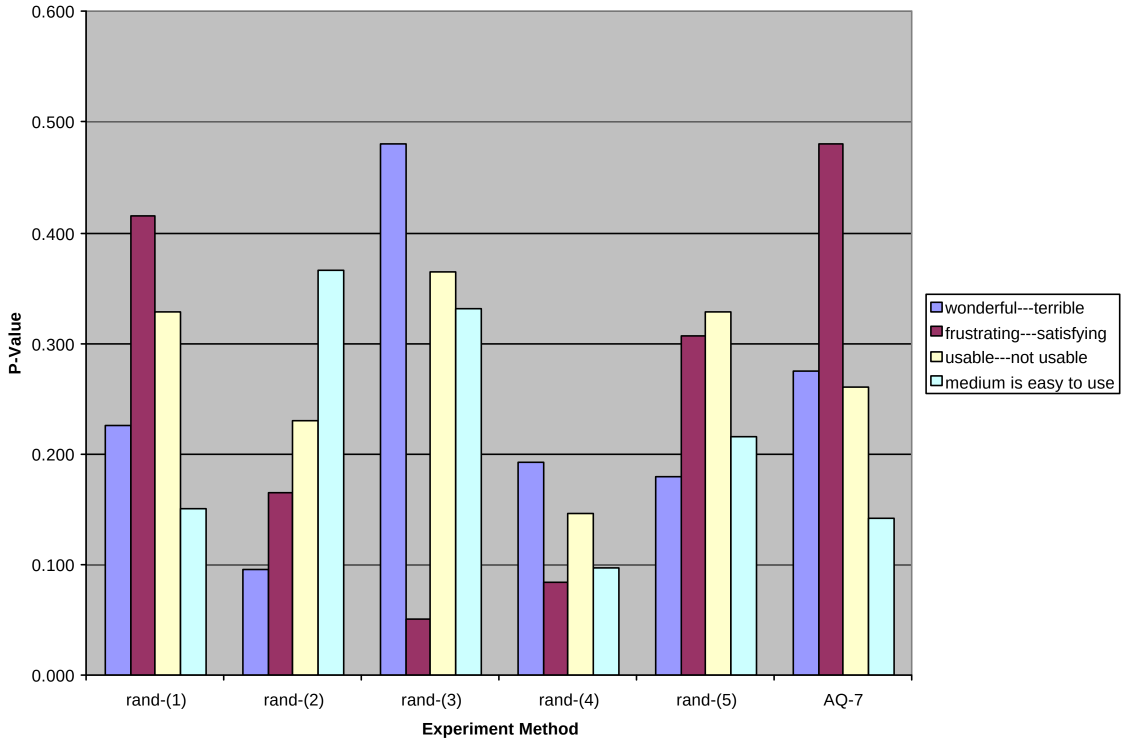

Figure 8: Comparison of statistical difference with revised algorithm for group size 7: Random Trial vs. Applications Quest

In the graph above, Applications Quest was able to produce all attributes with insignificant difference. Random trials one and three also successfully generated a complete set of attributes insignificantly different from the population. An interesting fact revealed from this figure and figure 3, Applications Quest group size 7 was able to produce complete sets of attributes insignificantly different from both versions of its algorithm. No other group size from the Applications Quest trials was able to demonstrate this. In this graph it can also be seen that the attribute usable---not usable was found insignificantly different in both the Applications Quest trial and the random trials.

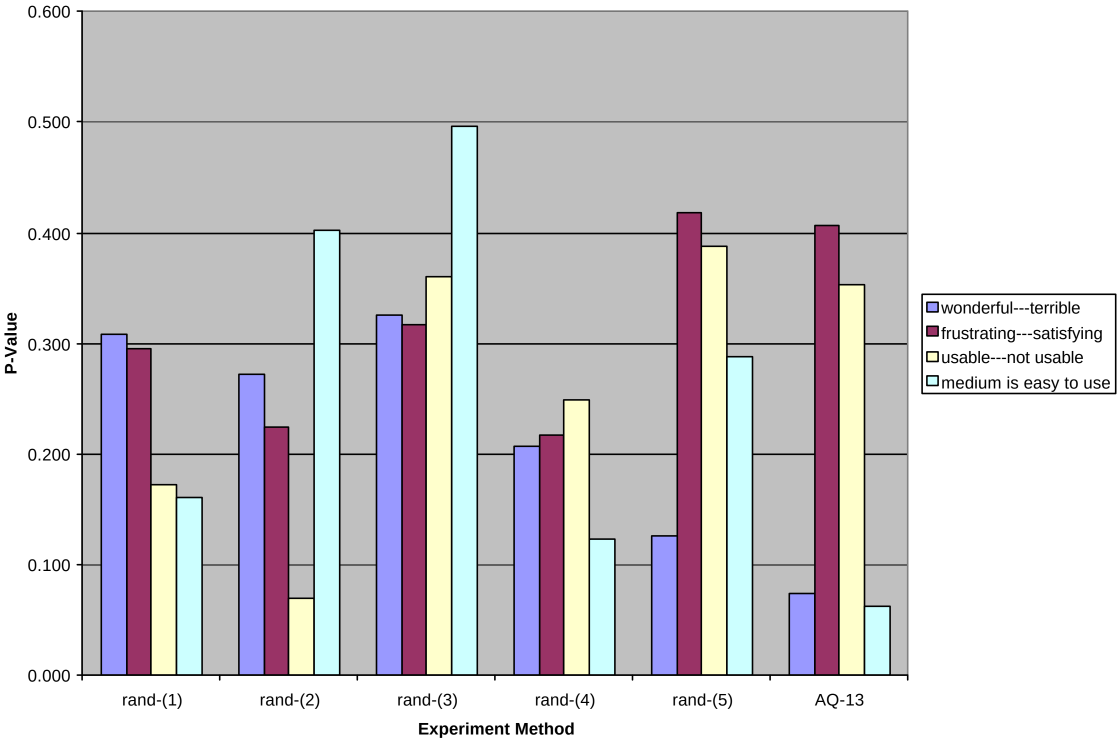

Figure 9: Comparison of statistical difference for group size 13:Random Trial vs. Applications Quest with revised algorithm

Here in Figure 9, 80 percent of the random trials produced insignificantly different results. Applications Quest was only able to generate two insignificant attributes. Frustrating---satisfying was the only attribute to consistently prove insignificant across each trial. Random trial three was able to provide a high level of confidence for each attribute but the point still remains throughout this experiment that the goal is to find the minimum number of participants that provide the same confidence level or higher.

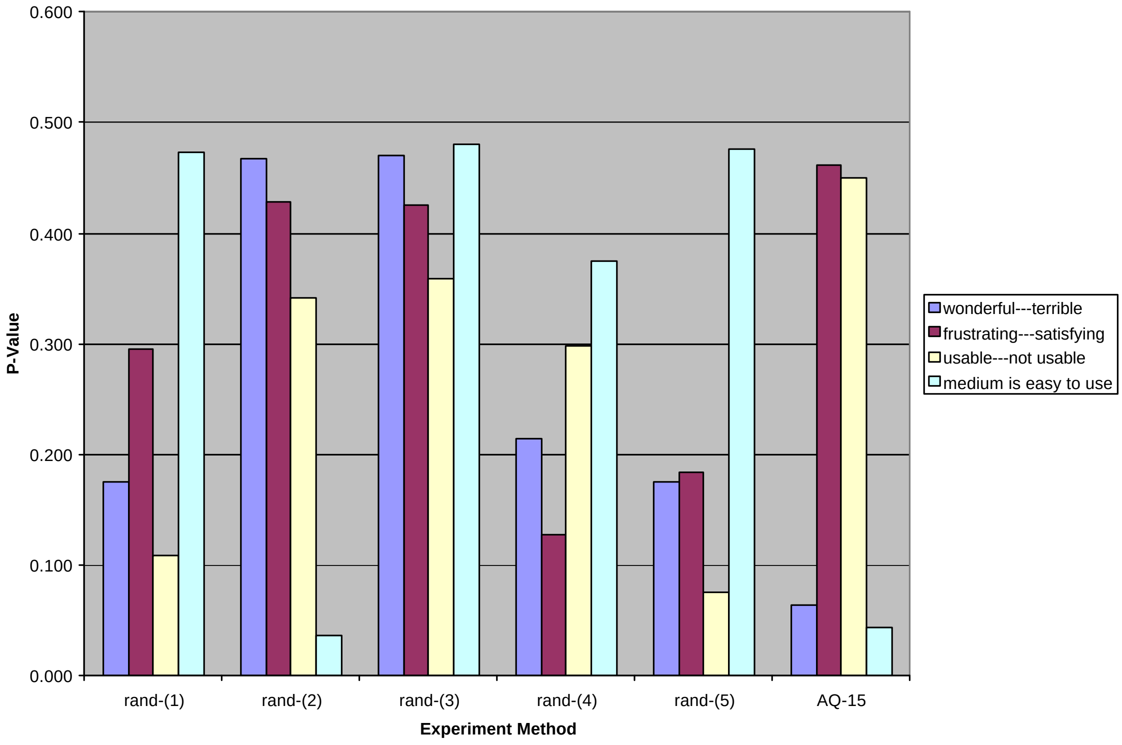

Figure 10: Comparison of statistical difference for group size 15: Random Trial vs. Applications Quest with revised algorithm

Figure 10 continues to demonstrate that the revised algorithm for Applications Quest can produce some attributes with insignificant difference but does not hold the proficiency that the original algorithm does. The graph shows that the random trials were superior in selecting participants as a complete set, but with an attribute breakdown, for the attributes frustrating---satisfying and usable---not usable Applications Quest selected the participants with a higher level of confidence. The increased level of confidence in the individual attributes suggests that as more participants are selected they represent a larger portion of the population, the population representing the targeted product audience; although the population is assumed to be somewhat similar, there should also exist some difference. In the case of this experiment, the data used can be considered mildly homogeneous thus the increase in confidence.

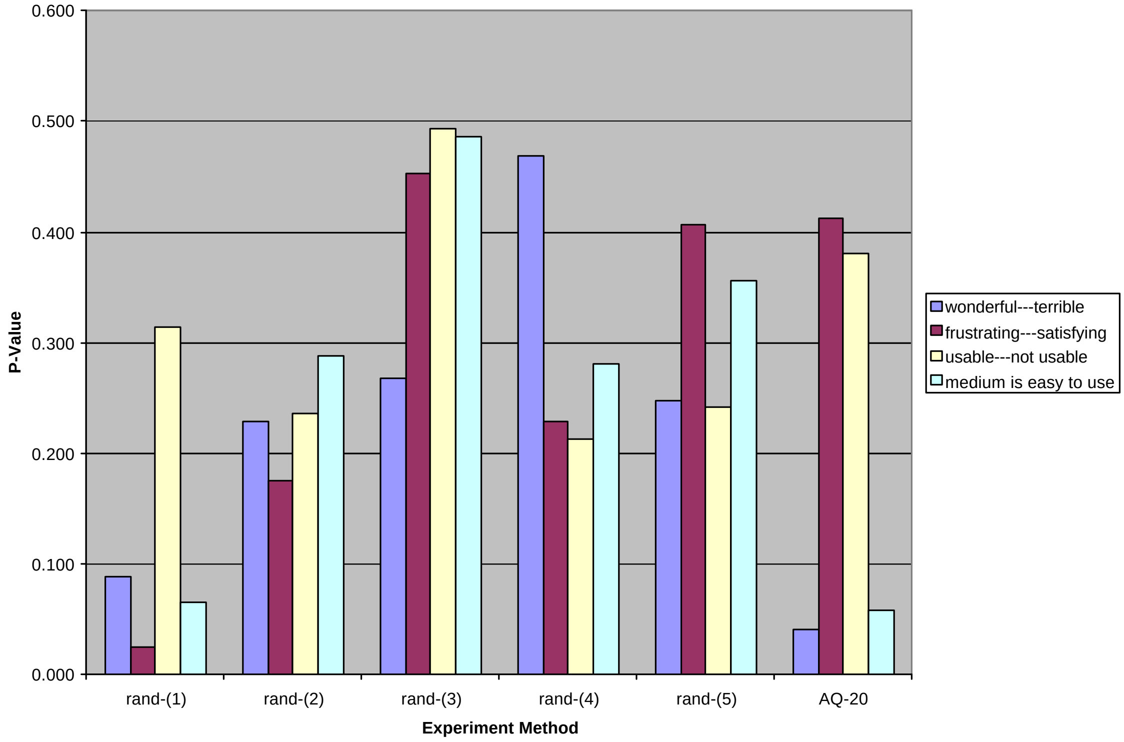

Figure 11: Comparison of statistical difference for group size 20: Random Trial vs. Applications Quest with revised algorithm

In Figure 11, Applications Quest produced two attributes whose results were insignificantly different. Throughout this part of the experiment (approach 2b), Applications Quest has done exceedingly well in selecting those two attributes, but across the board the random trials have sufficiently proven much better than Applications Quest. 80 percent of the random trials successfully selected participants representative of the population, this appears satisfactory but in the grand scheme of budgeting this number may be too large.

Although the revised algorithm for Applications Quest did not return all groups of users representative of the populace as hoped, meaning for every trial there were no groups significantly different; it did however present some other interesting results. Reviewing figures 7-11, one can see that the smaller groups' results returned were more representative than the random trials, percentage wise. Because the goal is to find the lowest number of participants representative of the population with a great number of certainty, Applications Quest outperformed the random samples. Another interesting finding was that group size 7 was returned from Applications Quest as the only group size that was found 100 percent insignificantly different in both approaches. This suggests that if a usability professional were designing a study he could be 100 percent certain that the group selected by Applications Quest would provide him the same results as the other seventy-two participants in the targeted population. With this certainty, the professional could use the seven users versus the seventy-two and save money on user testing, stay on budget for the usability design portion, and even stay on schedule for the time allotted to testing.

In the random trials, the results were promising but the problem was that the trials were too unpredictable. In Applications Quest, the algorithm for selection is the same every time, choose the participant that is the most similar or most different depending on the algorithm used. The random trial results returned the larger group as most representative which becomes a problem because the idea is to find the minimum number of participants. Group sizes 13, 15, and 20 had very high percentages of trials that were insignificantly different but that does not say much because that is almost a fourth of the populace.

In comparison, the Applications Quest algorithm for selecting the most similar participant was more effective in selecting a better percentage of representative groups. Also in the attribute breakdown, the most similar algorithm returned more attributes with 100 percent certainty of insignificant difference. These facts suggest that although both algorithms performed in close proximity, choosing which algorithm to use comes down to the goals of the study. If the study aims to find users that are most representative of the targeted population, they would use the most similar algorithm. If the study aims to find the most diverse yet similar users, they would use the most different algorithm.

5. CONCLUSIONS

5.1 CONCLUSIONS

The goal of a usability study is to work out the kinks of software and improve user satisfaction. When the job of designing a study and finding test users becomes too taxing, many designers and researchers abandon user testing and simply rely on heuristic evaluation. Those researchers who take on the task often find that recruiting users is daunting and time-consuming. The aim of this experiment was to help find a plausible solution to selecting participants for studies by using Applications Quest. Applications Quest would take a group of size N and from that group select participants that would be representative of the population. The selected participants would help reduce costs by minimizing the number of participants necessary while still maintaining result quality. Two approaches were used for comparison, random selection and Applications Quest selection. Although the random trials in this experiment were able to compete with the results of Applications Quest, Applications Quest was able to present results that were insignificantly different as well as consistently reproducible. The random trials were unpredictable and that fact does not lend to guarantee certainty or reliability when necessary in selecting participants. Upon comparing Applications Quest to itself when revising its original algorithm, the original version (selects the most representative) executed more effectively than the algorithm selecting the most different user. These findings suggest that Applications Quest could very well be a promising solution to the issue of user recruiting and selection. The results are reproducible and consistent and with more experimentation it could guarantee a higher percentage of insignificant difference.

5.2 FUTURE WORK

As stated previously and as seen in the graphs, the random trials of this experiment were able to contend with Applications Quest. With further research, we would like to see if Applications Quest can eliminate the competition. We would like to run the same experiment on less homogeneous data as well as run it on larger datasets. On a less homogenous larger distribution of data we may be able to find that Applications Quests can select the most representative more efficiently because a larger distribution will lend itself to comparing a less dense cluster. We would also like to try this experiment with more demographic data. If we could outline different sets of demographic data with human subjects into Applications Quest, maybe we could see a trend in what information usability professionals could use in recruiting participants.

REFERENCES

1. Faulkner, Laura. (2003) Beyond the five-user assumption: Benefits of increased sample sizes in usability testing, *Behavior Research Methods*, *Instruments, & Computers*. (March 9, 2007) from www.geocities.com/FaulknerUsability/Faulkner_BRMIC_Vol35.pdf
2. Nielsen, Jakob. (2000, March) Why You Only Need to Test With 5 Users, *Alertbox*. (March 9, 2007) from www.useit.com/alertbox/20000319.html
3. Heim, Steven. The Resonant Interface: HCI Foundations for Interaction Design, Pearson Addison Wesley, (2008).
4. Carroll, John M., Rosson, Mary Beth. Usability Engineering: Scenario-Based Development of Human-Computer Interaction, Morgan Kaufmann Publishers, (2002).
5. Landesman, Lori., Perfetti, Christine. (2001, June) Eight is not Enough, *User Interface Engineering*. (March 29, 2007) from www.uie.com/articles/eight_is_not_enough/.
6. Nielsen, Jakob. (2000, March) Recruiting Test Participants for Usability Studies, *Alertbox*. (March 9, 2007) from http://www.useit.com/alertbox/20030120.html.
7. Gilbert, Juan. (2004) Applications Quest. (March 29, 2007) from www.applicationsquest.com.

8. (2005, Feburary) Arteology: Sampling. (March 27, 2007) from www.uiah.fi/projects/metodi/152.htm

9. Rosenstein, Aviva. (2001, September) Managing Risk with Usability Testing, *Classic System Solutions*. (March 29, 2007) from http://www.classicsys.com/css06/cfm/articlesusability.cfm.

APPENDIX A

Table 1. Trial results from random selections

	Terrible-Wonderful	Frustrating-Satisfying	Usable-Not Usable	Easy medium
rand-5(1)	0.2597	0.2255	0.1941	0.2125
rand-5(2)	0.0058	0.0238	0.4758	0.0066
rand-5(3)	0.3016	0.0414	0.2065	0.1631
rand-5(4)	0.0686	0.2255	0.2856	0.0073
rand-5(5)	0.0707	0.3387	0.2593	0.0837
rand-7(1)	0.2265	0.4155	0.3289	0.1511
rand-7(2)	0.0953	0.1658	0.2300	0.3665
rand-7(3)	0.4797	0.0506	0.3652	0.3308
rand-7(4)	0.1928	0.0840	0.1462	0.0980
rand-7(5)	0.1791	0.3075	0.3289	0.2157
rand-13(1)	0.3089	0.2958	0.1721	0.1610
rand-13(2)	0.2724	0.2246	0.0703	0.4017
rand-13(3)	0.3261	0.3175	0.3598	0.4957
rand-13(4)	0.2071	0.2167	0.2485	0.1228
rand-13(5)	0.1265	0.4188	0.3874	0.2883
rand-15(1)	0.1759	0.2953	0.1094	0.4735
rand-15(2)	0.4674	0.4276	0.3413	0.0362
rand-15(3)	0.4706	0.4259	0.3587	0.4798
rand-15(4)	0.2138	0.1276	0.2983	0.3749
rand-15(5)	0.1759	0.1843	0.0754	0.4762
rand-20(1)	0.0892	0.0254	0.3141	0.0656
rand-20(2)	0.2292	0.1747	0.2366	0.2873
rand-20(3)	0.2677	0.4526	0.4934	0.4857
rand-20(4)	0.4689	0.2289	0.2126	0.2807
rand-20(5)	0.2471	0.4069	0.2423	0.3557

Table 2. Trial results from Applications Quest (most similar algorithm)

	Terrible-Wonderful	Frustrating-Satisfying	Usable-Not Usable	Easy medium
appsquest - 5	0.3016	0.3014	0.0101	0.0066
appsquest - 7	0.1794	0.3439	0.1118	0.1062
appsquest - 13	0.4269	0.1831	0.0489	0.0636
appsquest - 15	0.4108	0.3090	0.4537	0.0597
appsquest - 20	0.4296	0.3143	0.4295	0.0480

Table 3. Trial results from Applications Quest (most different algorithm)

	Terrible-Wonderful	Frustrating-Satisfying	Usable-Not Usable	Easy medium
appsquest - 5	0.4762	0.0144	0.4295	0.4473
appsquest - 7	0.2756	0.4808	0.2611	0.1420
appsquest - 13	0.0740	0.4059	0.3532	0.0630
appsquest - 15	0.0634	0.4611	0.4502	0.0436
appsquest - 20	0.0407	0.4125	0.3807	0.0583