Heather Richter
Teaching Statement

I am interested in an academic career because of the opportunity it affords me to work and interact with students on a variety of levels. My earliest choice of “what I want to be when I grow up” was a teacher. I decided to pursue a Ph.D. so that I could combine my interest in technology and my interest in teaching into one career.

I began interacting with students at Michigan State University, as a teaching assistant for an undergraduate data structures course, where I assisted students with laboratory and homework assignments. At Georgia Tech, I have been a teaching assistant for an undergraduate Software Engineering course and an introduction to Human-Computer Interaction (HCI) course. I recently taught that same HCI course, where I was responsible for all aspects of the course, including lecturing on basic HCI concepts, creating and grading tests, and guiding students through projects. Through these experiences, I learned that students’ knowledge of the skills of how to design, implement, and test any program that they work on is vitally important. Yet those skills are difficult to appreciate and learn until they are needed. Thus, the scope of work in project-based courses is important in order to motivate and teach many Software Engineering and HCI concepts.

Teaching the HCI course helped me realize the challenge in how to construct a valuable and educational project, where students are able to realistically try, and even fail at, a number of techniques. In this course, students were given a semester long group project where they investigated a problem of their choosing, designed and implemented interactive prototypes to address that problem, and evaluated their designs. I enjoyed the creativity and novelty of each of these projects, with subjects such as a mall kiosk, a home inventory system, and public displays for art. But that same creativity meant that students could potentially use different techniques and have different experiences. Thus, such an introductory course needs to balance between standardized assignments to build HCI design and evaluations skills and project-based assignments to put some of those skills to real use.

Software Engineering presents an even larger challenge, as one upper level project course is not sufficient. Students often spend several years implementing programs with poor habits that suffice until they are finally taught processes for design, implementation and testing in a Software Engineering course. Individual software engineering skills need to be instilled even in early programming courses, so that later courses can build upon those skills to introduce group processes and techniques. As part of a graduate seminar, I helped to develop a Software Engineering degree curriculum that focused around a studio model, where in the final two years of their degree, students would participate in several group oriented projects, performing different tasks at different stages of the process in each. This degree would more closely resemble the type of “portfolio” that architects create. This type of curriculum allows students to work on truly large-scale long-term projects at different stages and apply a larger number of techniques. This model may only be practical as part of a Software Engineering degree, but aspects of this studio could be implemented for other programs using one or more upper-level Software Engineering courses.

As my previous experiences show, I am most interested in teaching courses in my research areas of Human-Computer Interaction and Software Engineering, and will emphasize project-based components in both areas. I am also interested in teaching other HCI-related topics, such as user interface software and technology, Computer Supported Cooperative Work, and Ubiquitous Computing, and leading seminars in exploration of cutting edge research in HCI and Software Engineering.

Education does not just revolve around individual courses, but involves the entire educational experience of both undergraduates and graduate students. Research can play an important role, and I have supervised a number of undergraduates in research projects. Several have assisted on prototypes for my thesis research. Hussein Sharifi and Belinda Nambooze added features to TeamSpace, while Jesslyn Beattie implemented an early TAGGER prototype. Others have explored a new research project, implementing and evaluating a Digital Picture Frame on a PDA (John Ndukuba) and other devices (Jesslyn Beattie). Research is also valuable for lower level undergraduates with little programming experience. I have been involved with a program at Georgia Tech, the Intel Opportunity Scholars, that pairs freshman and sophomore women and minorities with graduate students who supervise small research projects and serve as general mentors. This program gives younger students an opportunity to experience the breadth of computing, opens their eyes to the possibilities of graduate school and research, and gives them a creative outlet to succeed in computer science. This year, my scholars will be conducting a field study of several Digital Picture Frame prototypes. At the end of the year, the students will present their research at a poster session within the department. This kind of special program is important to attract, retain, and provide sufficient support for success for undergraduates, and for women and minorities in particular. I will continue to foster the educational experience of students through such programs and other community building activities.