Editor: Scott F. Midkiff ■ Virginia Institute of Technology ■ midkiff@vt.edu

Rapid Prototyping of Computing Systems

Scott F. Midkiff, Virginia Institute of Technology

EDITOR'S INTRO

The future scientists and engineers who will conduct research and develop pervasive computing products and systems require relevant educational experiences. Current practitioners also need training to expand their capabilities and effectiveness. In each issue of *IEEE Pervasive Computing*, this column will provide an in-depth description of important and innovative educational programs to meet these needs. It will describe relevant academic courses, student design projects, degree and certificate programs, and training courses. The goal is to disseminate innovative ideas and success stories with the hope of expanding and improving education and training in pervasive computing for students and practitioners. I welcome your suggestions and comments for future columns. —*Scott Midkiff*

As reflected elsewhere in this inaugural issue of *IEEE Pervasive Computing*, pervasive, or ubiquitous, computing requires the integration of multiple technologies, including software, hardware, and human-computer interaction (HCI). To prepare students for this new paradigm in computing, we need multidisciplinary academic programs and courses. Furthermore, real-world design projects, design processes, and team experiences must play a primary role.

"Rapid Prototyping of Computing Systems" at Carnegie Mellon University, organized by Dan Siewiorek, a professor in the School of Computer Science and the Electrical and Computer Engineering Department, combines all these elements in a single innovative course offered in multiple departments at CMU (see the "Quick Facts" sidebar). Students learn topics in multiple disciplines and complete an industry-driven, team-based project using a well-defined design process. Although the course prepares students for a wide range of computing applications, the topics and projects focus on pervasive computing.

A UNIQUE OPPORTUNITY

The course offers benefits to industry partners, faculty, and students. Industry partners appreciate the innovative solutions that students generate. Unlike corporate engineers and scientists, students are unencumbered by preconceived notions of how to solve a problem. Industry partners are also interested in learning the process—that is, how to quickly produce prototype systems and services that require multidisciplinary design.

Each year has a different application focus, so the course never gets stale for faculty because it can take research ideas further into the prototype stage than a research group typically can. The class also helps faculty identify relevant research issues and thus can lead to thesis topics for masters and PhD students. The instructors' time commitment is not overly burdensome. After the first couple of weeks, instructors act primarily as consultants. They attend design meetings, help students plan, hold dry-runs of demonstrations and presentations, meet with each student to provide a performance review after each of the early project phases, and review documents and presentations.

For students, this truly unique course exposes them to multiple disciplines and lets them take a project from concept to essentially a product in just one semester. The class's start-up atmosphere—complete with a celebration and tee shirts for

QUICK FACTS

Course: "Rapid Prototyping of Computing Systems"

Units: School of Computer Science, Department of Electrical and Computer Engineering, School of Design, Human-Computer Interaction Institute, and Robotics Institute

Institution: Carnegie Mellon University

Lead Professor: Dan Siewiorek

Level: Upper-division undergraduate

URL: www.cs.cmu.edu/~wearable

1536-1268/02/$17.00 © 2002 IEEE

HANDY ANDY: PERVASIVE COMPUTING FOR EDUCATION

The Andrew system, a well-known distributed file system developed and first deployed at Carnegie Mellon, provides a unified, distributed computing and storage environment for education and research. The Wireless Andrew project followed, which led to IEEE 802.11b wireless local area network coverage of the CMU campus. Students in the "Rapid Prototyping of Computing Systems" course are both early developers and early users of Handy Andy, a new model for educational computing that some see as the next evolution to support university students and researchers. Particularly, Handy Andy uses the Wireless Andrew infrastructure and handheld computing to support the growing use of teams in education and research.

The Spring 2000 class, partnered with IBM's pervasive computing research group, identified and developed two prototype services to support team-based design projects. The first service, Portable Help Desk or PhD, lets a user determine location and other information about other users and services, such as printers. The second service, Idealink, is a shared meeting space and white board. As Figure A shows, Idealink lets teammates collaborate with additional features, including image storage, selective drawing playback for those not at the meeting, and real-time remote collaboration. The system can handle real-time, graphical updates for multiple users in disparate locations.

The Handy Andy prototype was developed for Hewlett-Packard Jornada handheld computers that use Lucent WaveLAN network interface cards to communicate via Wireless Andrew. Students in the Spring 2001 offering were loaned Jornadas for use during the semester to evaluate the system's effectiveness. They determined that the Handy Andy services and other applications were too limited and, hence, not as useful as planned. However, Idealink let multiple student designers work in parallel, and generate and share ideas more quickly than using a traditional whiteboard controlled by a single user at any given time. Based upon this user feedback, research students continue to develop Handy Andy and have added additional features, such as the ability to listen to audio clips. The Spring 2002 class is using the improved system.

Figure A. Screen shot from Idealink illustrating a design session.

the "first customer ship"—excites them.

As a practical matter, many students, like their potential employers, appreciate this as valuable preparation for work in today's corporate environment. The course looks good on a student's resume, and it provides a wonderful discussion topic for job interviews. Susan Ambrose, director of CMU's Teaching Center, held a focus group with students to evaluate the class in 1995. She reported that "overall, student reaction to the project/course was very positive; they all agreed that this was either the best or one of the best learning experiences they've had at Carnegie Mellon." She also stated that "I don't believe that I have ever interviewed a group of students who were more excited about a course than this one."

HISTORY AND BACKGROUND

The class grew out of a noncredit course on design offered to mid-level managers from industry in the summer of 1991. Instead of just giving lectures—a rather dry approach—instructors decided to have the participants do actual hands-on design in the afternoons. At the end of the course, each participant left with a prototype wearable computer. Everyone involved considered the experiment a great success.

During the following year, CMU offered similar design opportunities to undergraduate students through independent studies and undergraduate research projects. A formal course was first offered in the spring of 1993. CMU has offered the class each spring since then, including this semester, Spring 2002. The pedagogical approach and course management have evolved with each offering. In fact, the design artifacts' evolution closely matches the evolution of wearable computing and pervasive computing research at CMU.

During Spring 2001, the 30 students enrolled in the course designed a driver assistance system—a Contextual Car-Driver Interface—working with General Motors. In past semesters, students explored other application domains, including wearable computing for manufacturing at Boeing, assistance for bridge inspectors with the Pennsylvania Department of Transportation, pervasive computing applications for education for IBM (see the related sidebar), tools for off-shore crane operators for Chevron, and the Virtual Voyager system to let students virtually participate in water-quality measurements and other science experiments.

Although the design process's artifact changes yearly, the fundamental learning objectives do not. Students learn to

- generate system specifications
- partition functionality between hardware and software
- produce specifications for interfaces between subsystems
- use computer-aided design tools
- fabricate, integrate, and debug a system
- evaluate the system with respect to an end-user application

Also, the focus on ubiquitous computing has remained consistent over the years. Wearable computing provides the right scale of system for developing a prototype in about four months.

COURSE OPERATION

Course preparation takes place throughout the year. Each summer, work begins on the next spring's offering. Siewiorek identifies an industry partner and works with the partner to understand the parameters of a good class design project and the type of support that the course needs. Faculty and researchers conduct exploratory design exercises and propose a high-level system architecture to address the industry problem. This work serves as the class project's foundation.

The industry partner provides an application domain, expert staff to give lectures and serve as reviewers, training material, and financial support. The course evolves around a user-centered design methodology. Thus, the design team's most important members are the industrial partner's practicing field workers. These workers introduce the problem and interact with the students throughout the semester, including critiquing their designs. Whenever possible, the students visit an actual work site for first-hand observations.

Siewiorek also assembles a multidisciplinary team of course instructors. Built over the years, this includes faculty and researchers from CMU's School of Computer Science, Department of Electrical and Computer Engineering, School of Design, Human-Computer Interaction Institute, and Robotics Institute. The instructors act primarily as expert consultants, although they provide introductory lectures early in the semester.

The multidisciplinary design evolves in parallel, ensuring a strong cross-fertilization among the different teams. They collaborate through a framework for compatibility between interdisciplinary design tools and agreement on a design language and representations, so the latest information from each discipline is available and understandable to the others. To provide flexibility in the decision process during the initial design stages, each discipline formulates the problem from its own viewpoint in terms of design goals.

"I don't believe that I have ever interviewed a group of students who were more excited about a course than this one."

The course comprises three phases that form the project's design process: conceptualization, detailed design, and implementation.

Conceptualization

During the conceptualization phase, the class defines the problem and surveys the technology they can apply. Brainstorming and other methods help develop a design definition that specifies functionality, cost, performance, and techniques for prototype construction.

Students form discipline-specific teams, comprising four or five students, for this first phase. For example, electrical and computer engineering students might investigate hardware components while HCI students might develop use scenarios that help define the problem. In the final portion of this phase, students specify the system architecture and subsystems. The system architecture must address all aspects related to the defined problem including computational requirements, sensing, and human-computer interfaces. The system plan must also consider available technologies, people, and resources. Additionally, the students partition system functionality and assign them for realization in software or hardware components. They also refine performance, interfaces specifications, and evaluation criteria.

Detailed design

The second phase leads to a detailed design document for each subsystem. Students form multidisciplinary teams for this and later phases of the course, taking responsibility for system parts and subsystems related to their specialties. Instructors provide risk management to prevent decisions that will lead to later difficulties. Students use appropriate computer-aided design tools for each design task. They make component mock-ups to conduct HCI studies and use the outcomes to refine designs. In addition to this phase's written report, the oral presentation plays an important role because it offers instructors and industry partners an opportunity for review and suggestions.

Implementation

The third and final phase comprises three main activities: subsystems implementation using rapid-prototyping techniques, subsystems integration to create a system prototype, and quantitative and qualitative evaluation of the course methodology. System integration—one of the course's most important aspects—and test plans are developed in the detailed design phase based on the system architecture. The students test individual subsystems and then integrate them to form the complete functional system. Then, they evaluate all system aspects through user experiments.

The course concludes with final writ-

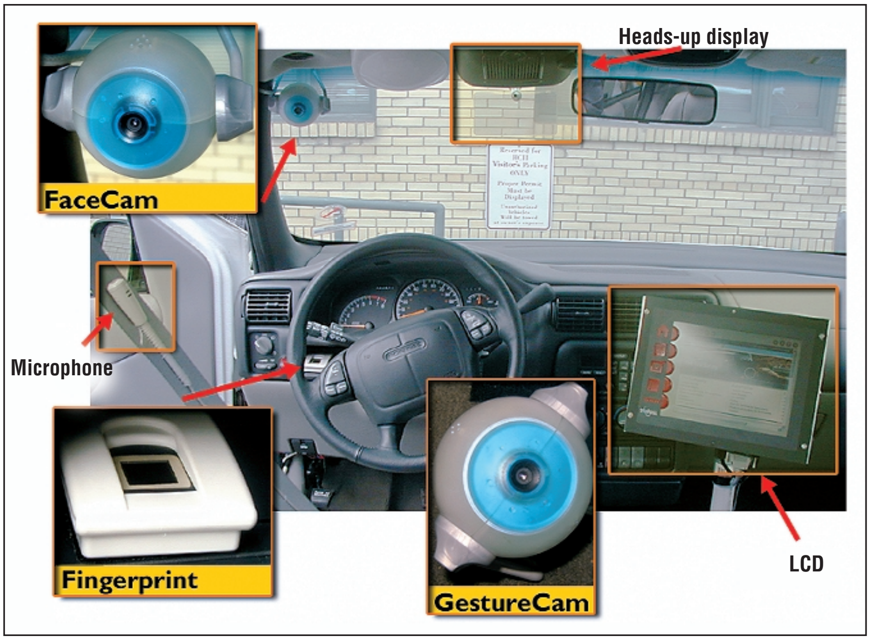

Figure 1. The GM Companion driver interface system.

ten reports, Web pages, and oral team presentations. The instructors and industry partners critique the presentations. The final report and prototype system form a comprehensive deliverable for the industry partner, and all students in the class contribute to it. An editorial board of typically three students compiles and edits the final report.

SPRING 2001 DESIGN PROJECT

During the Spring 2001 offering, students explored integrating computing applications into a vehicle through the Contextual Car-Driver Interface. The new system, the GM Companion, is intended to make driving a "more useful, more entertaining, and safer experience." The concept is to link information from multiple sources, including the driver's personal digital assistant, the current time and location, the Internet, and entertainment content. Figure 1 shows some of the system's hardware components.

The project investigated potential features and analyzed driver safety issues related to the human-computer interface. The project also led to the design of individual subsystems and to specifications of how the subsystems would operate together. Finally, the subsystems were implemented and integrated to form a working prototype of many of the features envisioned in the initial study. General Motors, the Spring 2001 industry partner, was impressed by how quickly the students pulled ideas together.

The students developed proactive agents that worked on the driver's behalf. Two of the agents prototyped for the GM Companion were Personal Assistant and Preferences. The Personal Assistant agent interfaces with the driver's PDA and reminds the driver of tasks at the appropriate time and place. For example, the system could remind the driver to pick up bagels for a morning meeting as the vehicle nears a bagel shop during the drive to work. The Preferences agent configures the car based on the driver's personal preferences. For example, it can adjust the seats, steering wheel, and mirrors to a driver's preferred locations.

LESSONS LEARNED

The class demonstrated that upper-division undergraduate students can achieve results-oriented, state-of-the-art design and build projects. Siewiorek has found that 30 is approximately the optimal number of students to take a reasonable project to the prototype stage in one semester. A larger class would require a larger project, making system integration overly difficult.

The course's only form of advertising is word of mouth. Relying on word of mouth tends to attract motivated students, who understand that this is not a typical class. However, not all students can adjust to the nontraditional format and the need to traverse discipline boundaries. Approximately 10 percent of the students drop the course during the first week, once they more fully understand it. Electrical and computer engineering students and computer science students expect to receive structured specifications for assignments, even for supposedly open-ended design projects. Students from design and HCI often cope more easily with the unstructured initial specifications. Those students that do complete the course receive an excellent introduction to pervasive computing and experience the design process from concept to prototype.

ACKNOWLEDGMENTS

I based this article on the course syllabus and other information provided by Dan Siewiorek and Asim Smailagic, personal communication with Siewiorek, and the final course project report written by the Spring 2001 students.

Scott F. Midkiff is a professor in the Bradley Department of Electrical and Computer Engineering at Virginia Tech. His research interests are in computer networks, including wireless and mobile, management, applications, and simulation. Contact him at midkiff@vt.edu.