JUS
JOURNAL OF USABILITY STUDIES

Issue 1, Vol. 1, November 2005, pp. 18-27

Iterative Usability Testing as Continuous Feedback: A Control Systems Perspective

Alex Genov, PhD
Staff User Researcher
User Centered Design
Consumer Tax Group
Intuit, Inc.
6220 Greenwich Drive
San Diego, CA 92122 USA
Alex_genov@intuit.com

Abstract

This paper argues that in the field of usability, debates about number of users, the use of statistics, etc. *in the abstract* are pointless and even counter-productive. We propose that the answers depend on the research questions and business objectives of each project and thus cannot be discussed in absolute terms. Sometimes usability testing is done with an implicit or explicit hypothesis in mind. At other times the purpose of testing is to guide iterative design. These two approaches call for different study designs and treatment of data. We apply control systems theory to the topic of usability to highlight and frame the value of iterative usability testing in the design lifecycle. Within this new metaphor, iterative testing is a form of feedback which is most effective and resource-efficient if done as often as practically possible with project resources and timelines in mind.

Keywords

usability method, usability metric, usability data analysis, teaching usability, laboratory study, metaphor

Introduction

This is an exciting time for the developing field of usability. A number of questions appear repeatedly in presentations, publications, website opinion articles, online discussion forums, and so on. Among those questions are: What is a usability problem (e.g., Nielsen, 1992b; Nielsen & Landauer, 1993; Kessner, Wood, Dillon, & West, 2001; Hertzum & Jacobsen, 2001; Cockton & Lavery, 2002)? What is the appropriate number of users to find all usability problems in an interface, i.e. for a usability test to be valid (e.g., Cockton & Woolrych, 2002; Lewis, 1994; Nielsen, 1994; Nielsen & Landauer, 1993; Spool & Schroeder, 2001; Virzi, 1990; Virzi, 1992; Woolrych & Cockton, 2001; Caulton, 2001)? Is there a place for statistics in usability testing? What metrics should be collected and reported? What is the appropriate place of usability testing in the design lifecycle?

All of these are topics with important practical implications. In this paper we suggest that such debates *in the abstract* are pointless and even counter-productive. We propose that the answers depend on the research questions and business objectives of each project and thus cannot be discussed in absolute terms. In other words, the usability and business objectives determine the research methods used. Sometimes usability testing is done with an implicit or explicit hypothesis in mind. At other times the purpose of testing is to guide iterative interface design and development. These two approaches call for different research methodologies and treatment of data. We apply control systems theory (CST) (Wiener, 1948) to the topic of usability to highlight and frame the value of iterative usability testing in the design lifecycle. Control systems is a theoretical framework that rationalizes the difference between usability hypothesis testing and iterative design testing and supports the value of iterative testing with 5-6 users with each iteration (e.g., Nielsen, 1990b; Nielsen & Landauer, 1993; Virzi 1992; Woolrych & Cockton, 2001). In practice, the CST metaphor can be used by usability professionals not only as a way of thinking about iterative usability testing, but also as a metaphor for communicating its value to development teams and management. Our discussion will take place in the context of the business environment where the choice of research methods is associated with production costs. It is our belief that the usually costly experimental research rigor should be balanced with project deadlines and the availability of resources.

Hypothesis testing and iterative testing

In usability testing, formal evaluations are broadly categorized as either formative or summative. According to Nielsen, the main goal of formative evaluations is to "learn which detailed aspects of the interface are good and bad, and how the design can be improved" (Nielsen, 1993, p. 170). Summative evaluations, on the other hand, are suitable for "assessing the overall quality of the interface, for example, for use in deciding between two alternatives" (Nielsen, 1993, p. 170). A similar dichotomy can be discerned in Rubin's discussion of exploratory and assessment tests on the one hand, and validation and comparative tests, on the other (Rubin, 1994, pp 31-40). It can be argued that summative evaluations are a category of research methods that involve testing a particular hypothesis, while formative evaluations are used in iterative interface design and development.

Hypothesis testing

Some research questions call for a more rigorous experimental approach. These are cases when usability and business goals involve hypothesis testing. According to Keppel (1982), "[a] research hypothesis is a fairly general statement about the assumed nature of the world that gets translated into an experiment" (p.25). Expanding on Nielsen (1993) and Rubin (1994), we will argue that summative evaluations and increased experimental rigor are most appropriate for doing benchmark and comparative evaluations, to take two methods. For example, with benchmark usability, one can test the hypothesis that average task completion times, error rates, and success rates for the particular version of a product differ from industry averages or from averages for previous versions. With comparative usability, one can test the hypothesis that average metrics for product A are significantly different (higher or lower) than those for product B.

Iterative usability

Experimental rigor is not necessary for all usability research. There are evaluations whose main goal is to help guide interface design and development, namely iterative or exploratory usability testing. The lack of experimental rigor however does not mean that exploratory evaluations, which involve 5 or 6 users per iteration are subjective and have no empirical value compared to summative evaluations. This argument is supported not only by the opinion and research of experts in the usability field (e.g., Nielsen & Landauer, 1993) but also by a theoretical scientific perspective, namely control systems theory (CST) (Wiener, 1948). When applied to usability research, CST suggests that it is better to have several small tests that span the length of the development process than fewer larger tests towards the end. In addition, the CST perspective supports the argument that testing is more critical early on in design and development, when more substantial changes are less costly.

Control systems theory: A basic example

Control systems (or cybernetic) theory provides an explanation for any behavior that can be described as goal-directed, or purposeful. At a high level of abstraction, it is applicable to usability testing because the latter is, or should be, guided by specific design and business goals.

Control systems theory has been applied to the explanation of phenomena in a variety of fields, such as Physiology (Cannon, 1929, 1939), Engineering (Wiener, 1948), and Psychology (Powers, 1973a). The basic idea behind the theory is that the behavior of a self-regulating system is effective and efficient only to the extent to which it successfully controls the end results of the behavior. In other words, it is important not only to produce a certain output in reaction to events in the environment, but also to monitor *continuously* the effects of that output and to adjust any subsequent behavior accordingly (Wiener, 1948). This is the idea of a *feedback loop*. Since purposeful behavior involves the movement toward a certain end state, behavior will be effective only if it reduces any discrepancy between the current state of the system and its desired (reference) state. For this reason, the kind of feedback essential for the proper operation of control systems is *negative feedback*. Figure 1 depicts a schematic representation of a very basic control system. In the next section these processes will be related to usability testing.

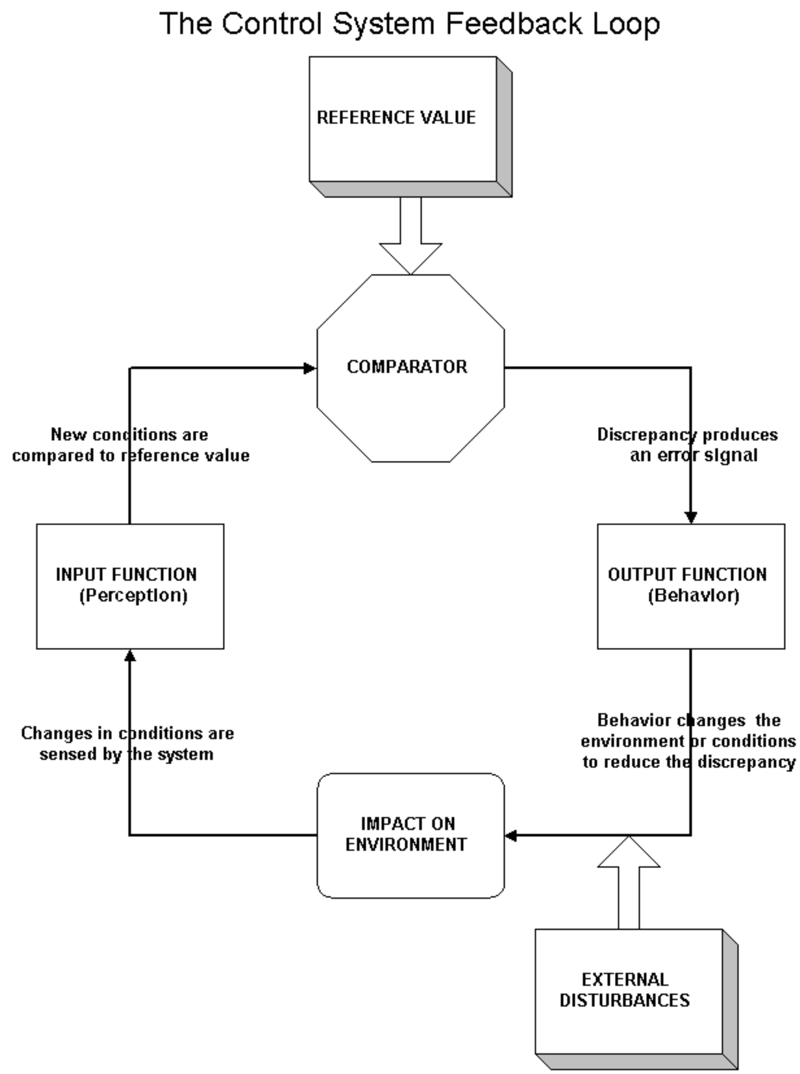

Figure 1. The basic control system feedback loop

The "input function" is the sensing of the current state. That perception is compared against a point of reference through a mechanism called "comparator." If a discrepancy is perceived between the present state and the desired (or reference) state a behavior is performed, which is the "output function." The goal of the output function is to reduce the discrepancy. The output has an impact on the system's environment (i.e. anything external to the system). Such an impact creates a change in the present condition, leading to a different perception, which in turn is once again compared with the reference value (Carver & Scheier, 1982, p.11).

To illustrate the nature of the continuous negative feedback loop let's take the widely used example of the home thermostat as a simple control system. The reference value for the room temperature, or the desired state (let's say 72°), is set by someone. A sensor, which is part of the input function, detects the current room temperature (say 68°). The difference between 68° and 72° is sensed by the thermostat's "comparator" and this discrepancy leads to the starting of the furnace, which is the output function or the "behavior." The furnace pumps hot air until the temperature sensor of the thermostat detects a room temperature of 72° at which time, it turns the furnace off since there is no more discrepancy between the current state and the desired, or reference, state. The goal has been achieved. One important point here is that the goal has been achieved efficiently because the thermostat sensor has been sampling the air temperature *continuously*. Imagine that the readings were taken every hour. What if at the end of the first hour the furnace has produced an air temperature of 80°. Now to achieve the desired temperature of 72°, the cooling system has to be turned on. Thus, infrequent readings of the current state of the system may lead to vast oscillations around the target value and a consequent waste of resources.

Granted, the house thermostat represents a very simple control system. How would we apply this example to usability? If we see the interface design system as a complex control system guided by goals, the same principle would apply. That is, infrequent feedback about the usability of a given interface may lead to the need for major product redesign and a waste of resources. Next we explore the idea of complex control systems and User Centered Design.

Complex control systems and usability testing

In order to apply the control systems model to usability testing, it would be useful to examine its applications to more complex behavioral systems. For instance, the idea of a feedback system has been applied in the context of Social Psychology. According to Carver and Scheier (1982), the main function of such a system is to maintain the perception of a desired condition. In control systems terms, the feedback system is designed to minimize the difference between the current state and the desired state (reference) state. For example, Jane wants to maintain an image of herself as successful. This is a very high level, abstract, goal. According to control systems theory, she will vary her behavior to align the perception of her current situation with her ideal state. Let's assume that Jane defines success with the amount of financial assets she possesses. Thus to view herself as success, she may work toward getting a pay raise, she may get a second job, she may invest, and so on. Periodically, Jane will compare her current assets with her desired assets. If there is a discrepancy, an error signal will prompt her to continue to act so as to reduce it and ultimately eliminate it.

One important notion of control systems theory, as applied to complex behavior, is that of *hierarchy* (Powers, 1973b) where a number of perceptual-behavioral feedback loops are interconnected at different levels of abstraction. The output of the more abstract loops provides the reference value for the more concrete loops. In our example, the goal of maintaining an image of success provides a very general guiding principle for self-regulation. In order to put this principle into practice, one has to perform more concrete behaviors, such as investing, for example. To do that, Jane has to make an appointment with an investment broker and drive to that appointment. Driving, in its turn, involves a sequence of behaviors that ultimately can be reduced to muscle movements. This example shows how a system can regulate the achievement of abstract goals (e.g. success in life) through the regulation of concrete and observable behaviors (e.g. muscle tension).

CST is a useful theoretical framework for usability testing for several reasons. Firstly, usability testing is as an integral part of the User Centered Design (UCD) process which is a complex system driven by goals. Secondly, UCD and usability goals are hierarchically organized. Thirdly, iterative usability testing (as opposed to benchmark testing) can be viewed as a form of *continuous* feedback that guides design (Figure 2).

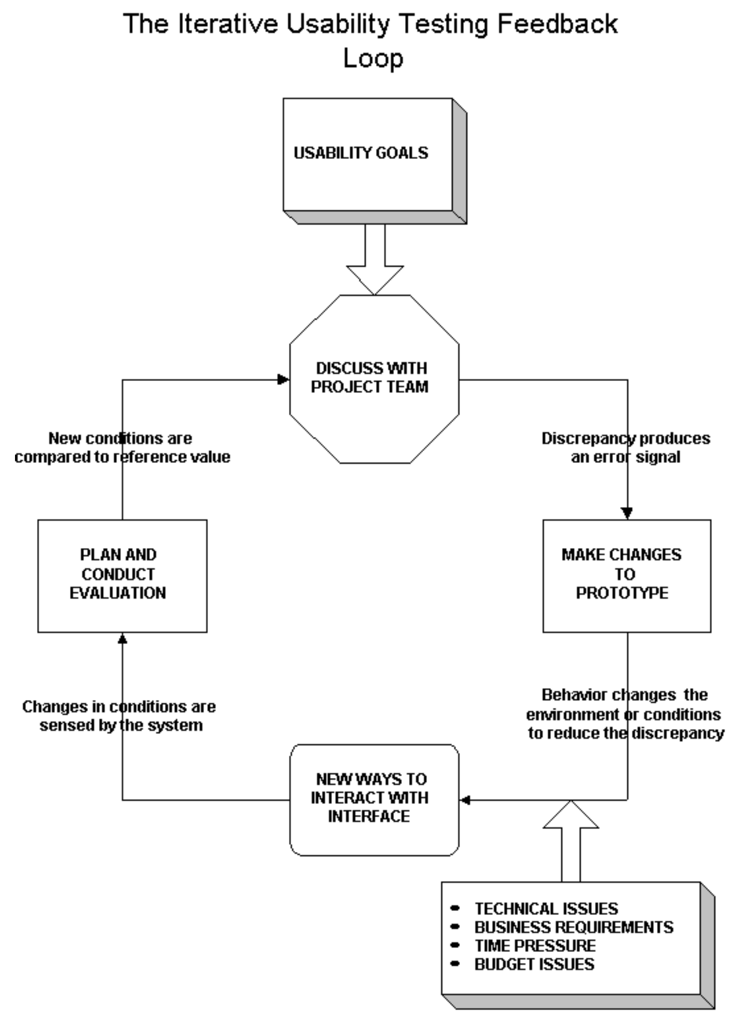

Figure 2. The iterative usability testing feedback loop

Gould, Boies, and Ukelson's principle on early focus on users (Gould at al., 1997) highlights the importance of goals in iterface design. User-centered design is driven by several types of goals with two broad categories being business requirements and ease of use goals. Regarding the latter, benchmark and comparative testing, for example, are driven by more concrete goals and should be accompanied by tests of statistical significance of any differences the corresponding metrics, such as task completion rate, time on task, error rate, and so on. This is warranted if expensive business decisions hinge on the testing results and thus justify the higher cost of such tests. Iterative testing, on the other hand, is (or should be) driven by the more general goal of identifying potential major problems that can be easily fixed early on in the design lifecycle. Iterative testing can make use of the same metrics, but the data gathered from the small sample (i.e. 5 users per iteration) should be used in more qualitative ways. For example, gathering success rate for 5 users over 10 tasks can reveal the existence of major usability problems for specific areas of the user interface. It may be the case that all 5 users failed at creating a profile on a website due to an interface usability issue. If we follow the ISO definition of "usability" as a combination of "effectiveness, efficiency, and satisfaction" (ISO 9241-14) our iterative testing results can inform the project team that there is a possibility that users in general may not be able to complete a key task in their workflow. Not being able to complete a key task is arguably a major or severe usability problem. Not being able to complete a task quickly indicates lack of efficiency, which can be considered a less severe problem. In sum, data from iterative usability testing can be used to pinpoint specific tasks with which users had the most difficulty and specific aspects of the interface that should be redesigned early in the design lifecycle.

In this model, the reference values come from various sources. The high level goals may come from the business. For example, one goal can state that customers should like the product so much that they would recommend it to their friends. Another business goal may be to create a product that satisfies a need so well that people adopt it to replace a previously manual task. More concrete usability goals may include the successful completion of a registration process for a secure website, for instance. Other reference values may come from established industry standards or competitive studies data.

The initial state of the system, i.e. the first prototype of a given product, is the result of collaborative efforts of product managers, business analysts, information architects, graphic designers, and software developers. The prototype is then tested with 5 typical users of the product, who complete a set of pre-determined tasks. The tasks, developed by usability engineers in collaboration with the project team, are based on the typical users' workflow and are designed to probe for potential difficulties users might have with key activities. The interaction of the users with the system is observed by usability engineers and any difficulties users have with the system are noted. The results of the test (the input function in control systems terms) are compared to the usability goals generated in advance (the reference value). If usability testing indicates the existence of even <u>one</u> major problem that prevents users from accomplishing a key task, there is a discrepancy between the current state and the desired state of <u>no</u> major problems. This discrepancy is the error signal that prompts redesign suggestions. These suggestions are then implemented and tested again until there is an indication that the target user group can accomplish the appropriate tasks effectively and efficiently using a given product, and also has a positive subjective experience with the product. To get this indication, iterative, or exploratory testing, also utilizes metrics such as timings, success and error rates, etc. However, unlike benchmark testing those metrics are used in more qualitative ways to guide the design by highlighting gaps between the desired and actual effectiveness, efficiency, and satisfaction with a particular interface (Figure 3).

Conclusion

In conclusion, control systems theory highlights and frames the value of iterative usability testing in the design lifecycle. It can be regarded as a form of feedback which is most effective and resource-efficient if done as often as practically possible with project resources and timelines in mind. The CST metaphor can be used by usability professionals not only as a way of thinking about iterative usability testing, but also as a metaphor for communicating its value to development teams and management.

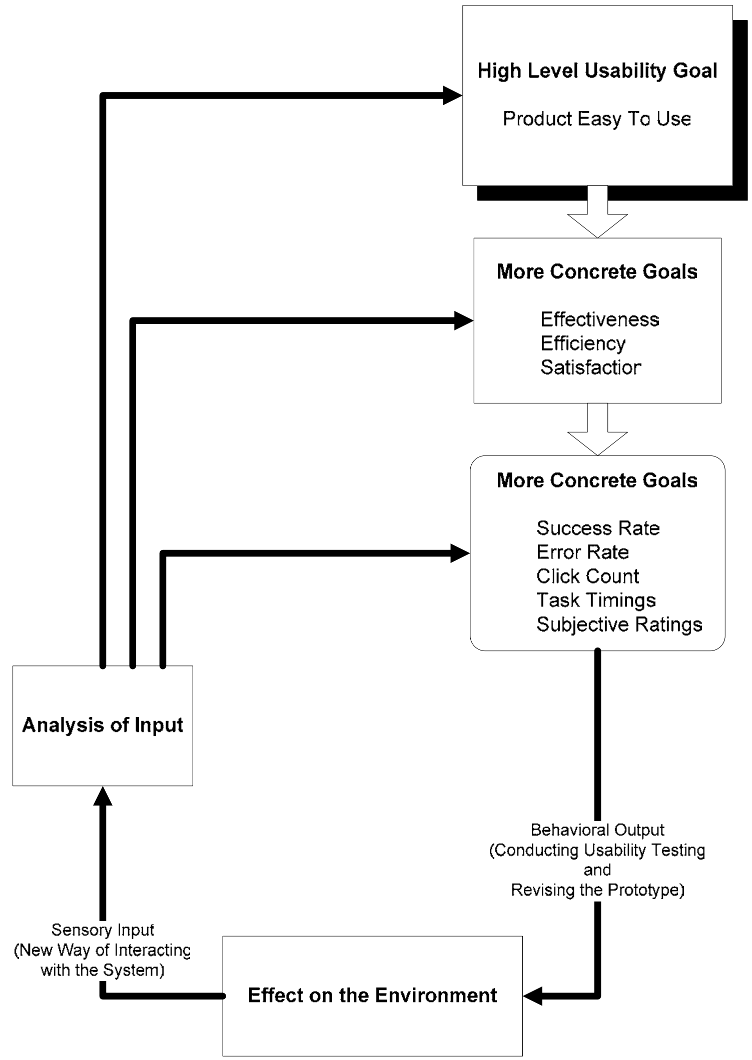

Figure3. Hierarchical organization of usability goals

Practitioner's take-aways

Here are a few practical implications of the above discussion:

- Do not debate the appropriateness of specific user research methods *in the abstract*.
- Before selecting a research method, always clarify the research questions and business objectives of each project and get team buy in.
- Consider iterative usability testing a form of feedback on the progress towards specific design and business goals.
- Start doing iterative testing as early as possible in the design lifecycle.
- Conduct iterative testing as often as practically possible with project resources and timelines in mind.

Acknowledgements

I would like to extend special thanks to Dr. Jim Laird for helping me see the practical side of Control Systems Theory and to Dr. Judy Ramey for assisting me in refining and sharpening the ideas.

References

Cannon, W.B. (1929). Bodily Changes in Pain, Hunger, Fear, and Rage. 2nd ed. D. Appleton and Co., New York and London.

Cannon, W.B. (1939). The Wisdom of the Body. 1st rev. ed. WW Norton and Company, New York.

Carver, C.S. & Scheier, M.F. (1982). Control theory: A useful conceptual framework for personality - social, clinical, and health psychology, Psychological Bulletin, 92(1), 111-135.

Caulton, D.A. (2001). Relaxing the homogeneity assumption in usability testing. Behavior & Information Technology, Vol. 20(1), 1-7.

Cockton, G., & Lavery, D. (2002). A framework for usability problem extraction. Proceedings of INTERACT 1999, 347-355. Amsterdam, The Netherlands: IOS Press.

Cockton, G., & Woolrych, A. (2002). Sale must end: Should discount methods be cleared off HCI's shelves? Interactions, IX.5, 13-18.

Gould, J. D., Boies, S. J., & Ukelson, J. (1997). How to design usable systems. In Helander, M. G., Landauer, T. K., & Prabhu, P. V. (Eds.), Handbook of human-computer interaction, 2d ed., 231-254. Amsterdam, The Netherlands: North-Holland.

Hertzum, M., & Jacobsen, N. E. (2001). The evaluator effect: A chilling fact about usability evaluation methods. International Journal of Human-Computer Interaction, 13, 4, 421-443.

ISO/IEC. 9241-14 Ergonomic requirements for office work with visual display terminals (VDT)s - Part 14 Menu dialogues, ISO/IEC 9241-14: 1998 (E), 1998.

Keppel, G. (1982). Design and analysis: A researchers's handbook. Prentice-Hall, Inc., Englewood Cliffs, New Jersey.

Kessner, M., Wood, J., Dillon, R. F., & West, R. L. (2001). On the reliability of usability testing. CHI 2001 Extended Abstracts, 97-98. New York, NY: ACM.

Lewis, J. R. (1994). Sample sizes for usability studies: Additional considerations. Human Factors, 36, 368-378.

Nielsen, J. (1990b). Big paybacks from 'discount' usability engineering. IEEE Software, 7, 3, 107-108.

Nielsen, J. (1992b). Finding usability problems through heuristic evaluation. Proceedings of CHI 1992, 373-380. New York, NY: ACM.

Nielsen, J. (1993). Usability engineering. Boston, MA: AP Professional.

Nielsen, J. (1994c). Estimating the number of subjects needed for a thinking aloud test. International Journal of Human-Computer Studies, 41, 3, 385-397.

Nielsen, J., & Landauer, T. K. (1993). A mathematical model of the finding of usability problems. Proceedings of INTERCHI 1993, 206-213. New York, NY: ACM.

Powers, W.T. (1973a). Behavior: The Control of Perception. Chicago: Aldine.

Powers, W.T. (1973b). Feedback: beyond Behaviorism. Science, Vol. 179, 351-356.

Rubin, J. (1994). Handbook of usability testing: How to plan, design, and conduct effective tests. New York, NY: John Wiley & Sons.

Spool, J., & Schroeder, W. (2001). Testing websites: Five users is nowhere near enough. CHI 2001 Extended Abstracts, 285-286. New York, NY: ACM.

Virzi, R. A. (1990). Streamlining the design process: Running fewer subjects. Proceedings of the Human Factors Society 34th Annual Meeting, 291- 294. Santa Monica, CA: HFES.

Virzi, R. A. (1992). Refining the test phase of usability evaluation: How many subjects is enough? Human Factors, 34, 4, 457-468.

Wiener, N. (1948). Cybernetics, or Control and Communication in the Animal and the Machine. New York: Wiley

Woolrych, A., & Cockton, G. (2001). Why and when five test users aren't enough. Proceedings of IHM-HCI 2001, Volume 2, 105-108. Toulouse, France: Cépaduès-Editions.

Alex Genov received his bachelor's degree in Psychology from Cornell University and his masters and doctoral degrees in Experimental Social Psychology from Clark University. Currently, Alex is applying his knowledge as a Staff User Researcher with the Consumer Tax Group at Intuit, Inc.