Available online at www.sciencedirect.com
SCIENCE DIRECT®
ELSEVIER
Int. J. Human-Computer Studies 59 (2003) 497–520
International Journal of Human-Computer Studies
www.elsevier.com/locate/ijhcs

Issues and strategies for integrating HCI in masters level MIS and e-commerce programs

Susy S. Chan*, Rosalee J. Wolfe, Xiaowen Fang

School of Computer Science, Telecommunications, and Information Systems, DePaul University, 243 South Wabash Avenue, Chicago, IL 60604, USA

Received 15 November 2002; received in revised form 24 February 2003; accepted 31 March 2003

Abstract

Human–computer interaction (HCI) is an important knowledge component for graduate management information systems (MIS) and E-commerce (EC) programs. HCI topics, such as user-centered design and usability testing, have begun to receive increasing attention in MIS/EC curricula because of their importance in the development of Web-based solutions. This paper discusses issues and approaches for integrating HCI topics into masters level MIS/EC programs. Research on HCI topics related to MIS provides a theoretical foundation for student learning. By bridging research with these curricula, researchers are challenged to examine how HCI approaches can improve user acceptance of new systems. A case study illustrates how HCI topics can be taught as a stand-alone course or incorporated in existing MIS/EC courses. Drawing from the case study, the paper also addresses pedagogical challenges regarding student skill sets, learning outcomes, innovative pedagogies, tools and technology, and HCI issues for advanced IS/EC topics.

Keywords: Human–computer interaction (HCI); Usability; E-commerce; MIS; Masters curriculum; User-centered design

1. Introduction

The Web has become a widely accepted platform for delivering interactive applications for customers and for supporting inter-/intra-organizational processes.

*Corresponding author.
E-mail addresses: schan@cs.depaul.edu (S.S. Chan), wolfe@cs.depaul.edu (R.J. Wolfe), xfang@cs.depaul.edu (X. Fang).

1071-5819/03/$ - see front matter
doi:10.1016/S1071-5819(03)00110-1

Web-based systems emphasize effective user-interface design and information architecture. HCI concepts, such as user-centered design and usability engineering, play an increasingly important role in the development of Internet-based solutions (Lazar et al., 2002). Researchers have acknowledged the growing importance of HCI research in the MIS discipline (Zhang et al., 2002). The wide adoption of Internet and the advancement of technology affect more users in their daily life at work and home as they interact with information technology (IT). Making IT systems easy to use and useful becomes more important particularly in an Internet-based environment. The increasing number of interface developers needed to support such interactions attests this trend (Galletta, 2002). The MIS/HCI research agenda proposed at the AMCIS 2002 conference emphasizes a broad range of HCI perspectives in developing and deploying technologies to ensure the holistic experience of human interaction with technology (Zhang et al., 2002).

An interactive relationship exists between MIS/HCI research and teaching. Rigorous HCI research can enhance the theoretical foundations of graduate programs aimed at preparing students for technical and managerial careers in the MIS or e-commerce (MIS/EC) fields. On one hand, findings from traditional HCI and MIS/HCI studies, such as the technology acceptance model (Davis, 1989; Venkatesh and Davis, 1996), can help students build or manage systems to increase user adoption. On the other hand, incorporating HCI research in MIS/EC curricula helps to disseminate the most advanced theories and findings to a broader audience who are usually not in the HCI field. From a curriculum perspective, integrating HCI topics in MIS/EC curricula may challenge researchers to re-examine the unique contribution of HCI methods in system development, relationship building and management (Carey, 2002), and task-technology fit (Strong, 2002). Discussion of HCI in the MIS/EC context may also inspire researchers to view future HCI research from a macro-level, instead of a traditional micro-level.

Several model curricula in the MIS discipline have started to address HCI issues (Lidtke et al., 1999; Gorgone and Gray, 2000). HCI is especially important to a Web-centric curriculum (Gorgone and Kanabar, 2002). However, based on a review of 48 masters level programs, discussed later in the paper, only a few MIS and EC programs currently offer HCI courses. Only one MBA program has included one HCI course.

The objectives of this paper are to examine the integration of HCI topics in masters level MIS/EC programs and to identify strategies for such integration. Parallel to the MIS/HCI research panel, a study of curriculum practice shows the current response to the shifting market demands in the MIS and EC fields. In this paper, we examine broad curricular issues facing the adoption of HCI topics in MIS/EC programs and discuss two approaches for integration. One approach is to offer a stand-alone HCI course. Another approach is to incorporate HCI topics in various existing courses. We then present a case study to illustrate how these approaches can be implemented. Furthermore, the paper addresses pedagogical challenges in teaching HCI topics in areas of student skill sets, learning outcomes, innovative teaching methods, tools and technology for supporting HCI teaching, and topics of emerging importance.

2. HCI in masters level MIS/EC curriculum

Masters level programs in MIS or EC typically have separate aims for different audiences. MBA degrees with concentrations in MIS or EC prepare students for managerial careers. The technically oriented programs, typically offered in the form of Masters of Science, focus on professional careers as developers, analysts, and project leaders. Neither type of degree prepares students as HCI specialists, but requires some level of understanding of HCI issues. Research oriented masters programs leading toward Ph.D. degrees, however, are excluded in this paper.

Employers expect MIS/EC graduate students to have training in making critical business and technology decisions when they take on the roles of analyst, developer, IT manager, and project leader (Lidtke et al., 1999). Clearly, usability is one criterion that cannot be ignored for software products (Madsen, 1999; Vredenburg, 1999; Rosenbaum et al., 2002). The advent of Internet and e-commerce has heightened the need for IT professionals with knowledge in usability and broad HCI concepts. It is therefore essential to address the awareness of usability and HCI-related issues in MIS/EC curricula through effective strategies. IT professionals with managerial responsibilities will need to understand the HCI issues as they pertain to the selection of software products, the implications of usability for marketing and e-commerce (e.g., Kalin, 1999), and the role of HCI specialists to perform in a development project (Burdman, 1999). For technically oriented IT professionals, an understanding of HCI issues enable them to develop more effective systems, and work more effectively with HCI specialists in collaborative development efforts (Hansen et al., 2001).

Employers have begun to recognize the high costs that resulted from the failure of incorporating user-centered design in Web-based information systems (Kalin, 1999). Furthermore, the new generation of application development has a strong HCI dimension (Hansen et al., 2001) and requires a dynamic and complex IT skills portfolio (Gomoiski et al., 2001). To ensure that the developers build user-centered Web systems, some companies have even brought developers to usability labs to observe how customers interact with the sites they built (Tedeschi, 2002). Companies are likely to value developers and analysts with strong HCI knowledge for application development teams. For those companies relying on packaged applications or outsourced development projects, HCI knowledge can enable companies to make better decisions on vendor and product selection as well as the development of Web portals that assimilate information from multiple enterprise applications.

In order to approach strategies for teaching HCI concepts, this paper examines the desired HCI learning objectives, how other non-HCI programs, such as computer science (CS) programs, integrate HCI-related discussions, and what topics should be covered in MIS/EC curricula.

2.1. Learning objectives for HCI

Recognizing a need to educate students in this field, in 1988 the ACM Special Interest Group on Computer–Human Interaction (SIGCHI) created a Curricular

Development Group whose goal was to produce a set of recommendations for teaching HCI. In 1992, this group published the *ACM SIGCHI Curricula for Human–Computer Interaction* (Hewett et al., 1992). In this publication, they laid out a basic approach to HCI education, including learning objectives and goals for acquired skills. The group identified 5 content areas and 12 learning objectives shared by *all* survey courses in HCI (see Fig. 1). One of the content areas covers "human information processing" which includes human factors theory and communication. Additional content areas address the study of computers in the context of human work, specifications on computer systems and architecture as they relate to the human–computer interfaces, and a discussion of design, implementation, and evaluation strategies.

The learning objectives serve as guidelines for positioning the content areas within the context of a course. In terms of acquired skills, students should recognize and recall pertinent terminology, use concepts to explain and analyse specific situations, and apply these concepts in solving problems in real-life scenarios.

2.2. HCI in model IS curricula

An interdisciplinary group developed the ACM curriculum (Hewett et al., 1992), but basically its creation was in the context of CS programs and appears as one of the core disciplines in the ACM computer science curriculum. In contrast, HCI is absent in the *Curriculum Model 2000 of the Information Resource Management Association and the Data Administration Managers Association* (IRMA/DAMA) (Cohen, 2000). The IRMA/DAMA curriculum model focuses on managerial perspectives in organizing the knowledge components required for the management of information and technology resources. Information users are identified, but not in the specific context of HCI as defined in the ACM SIGCHI framework.

Recently, however, HCI topics have appeared in two relevant IS curricula. HCI is an important knowledge component in both the *Model Curriculum and Guidelines for Graduate Degree Programs in Information Systems* (MSIS 2000) presented by ACM and AIS (Gorgone and Gray, 2000) and the *Information Systems-Centric Curriculum* (ISCC '99) (Lidtke et al., 1999). The MSIS 2000 curriculum identifies human factors as a career elective course. Reflecting the perspectives of industry practitioners, the ISCC '99 curriculum emphasizes human behavior and computer interaction as an industry defined attribute of an IS graduate, and HCI as one of the knowledge components. Its proposed course on HCI draws largely on the themes of the 1992 SIGCHI *Curricula for Human–Computer Interaction* (Hewett et al., 1992), including:

- The diversity of users and tasks, and their impact on design.
- The cost-benefit tradeoffs in user-centered design.
- How user-centered concerns can be incorporated into system development life cycles.
- The need to evaluate system usability.
- The existence of design, implementation, and evaluation tools.

Content areas of HCI	Learning Objectives
1. The Nature of HCI (Meta-) models of HCI 2. Use and Context of Computers Human social organization and work Application Areas Human-machine fit and adaptation 3. Human Characteristics Human information processing Language, communication and interaction Ergonomics 4. Computer system and interface architecture Input and output devices Dialogue techniques Dialogue techniques Dialogue genre Computer Graphics Dialog architecture 5. Development Process Design Approaches Implementation techniques Evaluation techniques Example systems and case studies	The student should become familiar with 1. The scope of issues affecting human-computer interaction, 2. The importance of the user interface to motivate the study of topics like HCI and user interfaces, 3. The impact of good and bad user interfaces, 4. The diversity of users and tasks (applications) and their impact on the design of user interfaces, 5. The limits of knowledge of individuals developing HCI systems, 6. The need to work with others, skilled in diverse areas such as software engineering, human factors, technical communication, statistics, graphic design, etc. 7. Cost/benefit trade-offs in HCI design, 8. Different system development life cycles including those particularly applicable to HCI systems (e.g., iterative design, implementation, evaluation, and prototyping), 9. How HCI concerns can be incorporated into systems development life cycles, 10. The need to evaluate systemusability (e.g., someone will evaluate usability even if not the developer, and, in some cases, not evaluating constitutes professional misconduct), 11. The existence of design, implementation, and evaluation tools for developers with diverse needs and technical expertise, and 12. The information sources available on HCI.

Fig. 1. The learning objectives shared by HCI courses (Hewett et al., 1992).

2.3. A survey of HCI programs and courses

To understand the role of HCI in graduate MIS/EC programs, we have identified 48 masters level programs that offer HCI majors, concentrations, or courses as listed

in the HCI Bibliography (www.hcibib.org) and ISWorld Net (www.isworld.org). Our search of the HCI Bibliography site yielded 56 institutions offering some form of HCI program. In November 2002 the list of ISWorld Net showed 148 institutions offering masters level MIS programs. In addition, there is a short list of e-commerce courses posted on ISWorld Net. After examining curriculum descriptions published on these institutions' official Web sites, we narrowed the list down to 48 entries based on three criteria. Each institution must offer (a) masters degree programs in HCI, (b) a concentration in HCI as part of other masters programs, or (c) HCI or human factor courses as part of a non-HCI masters program. The 48 entries, however, represent only 46 institutions because each of two institutions has two programs that meet the selection criteria.

Table 1 shows the interdisciplinary nature of HCI as an academic field. This finding is consistent with the reference disciplines identified by the AMCIS 2002 panel (Zhang et al., 2002). Many institutions have collaborated among different disciplines, such as computer science, psychology, engineering, information science, and management science, in order to offer HCI programs. Of the schools or departments that are responsible for administering the programs or courses, Computer Science (41.7%), Information Science and Management (16.7%), and Psychology (12.5%) have demonstrated the greatest interests in offering HCI courses. In comparison, only 4 programs in MIS/IS and 2 in EC (12.5% in combination) include HCI courses.

These 48 entries can also be grouped into three categories—15 institutions (31.3%) offer M.S. or M.A. degrees in HCI, 8 institutions (16.7%) offer masters level concentrations in HCI, and 25 institutions (52%) offer HCI-related topics either as required or elective courses. All six M.S. programs in MIS/EC cover HCI topics as

Table 1
HCI offerings by school or department

School/Department administering HCI program	M.S. or M.A. in HCI programs	M.S. programs offering HCI concentrations	Masters program offering HCI courses	Combined
Computer Science	4	3	13	20
Library and Information Science	4	1	3	8
Psychology	3	1	2	6
Information Systems/ E-commerce			6	6
Engineering/Industrial Engineering	2	1		3
Architecture		1	1	2
Art		1		1
Design	1			1
HCI	1			1
Subtotal	15	8	25	48

elective or required courses. Only one institution, among the 48 entries, includes a usability course as an elective for its MBA degree in e-business. This finding suggests that HCI topics have begun to move into the MIS/EC curriculum, but have not yet gained support in the MBA program.

From these 48 entries, we examined the HCI courses offered at 15 US institutions. Out of this smaller group, 5 institutions offer M.S. in HCI degrees and 10 offer one or more HCI courses in M.S. in IS, MIS, CIS (computer information systems), EC, or IM (information management) degrees. This smaller sample enabled us to compare the differences between requirements of the HCI profession (i.e., the full HCI degree) and the needs of HCI knowledge of professional developers, analysts, and project leaders (i.e., HCI courses in MIS/EC programs.).

These courses, as shown in Table 2, address six broad HCI themes: (1) human factors theory involving human perception, cognition, and memory; (2) user-centered design, evaluation, user testing, and usability; (3) information architecture and content organization; (4) communication through visual organization (graphic design, interface layout, and visualization); (5) implementation of and development tools, including interface prototyping; and (6) AI-related topics such as data mining, intelligent agents, and support software for on-line and collaborative communities.

All six themes correspond to the content area and learning objectives of the SIGCHI Curricular Development Group as outlined in Fig. 1. The first, human factors theory, falls into the content area of ''Use and Context of Computers'', and the learning objective of understanding the diversity of users and tasks. The second, user-centered design and evaluation, appears in the content area of ''Development Process'' and also in the objectives covering knowledge of system development lifecycles, and the need to evaluate system usability. The third theme, information architecture and content organization, and the fourth, communication through visual organization, are both examples of design approaches under the area of ''Development Process,'' but the fourth is also considered in the content area of ''Human Characteristics''. Themes 5 and 6, prototyping and AI-related topics, are part of the content area of ''Development Process,'' and play an important role in the learning objective covering knowledge of the existence of design, implementation, and evaluation tools.

Some of these themes, such as human factors, cognitions, interface design, and HCI theories, correspond to taxonomy identified in previous studies (Beard and Peterson, 1988; Carey, 1988). Other themes, such as on-line community, usability, AI-related topics, and information architecture, have emerged as results of new technologies. After comparing HCI courses in MIS/EC programs to those included in M.S. in HCI programs, it is clear that HCI courses in MIS/EC programs have focused mostly on broad HCI and human factors theories. There are fewer HCI courses focused on the Internet technology's impact on the analysis and development process and new forms of system–user interactions.

This analysis shows that MIS and EC programs have only begun to recognize the importance of HCI topics in masters level programs. Many barriers exist. Computer scientists have been struggling to integrate HCI into CS curricula for a long time because CS mainstream has been slow to recognize the importance of HCI. HCI

Table 2
Themes of HCI curses offered in selected institutions

Themes of HCI courses	In MS/HCI programs	In MS/IS and MS/EC programs
1. Human factor theory		
• Cognition, Info usage, learning strategies	3	
• Human computer interactions	3	6
• HCI advanced topics	2	
• Human factors	3	3
2. User-centered design, evaluation, user testing, and usability		
• User centered design	3	
• Usability	1	2
• Testing, assessment, evaluation	2	
• Design and analysis	1	
3. Information architecture and content organization		
• Information architecture	2	
• Digital formatting	2	
4. Communication through visual organization		
• Graphics and HCI	1	
• Interface/interaction design/GUI	5	1
• Visualization	3	
5. Implementation and development tools		
• Interface programming	3	
• Projects/internships/capstone	5	
• Database applications	1	
• Prototyping	2	
6. AI-related topics and on line communities		
• Knowledge based products	1	
• Natural language processing	1	1
• Online community		
• Computer supported cooperative work	1	

content is still not adequately represented although many CS degree programs do offer a HCI course as elective (Douglas et al., 2002). Educators in software engineering report similar experience and have proposed integration of HCI topics into software engineering courses (Perlman, 1995; Faulkner and Culwin, 2000; Weinberg and Stephen, 2002). Possible integration approaches include: (a) offering an introductory HCI course in the early stage of a program and incorporating additional HCI concepts in advanced courses, and (b) customizing contents to fit different student populations with an emphasis on implementation of interfaces (Scown and McManus, 1995).

The CS experience may be applicable to MIS and EC programs. In technically oriented programs for careers as system developer and analyst, a separate HCI

course needs to cover more than general topics as part of an overview. Students need to learn how to integrate HCI approaches in the systems development life cycle (SDLC). For MBA programs, students should develop a clear understanding about the impact of usability on Web applications (Kalin, 1999) and the interdisciplinary and collaborative nature of Web-based systems development process (Burdman, 1999) in order to make appropriate cost-benefit analysis and selection of tools.

2.4. HCI topics

The three published curriculum models as discussed above (MSIS 2000, ISCC '99, and ACM SIGCHI) have identified topics for a complete curriculum as well as for a single HCI course. In light of the gap between expert recommendations and the current state of implementation across masters level MIS/EC programs, we believe a more pragmatic approach should be considered. The wide adoption of the Internet and the Web as a common platform for service and information delivery are further evidence that graduate MIS/EC programs should cover the following HCI topics:

- *The importance of interface design and cost effectiveness analysis of user-centered design*: How should interface design and usability testing be incorporated into the development process? These topics can be included in systems analysis and design courses.
- *A survey of methods frequently used by the HCI profession, such as usability testing and prototyping*: Because most MIS/EC students will not become HCI professionals, it is not necessary to cover these methods in great depth. While they are not likely to perform these tasks by themselves in their jobs, they should be able to apply their knowledge to justify the necessity of employing any of these HCI methods.
- *Design guidelines*: The discussion of general design principles and how to use design guidelines provides students foundational knowledge about interface design. The guidelines should cover a wide range of issues including accessibility and trust.
- *Usability for Web site engineering*: The discussion of usability issues should focus primarily on Web-based solutions, but students should also have some exposure to the usability of general software products.

These topics can be incorporated in a required or elective HCI course. For programs that do not have rooms for a separate course, these topics could be incorporated into an existing systems analysis and design course to augment the coverage of systems development life cycle.

2.5. Differences in HCI and traditional MIS methodologies

Many may argue that the HCI concepts and methods are similar to the MIS approach; it is merely a matter of different terminology. However, HCI supports a radically different *view* of the software development life cycle. The HCI approach is called *user-centered development.* Textbooks covering user-centered development, as

part of HCI (Preece et al., 1994) or usability engineering (Nielsen, 1993; Mayhew, 1999), show a great deal of consensus when comparing the user-centered and traditional software development methodologies. These distinctions are intrinsic to user-centered design and are what impart its unique perspective. These differences can be categorized according to three major themes, which are (1) an emphasis on users, (2) a highly iterative nature, and (3) empirical testing.

2.5.1. *Emphasis on users*

In user-centered development, users are involved in all stages of the process. This not only includes analysis, design, and development stages, but also extends to order processing, customer support, and advertising. Instead of being technology driven, user-centered design is user driven. User requirements and user interface design should be the impetus for technical process rather than be an incidental aspect or an add-on that occurs late in the process (Mayhew, 1999).

During the analysis stage, the user-centered methodology employs many interviewing techniques in the quest to "know the user". Many of them are carried out with individual users in a face-to-face setting in the context of the user's work. These techniques include observing users as they complete tasks on the job and may involve "thinking aloud" or "talk right after." Other interviewing techniques probe a user's mental model of the information associated with the tasks, and involve some type of card sorting. Researchers in HCI concur that the traditional marketing techniques of focus groups and questionnaires are less useful because they take place out of the context of the actual tasks (Hackos and Redish, 1998).

The MIS approach to systems analysis and design typically views users as groups of individuals in organization settings. These users may share similar job functions or business goals. The granularity of the HCI approach to user and task analyses is not the focus of MIS analysis and design courses, as evident in the content covered in the systems analysis and design textbooks (e.g., Dennis and Wixom, 2000; Maracas, 2001; Whitten et al., 2001; Hoffer et al., 2002; Kendall and Kendall, 2002; Satzinger et al., 2002). Furthermore, the HCI approach emphasizes examining broader range of users, not limited to normal adults performing job related tasks. Business-to-consumer interface design has to consider users of diverse background, skills, and needs. Developers and analysts in the future will need to take a closer look not only at the user's cognitive aspect of experience, but also their holistic experience, such as playfulness (e.g., Agarwal and Karahanna, 2000), emotion (Venkatesh, 2000), and flow experience (e.g., Trevino and Webster, 1992). Studies on the technology acceptance model (Davis, 1989) have revealed a stronger impact of perceived usefulness than perceived ease of use on usage intention (Venkatesh and Davis, 2000). Recent research, however, has shown that user perception of a system's ease of use is related to the design guidelines (Lederer et al., 2000), task-technology fit (Goodhue and Thompson, 1995; Dishaw and Strong, 1998, 1999; Mathieson and Keil, 1998), and when interfaces are intrinsic to the task (Gefen and Straub, 2000). These MIS/HCI studies should be incorporated into user and task analysis during the requirement phase. Thus, the HCI approach helps to address the need to expand

the MIS focus to encompass a broader range of users and their holistic experience (Zhang et al., 2002).

2.5.2. Highly iterative

Traditional software development places emphasis on ensuring that any implementation conforms to the design specifications. Testing validates that the implementation correctly carries out all of the requirements of the design specifications.

Conformance to the design specification is important, of course, but all too often, defects in the design specification itself will cause the product or system to fail. This problem arises because in traditional development, users interact with a system only at the beta-test stages, after the system is implemented. Unfortunately, it is only through this interaction of users and system that unanticipated design requirements come to light, and in most cases, there are few remaining resources to address these newly discovered problems (Nielsen, 1993; Rosson and Carroll, 2002).

User-centered development employs techniques to break this conundrum of "We can't test it until it's implemented, but after it's implemented, we can't change it." These techniques involve *iterative refinement* to not only the implementation but to the design specification (Preece et al., 1994). *Prototyping* features heavily in these approaches, where models or prototypes serve in place of a fully functional system. Prototypes, especially paper prototypes, require far fewer resources to build and can be changed easily (Rettig, 1994).

In contrast to the traditional waterfall model of SDLC, the MIS approach has evolved to become more iterative, but still not to the degree of iterative process emphasized by the HCI profession. Rapid application development (RAD) can accelerate SDLC by bringing users into the requirement phase early with rapid prototyping tools. However, the emphasis of high fidelity prototypes (instead of paper prototypes) and the absence of empirical testing, as discussed in the following section, differentiate RAD from a true HCI approach.

2.5.3. Empirical testing

In user-centered development, conformance to measurable human factors is central to validation. Conformance is determined by empirical testing of the system with "appropriate users performing representative tasks" (Hix and Hartson, 1993). Examples of measures include speed of task completion, number of errors committed while performing the task, time required to learn to perform the task, and subjective satisfaction (Shneiderman, 1997).

Specifications for these measures, called "usability goals" are not estimated by guessing. Instead, these specifications are determined by comparison to other means of accomplishing the task. Comparisons may involve testing the current product or a competitor's product. If there is no extant product, it may involve measuring the current task as it is performed manually (Vredenburg et al., 2002).

Results from empirical user tests help designers and developers in making decisions on how to change design specifications and implementation.

In user-centered development, users (customers) define whether a product is successful or not, so it is important to provide the best customer experience possible.

This is in contrast to the traditional MIS-oriented philosophy that a proper system analysis will assure system usability. User-centered development posits that it is not possible to anticipate all users needs a priori, thus empirical testing is essential for fostering system usability. Another way to look at this is to examine the relationship of testing and system requirements in the development process. In the traditional waterfall SDLC model, testing evaluates the adherence of an implementation to the project specification. In user-centered development, empirical testing evaluates the project specification in addition to the implementation.

In fact, all three differences center on the role of users. This difference in development philosophy may be one of the reasons that there has been slow progress in integrating HCI topics into MIS/EC curricula. An open area for MIS/HCI researchers is examining how usability, system usefulness, and task-technology fit can be incorporated into the analysis and design for building more effective systems.

3. Approaches for integration: a case study

To incorporate HCI topics into MIS/EC curricula, we suggest a combination of two approaches:

- Create a required HCI course that covers a broad range of topics. This could be designed as an introductory course presuming no prerequisites.
- Incorporate key HCI topics, such as usability testing and user-centered design, in existing courses in Web design (Lazar, 2000), information systems (Khoo, 1999), and e-commerce (De Villiers, 2001).

Neither approach requires the restructuring of an entire curriculum because no specific prerequisites would be required and the discussion of HCI issues could be embedded within existing topics of advanced courses.

We have implemented both approaches in the M.S. program in E-Commerce Technology at the School of Computer Science, Telecommunications, and Information Systems at DePaul University. These courses are open to students majoring in IS, EC, HCI, and CS.

3.1. Offer an HCI course

We have introduced a required course on *Usability Issues for E-Commerce* in the advanced phase of an EC curriculum at our institution. This course provides an overview of the user-centered development process with an emphasis on Web site design, prototyping, usability testing, and guidelines outlining special consideration for the novice user (see Table 3). By using a combination of case studies, discussions of general concepts, and applications to solve specific usability problems, this course covers the four HCI topics discussed in Section 2.4.

Table 3
Weekly topics for *Usability Issues for E-Commerce*

Week	Topics
1	Benefits, User and Task Analysis, Usability Goals
2	Evaluation, Informed Consent
3	Usability Testing
4	Gathering Data, Content Organization, Card Sort
5	Visual Organization
6	Navigation, Design Principles
7	Text, Design Methods, Prototyping
8	Color, Bandwidth Reduction
9	Multimedia, Globalization
10	Accessibility, Trust

Case studies are used to demonstrate how user-centered techniques can be incorporated into existing development methodologies while keeping additional impact on resources to a minimum. Addressing the needs and priorities of novice users is particularly important because Web site users are typically impatient, have a low tolerance for frustration, and are likely to leave a site quickly if they encounter difficulties in finding what they want (Nielsen, 2000). The course utilizes the *paper* prototyping technique and a commercially available Web authoring tool for building prototypes for user testing. Thus students have an introduction to all aspects of user-centered design as it applies to Web site development.

To unify homework assignments, the instructor gives students a goal of redesigning Web sites that students created in previous courses. Each homework assignment covers a different part of the user-centered design process including user and task analysis, researching and setting usability measures, information architecture, screen layouts, and evaluation involving user testing. Students apply the card sort technique to discern a user's mental model for content organization, navigation, labeling, and searching systems (Nielsen and Sano, 1994; Rosenfeld and Morville, 1998). For the final projects, students choose Web authoring tools and create a redesigned version of the Web site.

3.2. Incorporate HCI topics in existing courses

In a core course on *Design and Strategies for E-Commerce*, we incorporate some of the usability issues in the Web site engineering method (Powell, 1998). This course surveys business-to-consumer e-commerce models and supporting Internet technologies. The goal of this course is to prepare Web developers with hands-on experience in Web site development in the context of business to consumer strategies. As shown in Table 4, students in this course learn the concepts of business models, consumer behaviors, the e-commerce value chain, on-line shopping, order fulfillment, digital payment systems, security, marketing, and e-commerce technologies. Concurrently, they apply these concepts in team projects by creating business to consumer Web sites following the Web site engineering method. Each team develops a business plan,

Table 4
Weekly topics for *Design and Strategies for E-Commerce* (Chan and Wolfe, 2000)

Week	Topics	Project deliverables
1	EC Framework, Web Site Engineering	A. Team Organization Statement
2	EC Value Chain, Internet Consumer Behaviors, B2C Value Propositions	
3	B2C Business Models	B. Business Case Statement: Value Proposition, Business Models, Market Analysis, and Intended Users
4	On-line Transaction, Shopping Cart, Usability, Requirement Analysis, Information Architecture	C: Requirements and Information Architecture Chart, Process Flow Chart, Page Definition
5	EC Technologies, Interface and Navigation Design	
6	B2B and Intranets, Prototyping	D: Detailed Design, Site Layout (sketches), Graphic and Interface Design, ERD
7	Transaction Security, Digital Payment	
8	Marketing, Promotion, and Maintenance of Web Site, Personalization, Search Engines, Site Usage Analysis and Metrics	
9	Implementation, Globalization and Legal Issues, Feasibility	E: Feasibility Report
10	Future Trends	
11	Presentation	F: Final Project

audience profile, information architecture, navigation, programming, prototyping, implementation, and site promotion and maintenance strategies (Chan and Wolfe, 2000).

All the teams follow a tight schedule of 11 weeks. Student teams have to meet the requirements of six project deliverables: (a) a team organization statement, (b) a business case statement, (c) a requirement report with an information architecture chart and transaction process flowchart for shopping cart and order fulfillment, (d) graphic design, Web page layout (in sketches), and navigation design, (e) feasibility and implementation plan, and (f) an in-class demo of a fully functional prototype site.

Students learn user-centered design through the course-long project. In deliverable B they have to articulate the user profile and intended tasks. In deliverable C they learn to produce the information architecture chart based on assumptions or data about users. In deliverable D they apply methods for graphic design, navigation, and interface in producing paper prototypes and page layout. However, students often do not perform usability testing because of time constraints. This course exposes students to the full process of Web site engineering with an added focus on

user-centered design. They develop an in-depth understanding of the usability testing process and techniques in the *Usability* course.

Additionally, we incorporate HCI-related topics in a course on *Advanced Topics for Server-Side Scripting.* This advanced course covers *n*-tier design, COM components, and other scripting techniques. Besides covering technical issues, the instructor introduces discussions on usability issues in several ways. In the discussion on tiered design, students learn that separating the interface layer from other layers makes it possible for interface designers to get involved in the early stage of development. Later, students learn how to use COM components in order to better comply with certain usability guidelines. For instance, interface features implemented by COM components can result in higher consistency. The pervasive discussion on HCI topics in the curriculum enhances student learning by emphasizing the linkage between HCI and MIS/EC.

It is essential to incorporate HCI-related discussions continuously in advanced MIS/EC courses, when covering such topics as project management, Web data mining, intranet and extranet development, and systems analysis and design. Through case studies in a project management course, students can examine the process and challenges of developing Web-based solutions through collaborative teamwork and the roles HCI specialists play in application development. This approach may be suitable for MBA students. For example, in a Web data-mining course, students can apply HCI principles to customize information presentation, search, and interface design based on user profile and data from collaborative filtering. In the development of intranet and extranet systems, students can apply design guidelines and usability testing to enterprise portals, and more data-centric Web sites in contrast to Web sites for consumers.

An IS course on systems analysis and design is a natural place to address HCI topics. User and task analysis, interface design, and prototyping are congruent with the SDLC and RAD approaches. However, traditional systems analysis and design courses and textbooks tend to focus primarily on process and data modeling methods from a business and organizational perspective; HCI topics are addressed mostly in the interface design stage, late in the process. As the Web becomes a prevalent vehicle for information access and exchange, it is important for analysts and developers to understand HCI approaches and user-centered design. Therefore, SDLC can be augmented with user/task analysis, information architecture, paper prototyping, and usability testing. Incorporating these HCI topics requires rearranging the topics currently covered in systems analysis and design courses.

4. Pedagogical challenges and strategies

Teaching HCI in graduate MIS/EC programs presents many challenges. Drawing from our own experiences and experimentation since 1999 (Chan and Wolfe, 1999), we examine these challenges and offer suggestions regarding students, learning outcomes, innovative pedagogies, tools, and technology.

4.1. Student skill sets

MIS/EC students typically possess a strong aptitude toward business strategies, systems analysis and development, coding, and databases. Yet, often they are weak in visual design, cognitive psychology, or patience for the iterative design process. They tend to approach design and analysis from a broad organizational perspective, rather than a user's perspective. These students typically are capable of following the phased approach of SDLC, but have trouble adjusting to the loosely structured collaborative environment of HCI. When teaching HCI topics, this general profile of MIS/EC students should be taken into consideration. The instructor needs to compare the value propositions for users to those for the organization. For team projects, students of complementary skill sets should be assigned to the same team.

We have found several useful strategies when teaching HCI topics to MIS/EC students. These include (a) teaching evaluation first, (b) leveraging student skills, and (c) placing any HCI courses early in the curriculum.

4.1.1. Teaching evaluation first

An instructor should introduce evaluation methods before design in order to compensate for the students' lack of basic HCI skills and motivate their study of usability and user-centered design. For example, an instructor can start off a course with an assignment evaluating existing Web sites. Witnessing a series of users struggle with a site helps to convince students of the importance of usability. Students also learn from redesigning existing Web sites with poor usability.

4.1.2. Leveraging student skills

When a course is open to majors from different disciplines, such as HCI, IS, EC and CS, the instructor can leverage the different strengths these students bring to a project team by emphasizing collaborative teamwork (Chan and Wolfe, 2000). This requires a clear definition of roles and responsibilities necessary for a collaborative team. Students from different academic fields exhibit different approaches to problem solving. The instructor should also set clear criteria for team composition, especially in the cases of self-selected teams, to ensure diversity and complementary skill sets.

4.1.3. Placing HCI courses early in the curriculum sequence

For technically oriented programs, a separate HCI course should either be placed before or immediately after an analysis and design course or a Web site engineering course, like the *Design and Strategies for E-Commerce* course discussed earlier. Initially, our EC curriculum offered the HCI course late in the program, however, we now believe that a prerequisite HCI course provides a solid foundation for helping students to focus on needs of the individual users and the process for supporting user-centered design. While this would improve the learning outcomes of a Web site engineering course, some students may desire to have a broad business context first. When it is placed immediately after a development course, this course can cover HCI topics in greater depth, but students may be anxious to skip tasks in order to acquire

hands-on experience. Therefore, this course should not be placed too late in the curriculum track. Placing this course too far into the program of study diminishes its benefit for technically oriented students, because they may have already formed bad habits that will deter their understanding of HCI techniques. However, for management oriented programs, HCI topics can be incorporated into courses on system analysis, IT project management, or e-commerce technology review.

4.2. Learning outcomes

The study of usability testing and user-centered design emphasizes the iterative process. However, MIS/EC students tend to find tools and implementation more compelling than the process. This can often be seen in their tendency to bypass paper prototypes and go directly to producing a high-fidelity prototype. Since many point-and-click tools are available for producing templates and visual design, students may feel more gratified by computer-generated output. However, doing so may cause students to lose focus on how to utilize qualitative and quantitative evidence for building the next iteration of an interface. It is important for the instructor to keep this in mind while teaching HCI topics to MIS/EC students.

The Web is an excellent vehicle for studying and practicing user-centered design. It is usually easy to build testable prototypes quickly. Because Web sites share many properties with other interfaces, these lessons translate well into other areas of software development. We recommend three methods for achieving the desired learning outcomes.

4.2.1. Making learning outcomes tangible

An instructor should structure learning outcomes into tangible deliverables. The most natural deliverables are the artifacts that would arise from engaging in the user-centered development methodology. For example, an HCI course can include assignments on user and task analysis, where students report on their interviews of users. Another important deliverable is a usability report which summarizes the result of user testing of a Web site. Other deliverables arise from card sorting for content organization, visual design for navigation, and user prototyping.

4.2.2. Emphasizing process

For many technically oriented students, a course based on process instead of end results can be a new experience. To foster student learning of the process and concepts, short in-class exercises and homework questions should accompany course-long projects. In the case of conducting a user test, it is important to grade students on their preparedness and professionalism when they conduct the test, not in the number or quality of the findings they produce. To encourage students to prepare thoroughly for user testing, the instructor posts the detailed checklist that is used while observing the students conduct the test.

It is essential that students learn that users must be involved in Web development. It is not possible to design a site in a vacuum. For this reason, it is important not to grade a Web site as an isolated homework or project, but to determine its usability

through user testing. To provide the most valuable learning experience, it is best to grade students *not* on how well the users perform during user testing, but on what students choose to do with the results of the user testing.

4.2.3. Designing special assignments

An instructor should create assignments and exercises that link HCI principles to advanced application development. This will reinforce student knowledge and HCI skills. For example, an assignment in a security course can consider the issue of how best to let a users know that they are actually using a secure site, and how best to ease their concerns for using their credit cards safely. When developing a set of JavaScript menus, the instructor can ask the students to conduct a card sorting exercise to best determine the structure of the menu content. Another possibility is to conduct a task analysis and compare the results with click stream data while developing personalized Web pages.

4.3. Innovative pedagogy

An interactive and collaborative approach to teaching HCI facilitates student learning. This approach involves team projects and a studio environment that will be discussed in greater detail below. Team projects resemble the collaborative nature of usability testing and user-centered design. Here we focus on two specific innovative pedagogies.

4.3.1. Collaborating in teams

We have used a collaborative team learning approach for several terms to pair the *Design and Strategy for E-Commerce* class with an *HCI Usability* class, for HCI majors only (Chan and Wolfe, 2000). A pair of project teams, one from each class, was formed for collaboration. The teams from the e-commerce class played the role of Web developers. We synchronized the course schedules and assignments between these two classes to facilitate the collaborative process. Each of the first four deliverables listed in Table 4 served as input for HCI consultants to perform usability tests. Thus the consultants could perform user/task analysis, card sorting, interface design, and usability testing on both a paper prototype and the final Web site. Results of usability tests helped the developer teams to improve the results of their final project.

4.3.2. Working with real clients

Working with real clients is well worth the time and effort involved. Students gain valuable experience in learning how to translate user/task analysis into requirements, and to negotiate with clients about site design. In this way, they will focus on meeting potential users needs as opposed to the client's. For example, in one case students were challenged to redesign the Web site of a local chapter of a national non-profit organization. They had to achieve consistent branding and design while improving the site's usability within many constraints imposed by the organization's national site. Working with non-profit organizations presents different challenges from

working with commercial or corporate clients. Students will learn to focus on novice users as well as organizations with less technological savvy. Design and usability decisions have to sustain waves of progression as the organization and its staff will go through their learning curves.

4.4. Tools and technology

Courses as proposed in this paper should provide laboratory or studio opportunities for students to gain experience in putting theories into practice. Such effort will require special facilities, tools, and technology.

4.4.1. Studios and labs

Facilities play an important role in facilitating teamwork and emulating real-world design experience. A studio environment encourages students to work in small groups and critique one another's work. A studio suits graphic design. A laboratory with workstations arranged in clusters, with space for teamwork, is the most desirable layout. The lab naturally groups students into teams and is suitable for prototyping and user testing. With proper facilities, the learning of students can be effectively enhanced.

4.4.2. Software

The software packages supporting these courses should include Microsoft Visual Studio, Microsoft Office 2000 with FrontPage, Macromedia Dreamweaver, and Adobe PhotoShop. Dreamweaver is used primarily for prototyping and layout design. PhotoShop is used for graphic design. Fireworks and Flash are useful for advanced development. We discourage students from using these packages in introductory courses in order for them to first focus on the basic principles and tasks.

4.4.3. Implementation tools

With Web design, HTML plus a Microsoft-based scripting technology such as ASP or ASP.NET is a good choice for students practicing high-fidelity prototyping and usability testing. The scripting technology is easy to learn, effective, and also widely available. Web authoring tools that incorporate server-side scripting are also useful.

4.4.4. Special tools

We also use software, developed by Jorge Toro, a doctoral student, to automate the card sorting process (Toro and Wolfe, 2003). Using this software, students can create and display content items on-line instead of using 3×5-inch index cards. Representative users participating in card sorting can arrange content items into groups of related items and give each group a label. The software records the results for each user testing and analyses the results for patterns of clustering.

4.5. Advanced topics

Several advanced HCI topics, such as globalization, personalization, disabilities, and wireless strategies, are of increasing importance to Web development. Introductory courses do not usually cover these topics in depth. However, a strong grasp of the implications of these issues for usability will enable developers and designers to produce more effective Web solutions.

4.5.1. Globalization

Coordination between the internationalization and localization of content, design, and usability is essential for global site development, as seen in sites for IKEA (www.ikea.com) and United Air Lines (www.united.com). However, it is difficult to find projects or assignments for students to practice these development techniques. Approaching this topic works best when students have ready access to people who grew up in a culture other than the US. Having a class with students from different cultures is a rich opportunity for discussing this topic. As a simple in-class exercise, ask students to write down the cultural meaning for the colors red, white, green, yellow, and black. Once the students have completed the survey, they will tally the responses, and discuss the results. Another possible homework exercise is to ask students to investigate Unicode and localization options on their computers and to configure a browser to correctly display pages that are written in such languages as Arabic, Japanese, and Chinese.

4.5.2. Personalization

Beyond simple design using cookies and session variables, personalization will be an important mechanism to customize the content and services that will build relationships with customers, employees, and users of different preferences. Students can benefit from applying various personalization mechanisms to customize content and services. Any sort of personalization will be ineffective without in-depth knowledge of the people who use the Web site. This is an opportunity for conducting user and task analyses. Students will also learn to tie HCI issues to customer relationship management (Carey, 2002).

4.5.3. Disabilities

Universal access is often left toward the end of a Web development or a usability course. Input and output devices beyond the Web will become important considerations in the near future. The World Wide Web Consortium's Web site (www.w3.org) is a rich resource for information on how to make a Web site more accessible to people with special needs.

As an exercise, have students experiment with screen readers. Ask students to complete a few simple tasks on their Web sites with a screen reader turned on and the computer monitor turned off, and have them report on the results. Many people with severe motor disabilities browse the Internet by using adaptive pointing devices such as a head wand or a switch coupled with an onscreen keyboard. These people do not have access to a mouse. Ask students to complete several tasks on their Web sites

without using a mouse to verify that it is possible. Use contexts also become critical considerations for wireless interface design (Kim et al., 2002).

4.5.4. *Wireless*

Mobile Internet and wireless technology will significantly alter the way users interact with the information. Limitations in the current state of wireless technology pose many constraints for designing effective user interface for wireless applications. Small screen display, limited bandwidth, and the simplistic yet diverse functionality of wireless handheld devices affect usability (Chan and Fang, 2001, 2003). Usability guidelines for wireless devices have not yet been firmly established. For developing wireless applications, students need to focus on requirement gathering, user and task analysis in the mobile and stationary contexts, content presentation, search and navigation, and interface design to support multiple form factors and in a multi-channel environment (Chan et al., 2002).

5. Conclusions

The rise of the Web has necessitated the creation of systems that can be used by people with minimal specialized training. MIS/HCI researchers have called for a new taxonomy and research agenda that address the holistic experience of human interaction with technology (Zhang et al., 2002). MIS/HCI faculty has also recognized this need and attempts to integrate HCI topics into masters level MIS/EC curricula. Institutions can try various approaches to introduce HCI topics, either in a separate course, or in existing MIS or EC courses. However, there are several challenges facing instructors. These include identifying appropriate textbooks and acquiring adequate institutional resources to support laboratory and software needs. Therefore, the actual "how to" in integrating HCI topics in established MIS/EC programs is a timely topic for deliberation and exchange among MIS and HCI academicians. This "How to" in mapping pedagogies to learning outcomes and student skill sets does present future opportunities for innovation and research. There have been some encouraging developments at the organizational level. The formation of an HCI special interest group within the AIS organization is a positive step. A special issue on MIS-related HCI research helps to place teaching and student learning in the context of bridging theories with practice. These initiatives can form a nexus for supporting further research and exchange among MIS, EC, and HCI faculty, which is essential for this effort to succeed.

References

Agarwal, R., Karahanna, E., 2000. Time flies when you're having fun: cognitive absorption and beliefs about information technology usage. Management Information Systems Quarterly 4 (24), 665–694.

Beard, J., Peterson, T., 1988. A taxonomy for the study of human factors in management information systems (MIS). In: Carey, J. (Ed.), Human Factors in Management Information Systems. Ablex Publishing Corporation, Norwood, NJ, pp. 7–26.

Carey, J., 1988. Human Factors in Management Information Systems. Ablex Publishing Corporation, Norwood, NJ.

Carey. J., 2002. Human computer interaction research in the MIS discipline. Panel Presentation at AMCIS'02. Available at: http://melody.syr.edu/hci/amcis02_panel.cgi

Chan, S., Fang, X., 2001. Usability issues for mobile commerce. Proceedings of the Seventh Americas Conference on Information Systems, pp. 439–442.

Chan, S., Fang, X., 2003. Mobile commerce and usability. In: Siau, K., Lim, E. (Eds.), Advances in Mobile Commerce Technologies. Idea Group Publishing, Hershey, PA, pp. 235–257.

Chan, S., Wolfe, R., 1999. Collaborative team learning approach for Web development. Proceedings of the Fifth Americas Conference on Information Systems, pp. 181–183.

Chan, S., Wolfe, R., 2000. User-centered design and web site engineering: an innovative teaching approach. Journal of Informatics Education and Research 2 (2), 77–87.

Chan, S., Fang, X., Brzezinski, J., Zhou, Y., Xu, S., Lam, J., 2002. Usability for mobile commerce across multiple form factors. Journal of Electronic Commerce Research 3 (2), 187–199.

Cohen, E., 2000. Curriculum Model of the Information Resource Management Association and the Data Administration Managers Association. Available at: http://gise.org/IRMA-DAMA-2000.pdf

Davis, F., 1989. Perceived usefulness, perceived ease of use, and user acceptance of information technology. Management Information Systems Quarterly 3 (13), 319–340.

Dennis, A., Wixom, B., 2000. Systems Analysis and Design: An Applied Approach. Wiley, New York.

De Villiers, C., 2001. Teaching the evaluation of electronic commerce sites using HCI techniques. Proceedings of the 16th Annual Conference of the International Academy for Information Management, pp. 126–135.

Dishaw, M.T., Strong, D., 1998. Supporting software maintenance with software engineering tools: a computed task-technology fit analysis. Journal of Systems and Software 44 (2), 107–120.

Dishaw, M.T., Strong, D.M., 1999. Extending the technology acceptance model with task-technology fit constructs. Information and Management 36, 9–21.

Douglas, S., Leventhal, L., Tremaine, M., Wills, C., Bill, M., 2002. Incorporating human-computer interaction into the undergraduate computer science curriculum. SIGCSE Bulletin 34 (1), 211–212.

Faulkner, X., Culwin, F., 2000. Enter the usability engineer: integrating HCI and software engineering. Proceedings of ITICS, ACM, New York, pp. 61–64.

Galletta, D. (2002). HCI research in MIS: a personal view. Panel Presentation at AMCIS'02. Available at http://melody.syr.edu/hci/amcis02_panel.cgi

Gefen, D., Straub, D., 2000. The relative importance of perceived ease of use in IS adoption: a study of e-commerce adoption. Journal of the Association for Information Systems 1 (8). Available at: http://jais.aisnet.org/articles/default.asp?vol=1andart=8

Gomolski, G., Dreyfuss, C., Dallas, S., Feiman, J., Morello, D., Witty, R., Young, C., Mingay, S., Jones, N., Matlus, R.T., Berg, T., 2001. Managing the dynamic IT skills portfolio. Gartner Report.

Goodhue, D.L., Thompson, R., 1995. Task-technology fit and individual performance. Management Information Systems Quarterly 19 (2), 213–236.

Gorgone, J.T., Gray, P., 2000. MIS 2000: model curriculum and guidelines for graduate degree programs in information systems. Journal of Communications of the Association for Information Systems 3, article 1.

Gorgone, J., Kanabar, V., 2002. Masters in information systems: a Web-centric model curriculum. Proceedings of Information Science InSITE –"Where Parallel Intersects," pp. 553–563.

Hackos, J., Redish, J., 1998. User and Task Analysis for Interface Design. Wiley, New York.

Hansen, S., Deshpande, Y., Murugusan, S., 2001. A skills hierarchy for web-based systems development. In: Murugesan, S., Deshpande, Y. (Eds.), Web Engineering: Managing Diversity and Complexity of Web Applications. Springer, Berlin, Germany, pp. 223–235.

Hewett, T., Baecker, R., Card, S., Carey, T., Gasen, J., Mantei, M., Perlman, G., Strong, G., Verplank, W., 1992. Curricula for human–computer interaction. Association for Computing Machinery. Available at: http://www.acm.org/sigchi/cdg/

Hix, D., Hartson, H.R., 1993. Developing User Interfaces: Ensuring Usability through Product and Process. Wiley, New York.

Hoffer, J., Geroge, J., Valacich, J., 2002. Modern Systems Analysis and Design. Prentice-Hall, Upper Saddle River, NJ.

Kalin, S., 1999. Mazed and confused. CIO Web Business. Available at: http://www.cio.com/archive/webbusiness/040199_use_content.html

Kendall, K., Kendall, J., 2002. Systems Analysis and Design, 5th Edition. Prentice-Hall, Upper Saddle River, NJ.

Khoo, B., 1999. An interactive case scenario for teaching user interface design. Proceedings of the Fifth Americas Conference on Information Systems, pp. 918–920.

Kim, H., Kim, J., Lee, Y., Chae, M., Choi, Y., 2002. An empirical study of the use contexts and usability problems in mobile Internet. Proceedings of the 35th Hawaii International Conference on System Sciences, IEEE Computer Society Press, Los Alamitos, CA.

Lazar, J., 2000. Teaching web design through community service projects. Journal of Informatics Education and Research 2 (2), 71–76.

Lazar, J., Preece, J., Gasen, J., Winograd, T., 2002. New issues in teaching HCI: pinning a tail on a moving donkey. Proceedings of ACM SIGCHI 2002 Conference, pp. 696–697.

Lederer, A.L., Maupin, D.J., Sena, M.P., Zhuang, Y., 2000. The technology acceptance model and the world wide web. Decision Support Systems 29, 269–282.

Lidtke, D., Stokes, G., Haines, J., Mulder, M. (Eds.), 1999. ISCC '99: An Information Systems Centric Curriculum: Program Guides for Educating the Next Generation of Information Systems Specialists, In Collaboration with Industry. NSF DUE/CIS-9352944, DUE-9455450, DUE-9796243. Available at: http://www.iscc.unomaha.edu

Madsen, K., 1999. The diversity of usability practices. Communications of the ACM 42 (5), 61–62.

Maracas, G., 2001. Systems Analysis and Design: An Active Approach. Prentice-Hall, Upper Saddle River, NJ.

Mathieson, K., Keil, M., 1998. Beyond the interface: ease of use and task/technology fit. Information and Management 34 (4), 221–230.

Mayhew, D., 1999. The Usability Engineering Lifecycle: A Practitioner's Handbook for User Interface Design. Morgan Kaufmann, San Francisco.

Nielsen, J., 1993. Usability Engineering. AP Professional, San Diego, CA.

Nielsen, J., 2000. Designing Web Usability: The Practice of Simplicity. New Riders Publishing, Indianapolis, IN.

Nielsen, J., Sano, D., 1994. Design of SunWeb—Sun Microsystem's intranet. Available at: http://www.useit.com/papers/sunweb

Perlman, G., 1995. Teaching user interface development to software engineers. Proceedings of CHI '95 MOSAIC OF CREATIVITY, ACM, New York, pp. 375–376.

Powell, T., 1998. Web Site Engineering: Beyond Web Page Design. Prentice-Hall, Upper Saddle River, NJ.

Preece, J., Rogers, Y., Sharp, H., Benyon, D., Holland, S., Carey, T., 1994. Human–Computer Interaction. Addison-Wesley, Reading, MA.

Rettig, M., 1994. Prototyping for tiny fingers. Communications of the Association for Computing Machinery 37 (4), 21–27.

Rosenbaum, S., Wilson, C., Jokela, T., Rohn, J., Smith, T., Vredenburg, K., 2002. Usability in practice: user experience lifecycle—evolution and revolution. Proceedings of the ACM SIGCHI 2002 Conference, pp. 898–903.

Rosenfeld, L., Morville, P., 1998. Information Architecture for the World Wide Web. O'Reilly, Sebastopol, CA.

Rosson, M.B., Carroll, J., 2002. Usability Engineering: Scenario-Based Development of Human–Computer Interaction. Morgan Kaufmann, San Francisco, CA.

Satzinger, J., Jackson, R., Burd, S., 2002. Systems Analysis and Design in a Changing World, 2nd Edition. Course Technology, Boston, MA.

Scown, P., McManus, B., 1995. Educating computer people about people and computers. Computer Bulletin 7 (1), 18–19.

Shneiderman, B., 1997. Designing the User Interface: Strategies for Effective Human–Computer Interaction, 3rd Edition. Addison-Wesley, Reading, MA.

Strong, D., 2002. Human computer interaction research in the MIS discipline. Panel presentation at AMCIS'02. Available at: http://melody.syr.edu/hci/amcis02_panel.cgi

Tedeschi, B., 2002. How Fidelity gets that usability religion. Business 2.0. Available at: http://www.business2.com/articles/mag/print/0,1643,43537,00.html

Toro, J., Wolfe, R., 2003. CardZort: a tool to assist the organization of hierarchical taxonomies in user interfaces. Interaction, submitted.

Trevino, L., Webster, J., 1992. Flow in computer-mediated communication. Communication Research 5 (19), 539–573.

Venkatesh, V., 2000. Determinants of perceived ease of use: integrating control, intrinsic motivation, and emotion into the technology acceptance model. Information Systems Research 4 (11), 342–365.

Venkatesh, V., Davis, F., 2000. A theoretical extension of the technology acceptance model: four longitudinal field studies. Management Science 2 (46), 186–204.

Venkatesh, V., Davis, F., 1996. A model of the antecedents of perceived ease of use: development and test. Decision Sciences 3 (27), 451–482.

Vredenburg, K., 1999. Increasing ease of use. Communications of the ACM 15 (42), 67–71.

Vredenburg, K., Isensee, S., Righi, C., 2002. User-Centered Design: An Integrated Approach. Prentice-Hall, PTR, Upper Saddle River, NJ.

Weinberg, J., Stephen, M., 2002. Participatory design in a human–computer interaction course: Teaching ethnography methods to computer scientists. SIGCSE Bulletin 237–241.

Whitten, J., Bentley, L., Dittman, K., 2001. Systems Analysis and Design Methods, 5th Edition. McGraw-Hill Irwin, Boston, MA.

Zhang, P., Benbasat, I., Carey, J., Davis, F., Galletta, D., Strong, D., 2002. AMCIS 2002 panels and workshop 1: human–computer interaction research in the MIS discipline. Communications of the Association for Information Systems 9, 334–355.