Pervasive Computing Education

Why, What, and How?

Audrey Girouard
Carleton University

Andrew L. Kun
University of New Hampshire

Anne Roudaut
University of Bristol

Orit Shaer
Wellesley College

Editors:
Andrew L. Kun,
University of New Hampshire
andrew.kun@unh.edu

In this column, we ask three central questions related to ubiquitous computing education. First, why is specialized ubicomp training needed? Next, what should the goal of such specialized training be? And finally, how should these goals be accomplished pedagogically? We argue that these questions should be answered by a community that supports new forms of teaching, training, and learning in ubiquitous computing.

Ubiquitous computing, or ubicomp, is defined by embedded, networked devices that become a part of the fabric of our lives. It is complicated to create ubicomp systems, because of the requirements that these systems support multiple scales, be reliable, be power-efficient, provide data security, and facilitate user interactions. All of these requirements need to be met simultaneously.

As a result of these challenges, ubicomp systems are often built by multidisciplinary teams. Teams might include members with expertise in electrical and computer engineering, computer science, psychology, anthropology, design, and other disciplines. Within ubicomp projects, all of these team members have to find a way to collaborate effectively. At the same time, the problems and tasks that team members face are often quite complex. After all, it is complicated to create user interfaces that are intuitive, just like it is complicated to create power-efficient circuits.

Due to the novel aspect of the technologies involved and the multidisciplinary skills needed to design such systems, teaching and training new innovators in this field is difficult. Traditional teaching programs are often not suited to cover such a wide range of applied skills.

The time has come to rethink the way ubicomp is taught.

Published by the IEEE Computer Society
1536-1268/18/$33.00 ©2018 IEEE

CHALLENGES OF TEACHING A MULTIDISCIPLINARY FIELD

The multidisciplinary nature of ubiquitous computing allows us to address issues that a single discipline cannot. It also helps each discipline to improve through cross-fertilization of ideas between disciplines, and it creates an intellectually challenging and satisfying environment for researchers and practitioners. However, multidisciplinarity also poses many challenges when it comes to teaching ubicomp.

It is unclear what constitutes the core of the discipline. Ubiquitous computing requires a blend of technical and social sciences, but ubiquitous computing instructors (both in academia and in the industry) come in many flavors. Thus, while the content of teaching units might be regulated by an institution or degree program, instructors will often emphasize their own interests, or on their own skills. For example, some instructors will devote more time to teaching the security aspects of ubicomp systems, while others will focus on human–computer interaction aspects. This creates a disparity between what is taught at different institutions.

It is unclear what depth of teaching is required. Because ubiquitous computing is based on multiple fields, there are many concepts and skills to learn, and it is difficult to know what material to include in a core curriculum that is required for ubicomp research and development. For example, one can choose to teach how to design a controlled experiment and how to analyze the results using statistical tools. Yet, to accomplish this, it is unclear if students need to learn about the entire field of experimental psychology. It is also not clear what level of statistics knowledge will suffice. A poor understanding of statistics can lead to the incorrect use of software tools, but surely there is no need to be a skilled statistician to understand the basis of parametric and nonparametric testing.

It is hard to teach students from diverse backgrounds. It is always easy for instructors to fall prey to the "curse of knowledge"—a tendency to assume that their audience knows more than they actually do.[1] This is especially true for interdisciplinary courses, where students might come from different backgrounds. For example, if we need to present material on digital signal processing, students with a computer engineering background might need less of an introduction than those with a computer science background, and computer science students are likely to need less introductory instruction than designers. It is not always clear how an instructor should present material such that students who have more preparation to understand it are still engaged, and those who need more support are not lost.

How can we address student diversity? The pool of potential ubicomp students is diverse in gender, culture, race, socioeconomic background, and abilities. This diversity represents the people who will use ubicomp in their everyday lives, and it needs to be reflected in the teams that create successful ubicomp products. However, due to the multidisciplinary nature of ubicomp, the field is also diverse in terms of professional culture—students from different professional backgrounds will value different aspects of work in the ubicomp domain. This means that they will also value different aspects of ubicomp education and training. Furthermore, the students and instructors within individual disciplines that constitute the field of ubicomp do not always reflect the diversity of society at large. We must consider how our teaching, in content and pedagogy, can foster diversity and inclusion.

CHALLENGES OF TEACHING AN APPLIED FIELD

Although the field of ubiquitous computing has many theoretical underpinnings, it is essentially an applied and practical field, with the goal of producing systems and devices that address human needs. Here, we see three potential issues.

Certain technical skills require considerable practice. While many ubiquitous computing systems are smartphone-based, researchers are also building new interactive systems that take on other form factors that involve physical prototyping. Learning such physical prototyping skills requires practice. It is one thing to know the principles of sketching, the basics of electronics, or how a fabrication machine works (e.g., laser cutter, 3-D printer), but using them requires a set of applied skills that come with practice.

Certain skills require access to specialized equipment. While many ubicomp (sub-) systems can be implemented on smartphones, or with crude prototypes, others require access to equipment such as a network infrastructure to run complex applications, or prototyping machines (e.g., CNC machines) to create physical objects. Importantly, the access that is required is physical access: it is not enough for students to design something, and then instruct someone else to implement the design remotely. Rather, the implementation requires skills, and we have to train students to acquire these skills.

Certain design skills require dedicated practice. One typical challenge that many of us have encountered is the difficulty of teaching students the principle of user-centered design (UCD). UCD requires the researcher to include the user in their design at every step, not only at the very end. However, it is natural as a human to believe that other people behave like we do, a cognitive bias known as the false-consensus effect, which means that students often do not understand this theoretical principle until they have experienced it themselves. As a result, students overlook activities that require understanding users' needs, such as review of critical incidents or contextual interviews. In contrast, those students who experience design for specific users will be better able to internalize the need for UCD. However, creating a curriculum that allows students to practice design skills with users can be expensive and time-consuming.

IMPROVING PEDAGOGICAL APPROACHES FOR UBIQUITOUS COMPUTING

Ubicomp instructors have access to a growing number of textbooks and general books about the field,[3] which sometimes include complementary lecture notes and exercises to engage students in the classroom. In addition, many instructors have developed clever active learning exercises to complement the material and encourage students to participate during class time. Furthermore, examples of syllabi can be useful[2]; however, they lack delivery details. These facts, and the challenges discussed above, motivate our argument that we must improve pedagogical approaches to better support instructors in the practical design and execution of their ubicomp teaching (whether a workshop, a course, or an online seminar). This is important both for established educators and for those who are transitioning from graduate school to teaching. Note that, while researchers have proposed the conceptual framework of an HCI living curriculum in 2014[4] and again in 2018,[5] no such community exists or has been proposed to address curriculum and pedagogical approaches designed to engage diverse students specifically with ubiquitous computing material.

Consequently, it is important to ask several questions about the training and education needed to help students become valuable members and leaders of ubicomp teams. A total of three central questions about ubiquitous computing education emerge—*why*, *what,* and *how*:

1. ***Why is training in ubicomp needed?*** Is it enough to train experts in narrow domains (e.g., experts who can create low-power embedded circuits, or usable applications), and then bring them together in teams that will tackle ubicomp problems? Or, do we need specialized training that targets ubicomp in addition to domain expertise? There is broad consensus that we do need specialized training, but often this argument is based on intuition and anecdotal evidence. We approach this question by first asking what the grand challenges are that we expect our students to tackle in the world (e.g., privacy and sustainability) by inventing and developing ubicomp solutions. Next, we ask who can better address the challenges: teams of domain experts, or teams where at least some team members have specialized ubicomp education. Answers to these questions will clearly identify problems that might exist with current ubicomp educational approaches.
2. ***What should constitute training in ubicomp?*** Once we identify the grand challenges, we need to ask further questions. What are the values, knowledge, and skills we should train students in ubicomp? What are the topics that should be covered? How do these depend on the background of students or their degree program? Answers to these types of questions will allow us to set goals for ubicomp education.

3. ***How should we teach and engage a diverse body of students?*** Once we identify specific goals for ubicomp education, we need to ask ourselves how those goals can be achieved. How does the unique nature of ubicomp challenge current pedagogical approaches? How can we create new pedagogical approaches for teaching and training in ubiquitous computing? Answers to these types of questions will help the community create the appropriate tools to reach our ubicomp education goals.

CONCLUSION

The three questions mentioned above are complex and require broad expertise both in the ubicomp community and in the education community. Answers will require input from a diverse set of stakeholders in academia, industry, and government. Thus, we propose that there is a need to create a community to support new forms of teaching, training, and learning in ubiquitous computing. This community can be kickstarted by one or more workshops or meetings that would include different stakeholders in the ubicomp field. The three questions mentioned above can help structure these workshops and meetings, and ultimately the creation of new ubicomp educational approaches. We also identify the need for an online repository of living documents including challenges for ubicomp to tackle in the world, learning goals for ubicomp education in various settings, ubicomp programs, and educational activities and materials for ubicomp education. This repository would be the catalyst for long-term, sustained conversations and innovations for ubicomp education.

REFERENCES

1. S. Pinker, *The Sense of Style: The Thinking Person's Guide to Writing in the 21st Century.* Baltimore, MD, USA: Penguin, 2015
2. SIGCHI Syllabi. [Online]. Available: https://sigchi.org/2017/01/syllabi/, Accessed on: Oct. 10, 2018.
3. A. L. Kun, "Textbooks for pervasive computing training and education," *IEEE Pervasive Comput.*, vol. 17, no. 40, pp. 59–61, 2017.
4. E. Churchill, J. Preece, and A. Bowser, "Developing a living HCI curriculum to support a global community," in *Proc. ACM Conf. Hum. Factors Comput. Syst. Extended Abstr.*, 2014, pp. 135–138.
5. O. St-Cyr, C. MacDonald, E. Churchill, J. Preece, and A. Bowser, "Developing a community of practice to support global HCI education," in *Proc. ACM Conf. Hum. Factors Comput. Syst. Extended Abstr.*, 2018, Paper W25.

ABOUT THE AUTHORS

Audrey Girouard is an Associate Professor with the School of Information Technology, Carleton University. Contact her at audrey.girouard@carleton.ca.

Andrew L. Kun is an Associate Professor of electrical and computer engineering with the University of New Hampshire. Contact him at andrew.kun@unh.edu.

Anne Roudaut is a Senior Lecturer with the Department of Computer Science, University of Bristol. Contact her at anne.roudaut@bristol.ac.uk.

Orit Shaer is the Class of 1966 Associate Professor of computer science and the Co-Director of the Media Arts and Sciences Program, Wellesley College. Contact her at oshaer@wellesley.edu.