JUS
JOURNAL OF USABILITY STUDIES

Vol. 7, Issue 3, May 2012 pp. 88-93

Usability of Interactive Systems: It Will Get Worse Before It Gets Better

Jeff Johnson
President and Principal Consultant
UI Wizards Inc.
231 Moscow Street
San Francisco, CA 94112
USA
jjohnson@uiwizards.com

Austin Henderson
Principal
Rivendel Consulting & Design
644 Cragmont Ave.
Berkeley, CA 94708
USA
henderson@rivcons.com

One would think that by now that software applications, online services, and electronic appliances that are poorly designed and difficult to learn and operate would be a thing of the past. After thirty years of CHI conferences, twenty years of UPA conferences, and countless books, articles, and websites about usability and interaction design, shouldn't all software developers "get it" by now that careful task-focused design and usability are crucial for the success of interactive products and services?

Alas, it is not so. In fact, more companies than ever still don't get it: They are churning out bad interactive products and services with little or no awareness that they are doing so, and/or no idea how to do better.

Certainly, the fields of HCI, interaction design, and usability have matured. The knowledge they have developed has spread worldwide—no longer is UI design practiced exclusively in the western, industrialized world.

Usability Professionals' Association
Promoting usability concepts and techniques worldwide

Explosion in Digital Technology

However, the marketplace and the pervasiveness of interactive computer-based technology also has grown immensely. When interactive computing was first emerging, the main products were computers themselves and their operating systems and applications.

Today, just about every product—e.g., watches, cameras, telephones, microwave ovens, PDAs, GPS units, cars, music players, even toasters—has one or more computers inside it. Companies that used to make watches now make digital watches, calculators, and music players. Companies that formerly sold typewriters now sell multi-function printer-fax-copiers. Companies that once made chemical titration tubes and litmus paper now make genome sequence analysis instruments and corresponding control and analysis software. Other examples of products that have become digital include: home heating systems, home security systems, refrigerators, parking meters, transit ticket kiosks, books, thermometers, cameras, metronomes—even picture frames and toilets. Furthermore, functionalities that were formerly provided in separate devices have been merged, e.g., camera-phone-music-players, printer-fax-copiers, pen-recorders, gas-station-pump-TVs. Finally, entire new functionalities have emerged, e.g., GPS position-tracking.

In addition, the Internet's importance today means that any company, organization, or person in the developed world will have some sort of presence on the Web and in social media. Regardless of whether this presence is a full-blown website, a blog, a Facebook "wall," a LinkedIn profile, a YouTube channel, or a Twitter feed, the owner has to design it (within the constraints of the platform). That effectively makes every company, organization, and person with any Internet presence into a software designer, whether or not they are one in the offline world. Of course, this is especially true for companies and organizations with websites providing online services, such as sales, reservations, maps, document creation and storage, and online forms.

Many Old Companies Are Software Development Neophytes

These changes mean that the number of companies and organizations developing interactive software is exploding. Unfortunately, knowledge about interaction design and usability is absent in many companies and is not being disseminated fast enough to keep up. As a result, the proportion of interactive products created without benefit of good UI design thinking and process is growing. So usefulness and usability is falling behind.

In our experience with helping companies and organizations create usable, useful, popular products and services, we find that a large number of our clients are at what Eric Schaffer (2004) calls Usability Maturity Level 0 (clueless) or Level 1 (piecemeal usability). That is, they have no one on staff with (a) UI design or evaluation experience; (b) knowledge of the HCI literature, and in some cases knowledge that there even is such a literature; and (c) membership in professional societies devoted to human-computer interaction and usability (e.g., ACM SIGCHI, UPA, ISO TC13, and HCI International). In some companies that develop software, there are no employees who even know what interaction design is—many confuse it with graphic design.

It's not just about knowledge; development processes add to the problem. Of course in companies that lack awareness of the need for interaction design, it is not surprising that there is little place for interaction design in their software development processes. What is surprising is the lack of a place at the product development table for interaction design at companies that do know about its importance. In our experience, we often see companies in which there are no managerial champions for usability and no organizational structures or processes to support good interaction design.

The result is that despite the arrival of innovative, useful, and easy-to-use products such as the iPhone, a great many new products are simply horrible from a usability perspective. One new toaster has a digital display and a touch pad with 14 buttons. Some companies develop appliances, then outsource the supporting utility and application software to development firms that apparently lack any interaction design skill, resulting in promising appliances with disappointing supporting software. The 1980s joke about people not knowing how to program their VCR has now broadened into the sad fact that people cannot fully control many of the appliances they own, e.g., "My oven turned itself off in the middle of baking a turkey yesterday, but I have no idea why, so I took it over to a neighbor's house and had them bake it for me." Alan Cooper was right when he lamented that when computer technology is added to appliances, the result is usually more like a computer than an appliance (1999). One person we know, on discovering that his new high-end car became completely inoperable when its startup-controller crashed, returned it to the dealer, saying, "I wanted a *car*, but what you sold me is a *computer*."

It's Getting Worse, Not Better

While knowledge of good interaction design and usability engineering principles has definitely grown and spread over the past several decades, we daresay that the number of companies and organizations producing poor interactive products and services due to sheer ignorance is larger now than three decades ago.

Most software developers, especially newcomers, focus on software technology. They hire skilled software architects and programmers. Few consider the interaction that people have with technology as a subject matter requiring experience and expertise in its own right. Fewer make interaction the focus of their design efforts. Therefore, no matter how advanced the fields of human-computer interaction, interaction design, and usability become, that knowledge does not get to those who need it to make their products useful and usable. Instead, many product developers today consider the work of using technology to be a problem for the users, not a problem for the designers or their companies.

Previous Alerts to the Problem

We of course are not the first to complain about the poor state of software design and usability. Winograd (1996) presented perspectives and recommendations on the state and future of software design from several industry experts. Tom Landauer (1996), Don Norman (1999), and Alan Cooper (1999) each devoted an entire book to describing the problem and outlining their recommended solutions. One of us (Johnson, 2000), in a book on common graphical user-interface (GUI) design mistakes, included a chapter explaining that mistakes by management are the root cause of many GUI design mistakes.

However, those arguments concerned the state of software design before the above-described recent explosive growth of digital devices and services. Over the last decade, the problem has grown much worse.

We Haven't Hit Bottom Yet

Furthermore, the problem will continue to worsen. Although colleges, universities, technical book authors, and technical trainers are working hard to educate current and future software designers and developers in user-centered and task-centered design and usability testing, the ones being educated and trained are not the ones who make the decisions.

Middle and upper managers make the decisions, structure the organizations, allocate resources, and determine development processes, schedules, staffing, and budgets. Few managers—especially upper managers—are naturally endowed with the depth of insight that the late Steve Jobs had on the value of giving customers products and services that are productivity-enhancing, easy to use, and delightful. Most managers need education on that—lots of education. But, for the most part, they—especially upper managers—aren't getting any such education.

Thus, in most companies, organizations, and agencies, until we change managerial thinking and practice, we can educate and train software designers and engineers all we want, but they will be severely limited in what they can do to improve how software is designed.

Our Last Hope: Teach Management to Optimize UX

To turn things around, HCI experts must convince management of the benefits of optimizing the users' experience with digital products and services. (Another approach is for enough of us to become management, but that is harder because management requires temperaments, intuitions, and skills that many of us in the HCI field do not have. In such cases, we can hope that when we are in a position to do something, the forces at work on us in those positions will not prevent us from doing what we know must be done.)

Our previous books (Johnson, 2000; Johnson, 2007; Johnson, 2010; Johnson & Henderson, 2011) and training efforts have been aimed mainly at the engineers who design, develop, and test interactive products and services, but now we are expanding and re-directing our educational efforts toward management. One of us recently began presenting at management conferences and manager training events, and the two of us are planning a book on maximizing the user experience aimed precisely at managers.

Reaching managers requires more than just a change in vocabulary; it requires a change in mindset. Managers are generally not interested in the "how-to" details of task-focused design and usability testing; they have teams of designers and engineers for that. They need an understanding of the "what" and the "why," presented briefly, clearly, and succinctly. For example, they need to know the following:

- Most people are not trying to use technology; they are trying to achieve goals. They use technology only as a tool for that. If a tool presents too many obstacles, people won't use it.
- Few people become experts or even want to become experts with a tool. They just muddle through, learning just enough to achieve their goals.
- To be popular, a tool must do more than just help people perform a task. It must also be easy to acquire, easy to set up, easy to learn, easy to operate, easy to adjust, and fun to use. All of those aspects must be designed.
- Good interaction design runs deep. It cannot be layered on top of a bad conceptual design.
- Even highly technically-trained users need user-friendly tools, because their time is precious.
- Making technology-based tools simple is not simple. It requires thoughtful design, creativity, feedback, and iteration.
- A crucial part of design is understanding the users, their goals, and how the new tool will change and improve users' achievement of their goals. Understanding these things requires research.
- Developing a successful product or service requires either a coherent, task-focused conceptual model or a strong project leader with a clear vision of the desired result, preferably both.
- To develop successful products consistently, an organization must develop staff, infrastructure, and processes that support good user experience; otherwise, product success will be unpredictable and rare.

The above is what managers need to learn. But how do we teach it to them? How do we get the opportunity to educate managers? How do we even get an audience with them? Here are some ideas that can help:

- Develop extremely brief, succinct "elevator pitches" to allow you to capture their attention.
- Realize that their focus differs from yours. You are focused on usability and utility. They are focused on costs, revenue, and profits, and growth.
- Frame usability in terms of product-quality. That makes usability a process-improvement goal.
- Recognize that usability is not the only important issue. Be prepared to negotiate with other realities.

- Don’t just lecture them with textual bullet points. Show them—with graphs, images, video-clips, and customer-visits. Recognize that managers may not have seen their product or service in use. Tell them what you see. More importantly, listen to what they see.
- Better yet, let them hear from customers. Set up situations where people can tell—or show—them what life is like using your product to do the things that they are doing. Bad and good!
- Don’t try to teach managers everything at once. Develop brief units that can be delivered in 30 minute segments. It’s not that managers have a short attention-span; it is that they have many demands on their time. (You know you have captured a manager’s interest when he or she doesn’t leave or dismiss you when your scheduled time is up, but rather wants to keep talking.)
- Realize that organizational change usually happens slowly, and that organizations often resist change. Be prepared for resistance, and for change to occur in small steps.
- Lastly, consider becoming a manager yourself. Then you can talk with them as a peer.

If more people in the HCI and usability fields would turn their focus towards educating management, we might be able to turn the interactive products industry around. The dream is that many companies besides Apple will have people lined up outside stores awaiting new products and releases.

Well, we can dream, can’t we?

References

Cooper, A. (1999). *The inmates are running the asylum: Why high-tech products drive us crazy and how to restore the sanity*. Indianapolis, Indiana: SAMS Publishing.

Johnson, J. (2000). Management Bloopers. In Chapter 8, *GUI Bloopers: Don’ts and Dos for Software Developers and Web Designers*. Burlington, MA: Morgan-Kaufmann Publishers.

Johnson, J. (2007). *GUI Bloopers 2.0: Common User Interface Design Don’ts and Do’s.* Burlington, MA: Morgan-Kaufmann Publishers.

Johnson, J. (2010). *Designing with the mind in mind: Simple guide to understanding user interface design rules*. Burlington, MA: Morgan-Kaufmann Publishers.

Johnson, J., & Henderson, D.A. (2011). *Conceptual models: Core to good design*. Morgan & Claypool.

Landauer, T. (1996). *The trouble with computers: Usefulness, usability, and productivity*. Cambridge, MA: MIT Press.

Norman, D.A. (1999). *The invisible computer: Why good products can fail, the personal computer is so complex, and information appliances are the solution*. Cambridge, MA: MIT Press.

Schaffer, E. (2004). *Institutionalization of usability: A step-by-step guide*. Addison-Wesley Professional.

Winograd, T. (Ed.). (1996). *Bringing design to software*. New York: ACM Press.

About the Authors

Jeff Johnson
Jeff Johnson is President and Principal Consultant at UI Wizards, Inc., a product usability consulting firm he founded in 1996. Recently, he co-founded a second design consultancy: Wiser Usability, focusing on Web usability and accessibility for seniors. He has worked in the field of Human-Computer Interaction since 1978.

Austin Henderson

Austin Henderson's 45-year career in Human-Computer Interaction includes user interface research and architecture at MIT's Lincoln Laboratory, Bolt Beranek and Newman, Xerox Research (both PARC and EuroPARC), Apple Computer, and Pitney Bowes, as well as strategic industrial design with Fitch and his own Rivendel Consulting & Design.