Teaching Accessibility in a Technology Design Course

Kristen Shinohara, Cynthia L. Bennett, Jacob O. Wobbrock, Wanda Pratt, University of Washington
Email: kshino@uw.edu, bennec3@uw.edu, wobbrock@uw.edu, wpratt@uw.edu

Abstract: The goal of college computer science and informatics design curricula is to prepare the next generation of technology designers. While accessibility is considered important for technology users with disabilities, it is treated as a niche or stigmatized subdomain of technology design. As a result, traditional computer science and informatics curricula do not expose students to the needs of diverse users, and students transition to the workforce unaware of the importance and impact of accessible design. We incorporated accessibility as part of a design thinking course and observed student learning. We sought to challenge notions of accessibility as a niche area, learn methods and barriers for including accessibility in the curricula, and identify impacts on student learning. We found that students grasped general design concepts while designing for users with and without disabilities, and they faced challenges specific to designing for both populations in concert. Contributions include insights for future courses in accessibility and design and challenges facilitating accessible user-centered design process.

Keywords: Design Thinking, Accessible Design, Teaching Accessibility.

Introduction

People with disabilities use accessible technologies to engage with others in society (Cook & Hussey, 2002; Scherer, 1993), yet most personal technologies are not designed to be usable by people with impairments (Scherer, 1993), indicating that accessibility is not included as a main function of technical development. Although the goal of college level computer science and informatics education is to prepare students for careers as technology innovators, typical curricula covers accessibility under legal and professional issues (Section 508[1], ADA[2]), and accessible pedagogy (teaching students with disabilities) (Rosmaita, 2006). Therefore, despite research emphasizing the need to create accessible technologies (Danford, 2003; Stephanidis, Akoumianakis, Sfyrakis, & Paramythis, 1998), computer science and informatics students are not exposed to the needs of diverse users as more than an "edge-case" (Rosmaita, 2006) in learning design and development.

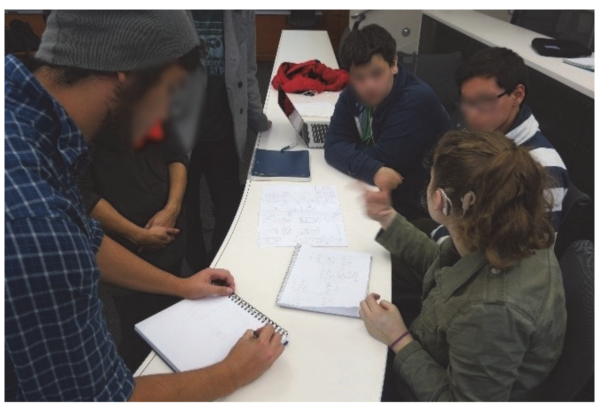

Figure 1. Students in our course getting feedback on their design ideas from a user with a hearing impairment.

Including accessibility as a key topic early in HCI and computer science curricula benefits technology design, society at large, and technological pedagogy; it is instrumental to ensuring both that technology is usable for people with disabilities and that technology instruction is accessible for students with impairments (Rosmaita, 2006). Including accessibility in computer science curricula positively impacted student learning (Poor et al., 2012; Rosmaita, 2006) and resulted in accessible technology designs (Bigelow, 2012; Ludi, 2007; Waller, Hanson, & Sloan, 2009). Yet, few computer science courses include accessibility as a *main* theme, and those that do remain disability-specific (Bigelow, 2012; Ludi, 2007; Poor et al., 2012; Waller et al., 2009), perpetuating the separation of users with disabilities from those without. Not including accessibility in traditional computer science and informatics education risks missing an important element of diversity, not just in technical domains, but in technology education overall. The notion of "teaching accessibility" remains a niche concept, something relegated to the few who wish to focus their efforts on this "edge case" (Rosmaita, 2006). The consequences are that technologies, like laptops, mobile phones, etc., continue to be unusable by those with disabilities.

The present technical landscape includes virtual and augmented reality (Microsoft's HoloLens, Oculus Rift), intelligent speech recognition (Apple's Siri, Amazon's Alexa, Microsoft's Cortana, Google Now), with self-driving cars on the horizon (Tesla, Google), all with potential to benefit both disabled and nondisabled users. These advances are opportunities for accessible design, it behooves technology designers and developers to engage in accessibility-focused practices.

To investigate how we, as technology educators, can effectively infuse accessibility as a main theme in technology design, we conducted a design course study assessing how students incorporate accessibility into a general introductory design thinking course (Figure 1). Our study investigated the tensions between concepts in accessibility and general design and we found that including accessibility as a core thread—as part of the practice of learning design (Wenger & Lave, 1991)—did not introduce barriers to learning design thinking overall. In fact, it broadened student thinking about diverse approaches to design. We also identified specific challenges faced by students and instructors in ensuring the feasibility and success of the course, with respect to both pedagogical and aspirational goals. Our contributions include a course outline for teaching accessibility as part of the main design theme, and insights on how to incorporate teaching accessibility in technology design courses. To allay instructors' fears that including accessibility in general design thinking courses is challenging enough to preclude teaching accessibility as well, we designed our study to identify *how* we can effectively incorporate teaching technical accessibility. Specifically, we considered: do students learn enough about design and accessibility to be able to create a prototype evaluated as usable by users with and without disabilities? How does including accessibility impact student learning about design thinking? We present our results and experiences in this report.

Background and Related Work

Teaching undergraduates accessibility is one way to increase accessible technologies in the future. In teaching design thinking, it is common to acculturate students to industry practices and strategies through hands-on learning objectives and projects. Engaging legitimate peripheral participation (Wenger & Lave, 1991) and communities of practice (Wenger, 1998) are key, and manifest in exposing students to authentic disabled experiences while also addressing tangible design problems, respectively. We briefly discuss teaching accessibility and design thinking, the two key elements of the course that served as objects of study.

Teaching Accessibility

Researchers have looked at how accessibility is addressed in teaching computer science, engineering, and design students. Rosmaita incorporated accessibility as a main theme in introductory web design courses (Rosmaita, 2006). Bigelow included Universal Design principles in introductory engineering courses, and found that students were inclined to emphasize accessibility in their engineering design projects (Bigelow, 2012). Both Rosmaita and Bigelow's courses did not involve working with people with disabilities, which we have done in our study. Waller et al., investigated a multi-year program on accessibility, with the focus on integrating accessibility throughout the learning experiences (Waller et al., 2009). Waller et al.'s multi-year approach frames our efforts here at a high level; we remain focused on specific elements and challenges in a single course. Ludi included stakeholders in a requirements engineering course to increase accessibility awareness, where projects were focused on requirements gathering in the engineering tradition, but designs were not produced (Ludi, 2007).

Despite research showing the benefits of a range of teaching accessibility practices (Putnam, Dahman, Rose, Cheng, & Bradford, 2015), students rarely consider disability and accessibility without provocation, and are not taught to include accessibility as part of the "main event" of technology design. For example, we conducted a brief overview of technology design courses representing three different undergraduate technical computing programs from two separate institutions. We assessed individual project information as posted on public websites and evaluated project descriptions to determine target audience and accessibility in design. We found twelve out of 179 (6.7%) student projects targeted users with disabilities, or might be considered accessible (some projects targeted caregivers, not people with disabilities, for example). Project information indicated that students were given high level prompts, and student groups self-defined design problems and selected target users for their projects. We elevate the concern that most computer science and informatics students do not consider the role of disability in technology development without prompting, relegating accessibility as a niche subdomain to mainstream technology design and development.

Design Thinking

Not all computer science curricula include design thinking principles and practice, but the number is growing. Far from the traditional waterfall method, computer science education research highlights the benefits of abductive reasoning in software design and engineering problem solving, including design thinking's close relationship with "computational thinking" (Hu, 2011, 2016). Alongside technical skills, developing design skills, by way of learning design thinking, has become sought after. Using abductive reasoning to create new forms, design thinking emphasizes rational (Simon, 1969) and reflective (Schön, 1987) envisioning of new artifacts. In contrast with deductive approaches common in functional requirements gathering, reflective and iterative design thinking involves a cyclical create-evaluate-revise approach, not to achieve a perfect solution, but toward an idealized particular (Brown, 2008; Stolterman & Nelson, 2012), generally:

1. Understand the user and develop empathy
2. Define the specific challenge being addressed
3. Explore multiple ideas at once through ideation
4. Prototype quickly and often for eliciting feedback

5. Test and iterate

Design thinking principles in HCI are commonly applied via user-centered design (UCD) (Gould & Lewis, 1985), encapsulating the above process in a way that centers on users' needs and preferences. In teaching design in computer science, Hu examined metrics used to evaluate software design effort, and emphasized "the ways [students] approach design decision making" (Hu, 2016) as more important than the artifacts themselves when students are learning. In the spirit of Hu's argument, we assessed how incorporating accessibility impacted learning about design, and if artifacts designed from an accessibility-driven approach met disabled[3] and nondisabled user needs. Students in our course learned design thinking concepts and principles, applied through typical user-centered methods and tools. We show that teaching accessibility in a design thinking course did not impede students' "approach to design decision making" (Hu, 2016).

Accessibility in Design

Like Bigelow, Ludi, and Waller's approaches to teaching accessibility, we view the key element of design and accessibility as working directly with people with disabilities in at least one point in the design process. It is hard to design for disability without receiving input directly from people with disabilities. In this vein, researchers have defined a variety of ways to design for disabled user populations, including: Design for User Empowerment (Ladner, 2015), emphasizes increasing the number of people with disabilities in technology design disciplines; User Sensitive Inclusive Design (Newell, Gregor, Morgan, Pullin, & Macaulay, 2011), encourages designers to get to know users with disabilities as part of their design work; Universal Design (Bigelow, 2012; Danford, 2003), employs an access-for-all approach in technical design, and in curricula and classroom design for students with disabilities; and, Ability Based Design (Wobbrock, Kane, Gajos, Harada, & Froehlich, 2011), focuses effort on the abilities users do have. We distinguish our approach by requiring student designers to include users with and without disabilities, and structuring the course with multiple face-to-face sessions.

Method

We conducted an IRB-approved design course study investigating how students learned design when we incorporated accessibility in overall design requirements. The introductory course was part of an informatics-based curriculum, focusing "on computer systems from a user-centered perspective and study[ing] the structure, behavior and interactions of natural and artificial systems that store, process and communicate information[4]." Course learning objectives were that students demonstrate an ability to: (1) create a technical design to a specific prompt, (2) complete the various stages of the UCD process while working with a user, (3) incorporate components into a final design concept, (4) produce a usable prototype exemplifying that concept, and (5) adequately communicate that concept for development. The 10-week course introduced design thinking concepts through readings (course texts included (Buxton, 2007; Norman, 1988)) and lectures, and students applied these concepts through techniques and tools based in the UCD tradition (Gould & Lewis, 1985). Design thinking concepts and UCD methods covered included needs assessment, user interviews, brainstorming, ideating, synthesizing, low-fidelity prototyping, high fidelity prototyping and user testing. Students were instructed to design for users with and without disabilities. At the beginning of the course, a blind guest speaker covered general etiquette tips in a question and answer forum to prepare students to work with people with disabilities. Every week, students had a lab section in which they applied the concepts covered in class. In every other lab section, students worked directly with users with disabilities to test ideas and elicit feedback. At the end of the course, students presented their work, evaluated by users with disabilities.

Participants

The course had 42 students. Twelve were female. Students did not have disabilities or a formal design background, and few interacted with people with disabilities previously. Introductory programming courses were a prerequisite and students had a basic understanding of coding techniques, data structures, and simple algorithms. Users with disabilities were recruited from local listservs, were visual or hearing impaired, and were referred to throughout the course as "expert users" to emphasize their expertise in using assistive technologies. Each student project group was assigned to work with one expert user during the entire course. Students met with expert users 4 times and expert users evaluated student designs during final presentations.

Course Work and Projects.

Students worked in randomly assigned groups; each group was assigned one of two projects to work on throughout the term. Groups 1-6 worked with users with visual impairments and were tasked with designing a real-time augmented reality navigation application. Groups 7-11 worked with users with hearing impairments and were tasked with designing a real-time live captioning application (see Tables 1 and 2).

Lectures were divided into lecture and activity sections where design thinking concepts were covered in a traditional lecture format, and students collaborated to apply the lessons relevant UCD technique in short, in-class activities. Lab sections were held once a week, during which students applied the week's concepts and techniques toward their project. Concepts were introduced successively such that students added to their projects as the course progressed. Expert users attended lab sections, allowing students to practice their design skills with and without expert users. Expert users assessed student work at the ideation, synthesis, prototyping and testing stages. Assignments followed the design process: interview protocols and summaries showed that students asked appropriate questions and were able to brainstorm from what they learned. Although students were given specific lab activities incorporating elements of each concept, assignments were typically completed outside of class time.

Table 1. "Project A" Real-Time Augmented Reality Navigation, groups and expert users.

Group	Student Designers	Expert User
G1	S12 (M), S22 (M), S41 (M), S31 (F)	E1 (M), Blind
G2	S1 (F), S26 (M), S28 (F), S36 (M)	E2 (M), Blind
G3	S19 (M), S21 (M), S33 (M), S35 (F)	E3 (F), Low Vision
G4	S6 (F), S8 (M), S23 (M), S34 (M)	E4 (F), Low Vision
G5	S2 (F), S9 (M), S15 (F)	E5 (F), Blind
G6	S11 (M), S13 (M), S25 (F), S42 (M)	E6 (F), Blind

Table 2. "Project B" Real-Time Live Captioning, groups and expert users.

Group	Student Designers	Expert User
G7	S14 (M), S30 (M), S38 (M), S39 (M)	E7 (F), Deaf
G8	S3 (M), S5 (F), S20 (M), S32 (M)	E8 (F), Hard of Hearing
G9	S10 (F), S16 (F), S29 (M), S37 (M)	E9 (M), Deaf
G10	S4 (M), S7 (M), S27 (M)	E10 (F), Hard of Hearing
G11	S17 (M), S18 (M), S24 (F), S40 (M)	E11 (M), Deaf

Data and Analysis

Study data comprised student assignments, including interview protocols and summaries, conceptual models, brainstorm ideas, sketches, low and high fidelity prototypes, usability heuristics, and user test results. Each group created a design specification detailing the form and operation of the final design, and each student completed a process book describing his or her own experiences and individual contributions to the project. Students completed 1-page reflective journal writing assignments each week. Journal prompts asked students to reflect on each week's topic, technique, lab activity or the student's overall experience. For example, after spending time on the concept of ideation, including several brainstorming sessions with and without expert users, students were asked: "How did your feedback session with your expert user go? What did you learn from the session? Do you feel that you received helpful feedback? Why or why not? What could you have done better? How will your group use this feedback?" A summative survey at the end of the term captured student thoughts of the course overall.

Qualitative analysis was conducted on the data described above, via openly coding student assignments, with particular focus on student journal entries. Following the open coding analysis methods of Miles and Huberman (Miles & Huberman, 1994) in the spirit of grounded theory (Glaser & Strauss, 1967), two of the researchers independently coded 10% of the journals, then via discussion and combined analysis, refined the codes before one researcher coded the rest. Once categories and themes were assessed and discussed, researchers focused on student learning and outcomes: where did students demonstrate an understanding of main concepts and how were designs exemplars of student learning? Criteria for assessing designs included how well students identified and addressed expert user needs (as evaluated by expert users and instructors), prototyped ideas, and evaluated usability (were students able to assess designs with usability heuristics?). Analysis prioritized how students applied UCD techniques and design thinking concepts, how they worked with expert users, and the quality of final designs.

Findings

We found that incorporating accessibility in a design thinking course can be accomplished via a holistic approach requiring accessibility as a key element in overall design. Specifically, we intentionally set expectations that design thinking concepts and UCD techniques ought to sufficiently address accessibility and usability requirements for users *with and without* disabilities. We assessed student understanding of design thinking concepts, and how well students used UCD techniques and tools to create designs. Although some students encountered challenges in specific techniques, they were able to adapt concepts and tools to create complete designs. We identify challenges and discuss implications for implementation of future courses.

Assessing Student Learning

All groups completed all phases of the design process as indicated by their assignment completion rate and successful presentation of final designs. Students demonstrated skill in presenting ideas and eliciting feedback when working with expert users throughout the course and in individual assignments, and final projects sufficiently met course expectations. Though they sometimes had mixed success in producing quality assignments (e.g., some students quickly found that some interview questions yield more useful responses than others), student journals gave insight into if and how they rebounded from less than stellar design activities. We discuss how we assessed student learning, describing what was required in the sequential design thinking process, how students reflected and demonstrated a grasp of key concepts, and how student assignments were judged.

Learning Design Thinking

We looked to the quality of completed assignments to gauge understanding at each stage of the design thinking process (Figure 2) and to assess students' ability to keep pace with the course. The sequential aspect of UCD (brainstorming before prototyping, etc.) and the schedule of meetings with expert users meant students had to keep up. If students failed to keep up with pre-requisite concepts, they would have difficulty completing subsequent assignments and activities. For example, students had to whittle down 90 brainstormed ideas required of the ideation assignment to synthesize a suitable idea to prototype. A prototype not driven by a common vision would be less likely to succeed in obtaining the kind of feedback needed to move to the next stage. Students needed to understand design constraints and develop a conceptual model before a prototype could begin to take shape. Further, expert user meetings motivated students to be well prepared before each session, as S23 explained:

> We would spend the previous evenings before every meeting with E4 preparing prototypes [so] that once she saw them, she was instantly able to help us improve in almost every way. (S23, Process Book)

Our in-class observations showed students engaging in design ideas, debating the pros and cons of each choice or attribute, and demonstrating an understanding of design concepts—such as constraints and reflection. Journals corroborated students' grasp of each step of the process toward an understanding of the overall course learning objectives. For example, S10 wrote:

> I feel that my sense for design has improved especially from the readings that include examples of studies, design approaches, and design techniques. The examples that Buxton and the other authors present give me a solid idea of how the design process works and how to make the process successful. I have come to really appreciate the different steps individually and together. Feedback is just as important as the brainstorming and sketching can be quick and easy while extremely valuable. (S10, Journal6)

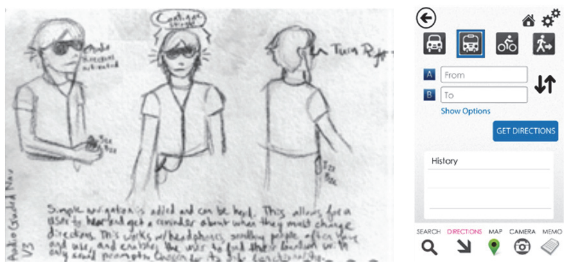

Figure 2. Completed assignments demonstrated students' grasp of concepts. (L) G4's sketch shows a smartphone with headphones directing a user. (R) A screenshot from G3's high fidelity prototype.

S30's understanding of iteration led him to value the sequence over selecting ideas too early in the process:

> Sketches are now no longer "prototypes" or firm ideas like I used to think they were but now just a way of me putting some start of an idea on paper and then having something to build off of. (S30, Journal7)

Working with Expert Users

Working with disabled users was initially challenging for students who were not familiar with disabilities, and who did not at first view their role as designers for people with disabilities. However, students quickly adjusted perceptions and addressed design requirements as the course continued, engaging users in a productive way. We observed students interacting professionally and generatively with expert users, asking questions specific to design and use, and conducting feedback and testing sessions focused on improving designs. A challenge in working with users, whether disabled or not, is that designers have to make on-the-fly changes to accommodate unexpected changes; S5's group was unprepared for initial outcomes:

> We had presented [to] the expert user our some of our rough ideas and the final selected sketch, the expert user affirmed us that our idea looked great, and then the conversation was pretty much done in about 20 minutes. Because the meeting was so pale in context, we had to pull out our very first draft on paper prototype to prevent the awkwardness of nothing to talk about. (S5, Journal8)

Instead of wasting time, they adapted to get as much information as possible from their expert user. In hindsight, the students should have been better prepared for their user. They reflected on their actions (e.g., should have prepared better questions) and made adjustments to improve their situation (asking pointed questions on on half-baked prototypes). Thus, students were able to follow the design process and meet the learning goals because they also adapted to accessibility challenges as they arose. Next, we show how student work illustrated how well students learned design, and how they incorporated accessibility.

Evaluating Design Artifacts

All student groups produced a high-fidelity prototype that was tested with expert users, demonstrating that students were successful at identifying and translating user needs into designs. User test results were mixed: some expert users had no trouble using student designs, while others found bugs. We recognize the non-trivial effort required in creating a high-fidelity prototype to a level that can be tested and we consider a variation in user testing outcomes typical, especially in an introductory design course. In this section, we break down how design specifications, student reflections, and process books provide evidence of students substantively and reflectively meeting course learning goals, particularly around designing an accessible solution.

Design Specifications

Each group was required to create a specification communicating the technical details of their design's expected user interaction in a way that a software developer could understand and implement. Design specifications were assessed on the level of detail necessary for user interface implementation and were not required to have code snippets or algorithms. Given that accessibility was inextricably tied to all course requirements, design specifications were expected to incorporate elements of accessible interactions. Although we could not be sure how developers would implement the specifications, the course instructor had prior experience as a software engineer and in creating design specifications for developers, and could adequately assess the work.

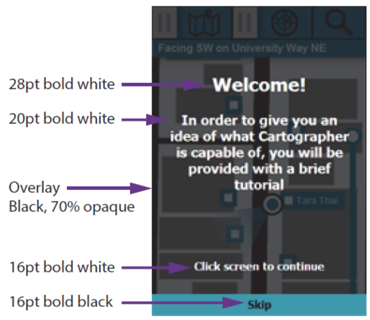

Figure 3. This screenshot from G4's design specification shows descriptions of each element within the interface.

As expected in an introductory design course, specifications varied in quality, but all contained minimal required elements: screenshots, details about specific elements of a user interface (Figure 3), descriptions of key elements of the design (the level of detail varied by group), how it should operate and be used, and a rationale for each major design decision (as required by the assignment). Specifications demonstrated that course learning objectives were met. Students: (1) could create a technical solution to a specific design prompt, (2) complete the various stages of the UCD process while working with a user *with a disability*, (3) incorporate *accessible* components into a final design concept, (4) produce a usable prototype exemplifying that concept, and (5) adequately communicate that concept for development. (Emphasis added to highlight how students met expectations for accessibility incorporated in learning objectives).

Student Reflections and Process Books

To support design specification evaluations, we referenced student journals and individual process books to corroborate their knowledge of course concepts. S23 described how his group incorporated expert user feedback to improve their design:

> …E4 still was often confused trying to navigate our prototype. Due to [her] feedback, we concluded that it was not clear with the amount of options we had at the top of the screen what view mode she was in and would be going into. Following this meeting, our group sat down and tried to simplify our design, and even considered cutting one or two of the views. But [teammate S34] came up with a solution that even let us add a view while making the system similar for the user. …[In] a two button toggle system,…One toggle would be the Directional/Global toggle to determine the user was looking all around them or in a specific direction and the other would be List/Map to determine how the user wanted to data displayed. (S23, Process Book)

Toward course conclusion, journals offered detailed reflections what students felt they learned:

> I really enjoy the fact that we are learning about the design process by getting the chance to apply every step and practice it... I am getting a lot out of this method, and I will retain this knowledge far better with memories of my experiences with it. I've been going through a cycle with feeling mildly overwhelmed with what the next step requires me to accomplish (such as all the options and decisions we had to make for our prototype) and once I break it down into the required assignments it suddenly ends up conquered. (S2, Journal7)

Challenges

Our findings indicated that students grasped general design concepts while tasked to create accessible designs, but they also faced challenges specific to designing for disabled and nondisabled users. Groups with hearing impaired users sometimes had difficulty communicating, and receiving feedback on sketches was challenging for groups with visually impaired users. As with any project involving people with disabilities, finding users was difficult, particularly volunteers who could participate at the same time and place with students. With financial support, we mitigated typical issues with recruiting and scheduling by providing generous compensation to expert users. We note that having resources can make a difference in finding people from non-typical user populations to work with. In this section, we identify how challenges impacted learning goals, and highlight methodological and substantive challenges.

Inaccessible Design Methods

We changed as little of the design process as possible to expose accessibility issues, and working with disabled expert users revealed where the UCD methods and tools used in this course assume nondisabled users. Visual techniques—such as sketching or paper prototyping—assumed sighted users and students struggled to find ways to make each successive step accessible in a way that showed progress. Inaccessible techniques and tools highlighted where the UCD process overall tends to assume users and designers without disabilities. These challenges emphasize opportunities to develop alternative, accessible design methods.

Lack of Disability-Specific Knowledge

Students began the class with little knowledge or experience of disability, and though the initial Q&A with the blind guest speaker was helpful, it was their only introduction and was insufficient in contextualizing experiences of using accessible technologies that students would come to rely on. Students spent considerable time becoming acquainted with disability-specific technical knowledge and many took it upon themselves to learn more about disability. Inaccessible design methods made it difficult for students to meet specific learning objectives of eliciting feedback and creating testable prototypes. Students overcame these challenges in creating minimally usable and accessible designs. Yet, we cannot be sure of how much more students could have accomplished if tools at their disposal were accessible. Unfortunately, such challenges revealed barriers to a truly accessible design *process*, though they did not critically block students' ability to create designs for their expert users.

Discussion

We discuss the implications of our findings and offer suggestions for similar courses in the future. Students in our study were able to meet course objectives and complete a design project while incorporating *accessible design*. Despite challenges, barriers to *teaching* accessibility are low. We add specific insights to previous work in this area (Poor et al., 2012; Putnam et al., 2015; Rosmaita, 2006) for future design thinking courses: (a) expectations should include accessibility, (b) students should work with disabled users throughout the design process, (c) students should be required to create designs for both users with *and* without disabilities, (d) courses should include instruction on disability etiquette and existing accessible technologies.

The challenges identified in the course highlight opportunities to improve the accessibility of the design process. Establishing a baseline of technical accessibility or awareness of issues around disability would equip students to take on disability and design, improving the design outcomes. For example, students could learn what orientation and mobility skills a blind person might have or what kinds of technologies are accessible. Facilitating accessible methods (paper prototyping for blind users?) and creating accessible tools enables students to more thoroughly engage with accessible aspects of design. Further, accessible methods and tools are useful for any designer and could be used in practitioner and pedagogical settings.

Conclusion

Students in our study reported an increased awareness of implications for inaccessible design; and changed their perspectives that accessibility is "someone else's job" to understanding their role as designers in creating an accessible future. We found that in immersing accessibility, students grasped concepts around design thinking and (1) created technical solutions to design prompts, (2) completed the UCD process while working with a user *with a disability*, (3) incorporated *accessible* elements into final designs, (4) produced usable prototypes, and (5) adequately communicated their designs for development. Incorporating accessibility in design is yet imperfect due to inaccessible methods and tools, but teaching accessibility in a design thinking course was effective when resources were in place and when expectations included accessible outcomes.

Endnotes

(1) https://www.section508.gov/
(2) https://www.ada.gov/
(3) It is accepted practice in disability studies to use 'people with disabilities,' 'disabled people,' and 'nondisabled' to center disability (Linton, 1998).
(4) Classification for Instructional Programs: http://nces.ed.gov/ipeds/cipcode/cipdetail.aspx?y=55&cipid=89325

References

Bigelow, K. E. (2012). Designing for Success: Developing Engineers Who Consider Universal Design Principles. *Journal of Postsecondary Education and Disability*, *25*(3), 211–225.

Brown, T. (2008). Design Thinking. *Harvard Business Review*, *86*(6), 84–92.

Buxton, W. (2007). *Sketching user experiences: getting the design right and the right design*. Amsterdam; Boston: Elsevier/Morgan Kaufmann.

Cook, A. M., & Hussey, S. M. (2002). *Assistive technologies: principles and practice*. St. Louis: St. Louis : Mosby.

Danford, G. S. (2003). Universal Design People with Vision, Hearing, and Mobility Impairments Evaluate a Model Building. *Generations*, *27*(1), 91.

Glaser, B. G., & Strauss, A. L. (1967). *The Discovery of Grounded Theory*. Chicago, IL: Aldine Publishing Co.
Gould, J. D., & Lewis, C. (1985). Designing for usability: Key principles and what designers think. *CACM*, *28*(3), 300–311.
Hu, C. (2011). Computational thinking: what it might mean and what we might do about it. In *Proc of the 16th conf on Innovation and technology in computer science education* (pp. 223–227). Darmstadt, Germany: ACM.
Hu, C. (2016). Can Students Design Software?: The Answer Is More Complex Than You Think. In *Proceedings of the 47th ACM Technical Symposium on Computing Science Education* (pp. 199–204). Memphis, Tennessee, USA: ACM.
Ladner, R. E. (2015). Design for user empowerment. *Interactions*, *22*(2), 24–29.
Linton, S. (1998). *Claiming Disability: Knowledge and Identity*. New York, NY, USA: New York University Press.
Ludi, S. (2007). Introducing Accessibility Requirements through External Stakeholder Utilization in an Undergraduate Requirements Engineering Course. In *Proc. Soft. Eng. '07* (pp. 736–743). IEEE Computer Society.
Miles, M. B., & Huberman, A. M. (1994). *Qualitative data analysis: an expanded sourcebook*. Thousand Oaks: Sage Publications.
Newell, A., Gregor, P., Morgan, M., Pullin, G., & Macaulay, C. (2011). User-Sensitive Inclusive Design. *Universal Access in the Information Society*, *10*(3), 235–243.
Norman, D. (1988). *The Design of Everyday Things* (2002 Edition). New York: Basic Book.
Poor, G. M., Leventhal, L. M., Barnes, J., Hutchings, D. R., Albee, P., & Campbell, L. (2012). No User Left Behind: Including Accessibility in Student Projects and the Impact on CS Students Attitudes. *Trans. Comput. Educ.*, *12*(2), 1–22.
Putnam, C., Dahman, M., Rose, E., Cheng, J., & Bradford, G. (2015). Teaching Accessibility, Learning Empathy. In *Proc ASSETS 2015* (pp. 333–334). Lisbon, Portugal: ACM.
Rosmaita, B. J. (2006). Accessibility first!: a new approach to web design. In *Proc SIGCSE 2006* (pp. 270–274). Houston, Texas, USA: ACM.
Scherer, M. J. (1993). *Living in the state of stuck: how technologies affect the lives of people with disabilities*. Cambridge, MA: Brookline Books.
Schön, D. A. (1987). *Educating the reflective practitioner*. San Francisco: San Francisco : Jossey-Bass.
Simon, H. A. (1969). *The sciences of the artificial*. Cambridge: Cambridge, M.I.T. Press.
Stephanidis, C., Akoumianakis, D., Sfyrakis, M., & Paramythis, A. (1998). Universal accessibility in HCI: Process-oriented guidelines and tool requirements. In *Proc. User Interfaces for All '98*.
Stolterman, E., & Nelson, H. G. (2012). *The design way: intentional change in an unpredictable world*. Cambridge, Massachusestts: Cambridge, Massachusestts.
Waller, A., Hanson, V. L., & Sloan, D. (2009). Including accessibility within and beyond undergraduate computing courses. In *Proc. ASSETS '09* (pp. 155–162). Pittsburgh, PA, USA: ACM.
Wenger, E. (1998). *Communities of practice : learning, meaning, and identity*. Cambridge, U.K.: Cambridge, U.K.
Wenger, E., & Lave, J. (1991). *Situated learning: legitimate peripheral participation*. Cambridge [England]: Cambridge England.
Wobbrock, J. O., Kane, S. K., Gajos, K. Z., Harada, S., & Froehlich, J. (2011). Ability-based design: Concept, principles, and examples. *ACM TACCESS*, *3*(3), 1–27.

Acknowledgments

This work was supported in part by the National Science Foundation under grants IIS-1217627, IIS-1230435, and IIS-0952786. Any opinions, findings, conclusions or recommendations expressed in this work are those of the authors and do not necessarily reflect those of any supporter listed above.