Software User Research: Psychologist in the Software Industry

Christopher N. Chapman, Ph.D.
Microsoft Corporation*

* Opinions and statements expressed here are those of the author, and do not necessarily reflect the views, practices, or policies of Microsoft Corporation.

Introduction

"You went to work at Microsoft? What do you do, treat stress?"
"You'll hate a job in industry; it's all just about money."
Those statements are the most common reactions I heard from psychologist friends when I changed careers three years ago. I had a postdoctoral fellowship in a prominent research lab, a rewarding psychotherapy practice, and the hope of an academic and clinical career that I had been working towards during 14 years of college, graduate school, and postgraduate training.

My reasons for changing were diverse. I was tired of academic politics. My research interests did not have the focus that is so important for success in academia. My family did not wish to relocate (possibly several times) according to academic position. Finally, an alternative, full-time clinical career was uncertain – it seemed to risk both financial health and emotional burnout.

After focused thinking about my skills and interests, a great deal of research, and a lot of luck, I found a position that I love. I work as a usability engineer at Microsoft, where I help design software that better fits peoples' needs and abilities. My work affords me the opportunity to learn constantly, to express my passion for helping people, to engage in varied and stimulating research, and to have a rewarding career.

In this chapter, I write about my experience at Microsoft. My thoughts are also informed by contact with colleagues at other technology companies. I hope to communicate enough about my work to enable you to determine whether it would interest you, and to suggest how you can learn more about it.

Most new psychologists in my experience are unfamiliar with the specialized research areas of usability and human-computer interaction (HCI), so I provide a broad introduction to the field and suggestions for further reading. If

you do have first hand knowledge of HCI, you may be interested to read the comments about other roles for psychologists and notes on applying for jobs.

My suggestions and recommendations here apply to psychologists at all levels. I have colleagues who have transitioned to the software industry at all stages of careers, ranging from new graduates to tenured faculty. However, new psychologists often share characteristics that could make the software industry of particular interest: they are up to date on research methods, have a diverse and recent background in empirical research, are accustomed to adapting to dynamic situations, and may have enthusiasm for computer technology.

Psychologists at Software Companies

We may start broadly by thinking about many possible careers for psychologists at software companies (and I include Internet companies in this; Web sites are software). Psychologists at software companies include project managers, consultants, psychometricians, cognitive science researchers, human resources specialists, employee assistance providers, and technical writers.

In general, we may break those opportunities into four categories. First, there are positions that emphasize specific psychological training and skills, such as employee assistance, human resources, and psychometric test specialists. The first two of these are common to all large companies, but some readers may be unaware of the role of psychometrics in the software industry. Many large technology companies have specific curricula that allow specialists to become certified as engineers and technicians with specific skills in technology. For example, a Microsoft SQL Server administrator may wish to become certified as a "Microsoft Certified Database Administrator" (MCDBA); a Cisco network technician may wish to become a "Cisco Certified Internetwork Expert" (CCIE). These certifications are based on a mixture of coursework and formal testing. Psychometric specialists are often employed to design, develop, and validate these tests and related training materials.

Second, there are positions that are open to psychologists on the basis of general education and organizational skills. At many software companies, there is a title of "program manager" that encompasses a wide range of positions where people are responsible for managing the scope, features, specifications, and delivery of various projects. Some of these positions are highly technical and employ experts on specific technologies. Other positions are open to people from various backgrounds who have a proven ability to organize complex projects and guide them to success.

Third, there are some opportunities that engage psychologists because of previous clinical, statistical, or other experience that is important for specific projects. For instance, a group within a company may focus on sales opportunities within the healthcare or mental health industries; or a software company may specialize in products that are used by psychologists, such as statistical analysis packages or clinic management tools. Psychologists may be needed as consultants on these projects and as field sales representatives,

especially if the customers are other psychology researchers or mental health professionals.

Finally, there are opportunities for psychologists to engage in applied research that emphasizes skill in behavioral data collection and analysis. Many technology companies have dedicated research groups that offer opportunities to pursue research in artificial intelligence, natural language, cognition, perception, and other kinds of psychological and psychophysiological study; these opportunities are generally well-known to academic psychologists. However, there are opportunities for applied research within product development that are less obvious to most psychologists.

For the remainder of this chapter, I will talk about the field in which I work, an area of applied research known as "human-computer interaction." It is large, diverse in scope, composed largely of psychologists, and corresponds well with the skills that many psychologists have. Within the software industry, it is most commonly present as a particular area of specialization: "usability engineering." Although I focus on usability engineering, the other areas previously mentioned are also open to psychologists. Many of my comments about skills and job seeking apply equally well to those areas.

Usability Engineering

Usability engineers study the interactions between people and computer software systems. They study interactions with traditional software programs as well as other media, such as Web sites. The goal of usability engineering is to improve software so that it is easier to use, does what people want and need, and opens new possibilities for people. Usability engineers work in a variety of settings: at large technology companies such as Microsoft, IBM, and Apple; at specialized consumer and business software companies; at consulting groups that provide usability coverage for other companies; and in software groups within other companies and institutions, where they work on projects for which the user interface is important.

Historically, usability engineering derives from the field of human-computer interaction, which is a branch of the larger field of human factors engineering. Human factors engineering arose in the mid-twentieth century, when it became apparent that interaction design was a critical factor affecting people's ability to use complex mechanical systems. Humans have various cognitive strengths and limitations. When these are not taken into account, a system can be created that is nearly impossible for anyone to control reliably. In classic research, human factors engineers studied the layout of complex environments such as airplane cockpits and nuclear power plant control centers. They discovered that the application of empirical behavioral research methods could lead to better designs for these controls, which led to safer systems and improved performance.

As computer technology became widespread and increasingly important to businesses and consumers, the study of human interaction with technology was extended to the study of computer interaction. One early area of research

was computer interface hardware. Researchers investigated many areas of direct human-computer interaction, such as designing monitors, mice, keyboards, and other interfaces. In the early years of the software industry, developers focused primarily on traditional computer science issues such as efficient algorithms. However, as software systems became more complex, it was increasingly apparent that the industry – and individual products – would benefit from attention to behavioral issues. The field of usability engineering was born.

What is "usability"? The concept is amorphous, and a definition depends upon the product in question and the goals that one has for the product. Usability engineering focuses on how people *interact* with software; this makes it different from computer science, which focuses on theory and development of computer systems. In general, usability engineers tend to think about four questions:

- What are the goals that a user has, when interacting with a software product?
- How does a user accomplish those goals, and how difficult is it to perform the tasks that are involved?
- How does a user understand the processes involved with a product?
- What are the user's cognitive and emotional responses? Is he or she satisfied, excited, bored, confused, or angry?

The importance of these questions varies for different products. For instance, evaluation of a new video game would be likely to emphasize factors such as whether the game is fun and engrossing. Evaluation of an accounting system would be likely to ask whether operations can be completed quickly and accurately.

Usability engineers work as part of software development teams and interact closely with team members over the course of product development. The software "lifecycle" denotes the sequence of stages in a product's history. It begins with initial planning, moves through design, development, and testing, and culminates in product release and ongoing customer support.

Early in the lifecycle, a usability engineer (UE) can help to identify problems with current products that may be addressed by a new product or version. The UE can also assist the team to develop realistic usage "scenarios" (examples of how the software may be used for various tasks), based on customer data. As new software is developed, the UE may examine it in various stages of completion (various "builds" of the product) to give feedback to the team. When the product nears completion, it can be tested in a laboratory with representative users. After the software product is completed and released to customers, a UE may talk with customers to understand their problems, review customer support logs, and suggest ways to improve customer support and future versions.

Typical Activities in Usability

At Microsoft, I work as a usability engineer (UE) in the Hardware division on home networking products. We make devices that allow home users to

deploy networks, share network resources such as printers, share broadband Internet connections, and connect wirelessly to their networks. My job is to investigate how home users understand networks and to determine how we can improve the interaction experience with our products to better serve our customers (and win new customers). Prior to this position, I worked for 3 years as a UE in the Microsoft Windows Server product line. Windows Server is a product that supports centralized services that can be used by many other people ("client systems") at a time. Servers run applications such as large databases, email services, Web sites, Internet information routing, information storage, backup, and network security. In that position, I worked to make our software easier for network administrators to use.

Perhaps the most representative activity of a UE is a laboratory usability study. In a usability study, people from the community are recruited on the basis of their match to a target market for the software, and are invited to come and use the new software. In a typical session, a user is asked to perform representative tasks according to a predetermined plan, and is asked to "think aloud" about his or her actions while working. For example, in a home networking laboratory study, I might ask a user to set up a new network device and configure computers to connect to it, just as if he or she were setting up a new product at home.

In a lab study, the UE explains the value of the user's feedback and emphasizes that it is the software, not the user, that is being tested. The development team watches behind a one-way window. The UE records data on the user's ability to perform tasks, where he or she experiences problems, what causes the problems, how well the user understands the tasks, how long it takes to complete the tasks, and similar information. After a number of different people have performed the same kinds of tasks, the UE provides feedback to the development team on how users are performing, and what would improve the product.

However, laboratory studies are only one way that UEs may contribute to products. Laboratory studies represent a certain gold standard of behavioral research, but they are also costly, time-consuming, and unable to address some kinds of questions. Often there is no need to run a laboratory study; a UE may be able to answer a question about whether a design is adequate on the basis of past experience with customers. It is also important for UEs to understand their users' thought processes and real world work, and that information is best acquired through customer visits, field observations, interviews, or surveys. Finally, UEs can have large impact during the planning process; it is vital for UEs to be involved with software planners and developers as they detail the goals and features of a new product or version.

There is no typical week or month in this work. The software field is continually changing; every product is something new. We are always challenged to learn something new, to find new ways to understand our users, and to represent their needs to the development teams. However, during the course of a given month at Microsoft, I may perform activities like these:

- Meet with a team to review initial plans for a product that has not yet entered coding (actual programming)
- Conduct a laboratory study of a product that is nearing completion, to see how well users perform common tasks
- Review the specifications for a product from a different group and provide feedback to them on how my customers might respond to their product
- Invite a group of customers to Microsoft headquarters to participate in a focus group about new products
- Visit a customer's workplace to learn about the customer's usage of our products and related technologies
- Give a presentation of top usability issues to a product team or management
- Coach a project's manager on how he or she can better think about usability issues as part of the design and specification process
- Collect and analyze survey data from customers
- Engage with a group of usability engineers from other companies to discuss issues that are affecting the technology industry as a whole

A summary of common tasks for usability engineers is shown in Table 1.

[Insert Table 1 approximately here.]

The Joys of Usability

There are many things that I enjoy about my work as a usability engineer. The most important is that I am still in a "helping profession"; my job is to help customers by ensuring that we make the best software possible for them. The users with whom I work – home users who want to benefit from network technology – are people who are interested to use technology to make their lives richer, simpler, and more efficient. They benefit from technology that allows them better access to information and shared resources. By deploying home networks, they also are the de facto managers of technology on which they and their households come to depend. By assisting them with the often difficult tasks in networking, I help them and their households and families in many areas of life: work, study, entertainment, and other important activities and relationships that depend upon information and online contact.

Another rewarding aspect of a career in usability is the continual challenge and diversity of the work. As you may infer from reading the lists of activities previously mentioned, there is no single right or wrong way to do usability work. Usability engineers are accountable for finding ways to contribute to product design and success, but there is no blueprint for doing that. In the course of contributing, we are invariably drawn into acting as researchers, interviewers, designers, writers, students of new technology, and teachers about human behavior. We are consultants who constantly seek new ways to engage with both product developers and customers to yield the best possible software systems. I have never heard a usability engineer complain of boredom in his or her job!

Finally, the software development workplace can be personally exciting and rewarding. The industry is notorious for long work hours (an inevitable result when market-necessitated deadlines conflict with everyone's desire to get a product "right"). However, it is a field that employs smart and hard-working people, that emphasizes achievement over superficial appearance, and that rewards a love of learning. Salary and benefits are good, and there are many opportunities for strong performers to learn, grow, advance, or change positions across a career.

Thinking about a Career in Usability

How do you know whether usability might be the right career for you? There are two absolute prerequisites: you must have a passion for making technology better for people, and you must have strong behavioral research skills. One thing is generally unnecessary for usability engineers: detailed technical knowledge or experience. Programming experience or other deeply technical background may be helpful for some projects, but it is not necessary to work on most kinds of software, and most usability engineers are not trained in programming or computer science.

To help determine whether usability might appeal to you, consider the following questions:

- Are you interested in technology and do you enjoy working with it?
- When using software products or web sites, do you often think that they don't make sense? Could you suggest improvements?
- Do you enjoy working with people and trying to understand their needs and how they think?
- Do you enjoy laboratory research studying behavior?
- Do you have a strong research background, with a proven understanding of empirical methodology?
- Can you respond positively to dynamic work environments where you are responsible for finding ways to contribute amidst changing requirements and conflicting goals?

If the answers to those questions are "yes," then you might enjoy the work of a usability engineer.

Many psychologists have not worked in industry or high-tech settings before. It is important to consider the positive and negative aspects of a career in industry, compared to the more familiar realms of academic, research, and clinical careers.

Compared with academia, a corporate career offers fewer opportunities to achieve the personal status and public recognition that is afforded to professors and researchers. There is less security than one achieves with tenure, and one has less opportunity to follow one's own research interests. Work and projects are directed by the marketplace, not by the interests of science. Working hours and tasks are less flexible, and the work environment may be less idyllic than that of a college campus. On the positive side, work is clearly delineated and constrained; unlike much academic work, it does not blend into all hours of one's

life. Salary and benefits are likely to be better, especially early in a career; colleagues may be less likely to engage in political disputes that neglect one's accomplishments; and there is a clear sense of contributing to the success of a larger group. There can be more ability to choose where one lives, and one has the pleasure of seeing immediate real-world results from one's work.

When compared with clinical careers, careers in industry do not offer the emotional rewards that come from direct engagement in helping people with life's crises. Clinical careers offer possibilities to work for oneself, to function at high levels of maturity, to have great flexibility in working hours, styles, and locations, and to engage in many different kinds of work, including psychotherapy, supervision, teaching, and management. On the other hand, clinical careers can be difficult to start. They are financially uncertain and the work may lead to stressful situations with difficult clients, overwhelming personal responsibility, or professional isolation. Careers in the software industry offer clearer paths to success than clinical work, they bring the delight of creating new products, and they offer possibilities to help many people, even millions of people, through attention to the human needs of technology users.

How to Prepare for a Career in Usability

There are five activities that will assist you in learning more about possible careers in usability. First, learn as much as you can about usability work. I recommend that you start by reading two books. The first book I recommend is Donald Norman's *The Design of Everyday Things* (also published as *The Psychology of Everyday Things*). Norman gives an enjoyable introduction to the ideas that underlie good technology design. If those ideas interest you, I recommend that you learn more about the profession of usability engineering in Jakob Nielsen's *Usability Engineering*. Nielsen's work is the best overview of usability engineering in practice, and his book is quite readable (unlike many titles in the field, which are reference books or academic texts). Nielsen presents the basic theory of usability along with practical suggestions and exercises for learning how to do usability research.

Second, unless you have a very strong empirical research background, such as Ph.D. from an experimentally-oriented program, think about strengthening your research background. Usability engineering usually does not require advanced statistics, but it *does* require strong experimental design skills and the kind of intuition about sampling, power, and the importance of results that comes from substantial experience in behavioral research. In addition to the offerings of traditional graduate schools of psychology, you may be able to find courses that offer specific training in usability methodology at your local college or university. Such courses may be offered through a university's extension school or in various academic departments. You may wish to check course listings in Psychology (cognitive), Human Factors, Computer Science, Technical Communications, and related areas.

Third, perform some usability research of your own. Once you begin thinking about it, it is likely that opportunities will abound. For instance, whether

you are currently in school or working, it is likely that you, your department, your colleagues, or a professional group will have a Web site with important content for you and your peers; or perhaps you work with software that is important for your research or professional work. After reading Nielsen's book, think about the common tasks that one would need to perform with the Web site or software, and invite a few friends or colleagues to participate in a usability study. Collect data on their task performance. If possible, find the person or people responsible for the Web site or software and share your results with them. If this is not practical, you could instead visit some popular Web sites, such as Amazon.com or MSN.com. Identify as many positive and negative aspects about the user experience as you can, and list ways that the experience could be improved.

Fourth, research the software industry and any companies that may interest you. Very large companies such as Microsoft often have a number of books written about them. Among those books, some may cover a company's work environment or follow the course of a specific project. Those kinds of titles will help you to learn what it might be like to work there. For Microsoft, Randall Stross's *The Microsoft Way* talks about what Microsoft is like for employees and whom it hires, and Fred Moody's *I Sing the Body Electronic* follows the development process for a consumer software title. If you are interested in smaller companies or consulting firms, look up information about them online: do a search at your favorite search engine, and research them in the online business database at a university library.

Finally, go online and look at job postings. Most technology companies post positions online and recruit people through online submission and review of resumes. Reading these job postings will acquaint you with the specific requirements and expectations you would need to meet, to enter this career.

Three suggestions may assist with reviewing job postings. First, try a variety of key words when searching a company's database; not all companies use the same job titles or descriptive phrases. Usability engineers may be found with titles such as "usability engineer", "user testing", "human factors engineer", "user researcher", and even with generic titles such as "program manager." If you can do a keyword search, try "usability", "psychology", "user AND research", "human-computer", "human factors," and similar phrases.

Second, when reading job postings, remember that what they call a "requirement" is often not a requirement. Job postings are often written hastily, recycled from the past, or written by committee. If you read a posting that says "X skills, Y experience, and Z years of work are required," I suggest that you read it as "an ideal candidate would have X, Y, and Z … but other candidates will be considered." The most important thing is to be able to communicate honestly and effectively how you could contribute to the success of the company.

Third, if you apply, be prepared for a very unpredictable timeframe. The private job market does not operate on the kinds of fixed schedules that are found in graduate school and academic careers. After an applicant submits a resume online, a company may call within a day or two to conduct or schedule an interview. Alternatively, an application may languish and never receive a formal

rejection. It is important for an applicant to respond with flexibility and clarity to whatever situations may arise.

If you pursue a career in usability, you can be proud that you are helping to make people's lives better through technology. If you pursue other career paths, please let those of us in the software industry know when we can help you! It is only through learning about our users and their needs that we can continue our work to make software better.

References

Moody, F. (1996). *I Sing the Body Electronic: A Year With Microsoft on the Multimedia Frontier*. New York: Penguin.

Nielsen, J. (1994). *Usability Engineering*. San Francisco: Morgan Kaufmann.

Norman, D. A. (1988). *The Design of Everyday Things* (also published as *The Psychology of Everyday Things*). New York: Basic Books.

Stross, R. E. (1997). *The Microsoft Way: The Real Story of How the Company Outsmarts Its Competition*. Cambridge, MA: Perseus.

Table 1
Common work duties for usability engineers

Product research and development activities

Meet with a product team to assess what its members need to know from
customers and how user research can impact the product
Design and conduct a laboratory study to observe participants using software
products (experimental or quasi-experimental research)
Conduct a focus group to gather customers' feedback about their technology
needs
Design and deploy an online survey to assess customer needs, satisfaction, or
behavior
Visit customers' homes or workplaces to understand the context in which they
use technology products
Present research findings and recommendations to a product team and
management
Write study proposals and research reports
Review research proposals, plans, and results from other people, such as other
usability engineers or market researchers

Education for oneself and others

Learn about technology developments for new products
Coach a development team on how to understand customers and apply user
research data in the development process
Teach a class for new usability employees about a research method
Take a training course to expand skills with research methodologies or
technology
Mentor a new employee to help him or her perform better and have more impact
on product development

Business activities

Contract with an outside company that will provide specialized research services
Interview job candidates for usability, user research, and design positions
Develop and maintain research plans and associated budgets