Teaching Tip
How to Teach Information Systems Students to Design Better User Interfaces through Paper Prototyping

Michael J. Scialdone and Amy J. Connolly

Teaching Tip
How to Teach Information Systems Students to Design Better User Interfaces through Paper Prototyping

Michael J. Scialdone
Department of Information Systems and Operations Management
Texas A&M University
College Station, TX 77843, USA
mscialdone@mays.tamu.edu

Amy J. Connolly
Computer Information Systems & Business Analytics
James Madison University
Harrisonburg, VA 22807, USA
conno3aj@jmu.edu

ABSTRACT

Given the ubiquity of interfaces on computing devices, it is essential for future Information Systems (IS) professionals to understand the ramifications of good user interface (UI) design. This article provides instructions on how to efficiently and effectively teach IS students about "fit," a Human-Computer Interaction (HCI) concept, through a paper prototyping activity. Although easy to explain, the concept of "fit" can be difficult to understand without repeated practice. Practically, designing "fit" into UIs can be cost-prohibitive because working prototypes are often beyond students' technical skillset. Accordingly, based on principles of active learning, we show how to use paper prototyping to demonstrate "fit" in a hands-on class exercise. We provide detailed step-by-step instructions to plan, setup, and present the exercise to guide students through the process of "fit" in UI design. As a result of this activity, students are better able to employ both theoretical and practical applications of "fit" in UI design and implementation. This exercise is applicable in any course that includes UI design, such as principles of HCI, systems analysis and design, software engineering, and project management.

Keywords: Human-computer interaction (HCI), Paper prototyping, Active learning, Constructionism, Teaching tip

1. INTRODUCTION

With computing devices peppering nearly every aspect of our lives, how people interact with these technologies is critically important to all computing fields. In fact, failure to properly account for interface usability can bring about a range of problems from minor annoyances to literal disasters. Instances of poor usability have been identified in common computing interfaces, including Windows, Apple products (notably Apple Watch and Apple Maps), Smart TVs, and social media (Burton, 2016; Pogue, 2016). Beyond day-to-day annoyances, poor design can have catastrophic results. For example, in 1988, due to an inadequate visual display, the U.S. Navy shot down an Iranian civilian flight (Pogue, 2016); while in 2003, Space Shuttle Columbia burned up upon re-entry, partly due to a poorly designed PowerPoint presentation (Park, 2015). "Complex technical information – mass, velocity, etc. – disappeared behind bullet points. Lower-level bullets

mentioned doubts about safety, but top-level points (and the summary) were optimistic" (para 5); and as a result, executives at NASA incorrectly concluded that re-entry would be safe.

Human-Computer Interaction (HCI) is concerned with "the ways that humans interact with technologies for various purposes" (Zhang and Li, 2005, p. 228). The central phenomenon of interaction is the user interface (UI) which provides the main point of functionality connecting human objectives and computing resources. The practical goal of HCI is to "achieve high usability for users of computer-based systems" (Hartson, 1998, p. 103). Usability is broadly defined as the degree of satisfaction, efficiency, and effectiveness that an information system (or components thereof) provides to an end user in respect to an intention or end-state (International Organization for Standardization, 1998). It is important to understand how peoples' abilities and limitations impact interface usability in order to create and evaluate "better, more successful" technologies (Cooper et al., 2014, p. XXIII) that

avoid frustration (or even disaster) and promote well-being. While those who work professionally in information systems (IS) should know better, all computing professionals could stand to learn more HCI concepts given the continued recurrence of poorly-designed interfaces.

Chan, Wolfe, and Fang (2002) emphasize the importance of teaching HCI and usability-related topics within IS, noting that awareness of these is absolutely essential for graduates. That is, IS students must comprehend how critical UI design is when working with developers to implement efficient and effective systems, as well as how to select and evaluate existing tools to "fit" and facilitate workflows. Accordingly, students should learn, at a minimum, basic principles of HCI (Faiola, 2007) through instruction that facilitates experience with "methods and skills to understand current users, to investigate non-use, and to imagine future users" (Churchill, Bowser, and Preece, 2016, p. 70). However, practical training with UI design and evaluation is often underemphasized. For example, the 2010 IS Model Curriculum considers HCI only an elective (Janicki, Cummings, and Healy, 2015).

With a strong focus on technical characteristics of technology and other requisite business-oriented topics such as project management, the empathetic aspect of systems design and implementation is arguably not one of the more salient characteristics of IS curricula. At best, the interaction between designers and users may be touched on with respect to requirements engineering and validation phases of the systems development life cycle, yet HCI concepts such as UI design are unlikely to be underscored to, much less practiced by, students. As such, we fear that all too often IS students gain limited knowledge of basic usability concepts and that faculty find minimal opportunities to teach them. In an ideal IS degree program, a required 100 or 200-level HCI course would be worthwhile; yet, we recognize that such is not necessarily viable (especially in light of the Model Curriculum).

To this end, we present an activity in the form of a problem-based scenario to give students experience with UI development through an exercise that contains aspects of both design (requiring reflection on HCI concepts) and implementation (through role-play that highlights how users employ these designs) in which students create and test paper-based UI prototypes.

In our execution of this activity, students were required to complete the exercise during an HCI class without being able to reference outside materials such as textbooks or digital sources. We enforced these constraints to promote consistency and originality as students could then only draw from their own knowledge and imaginations. While these conditions suited our learning outcomes for the specific class (largely based on the concept of "fit" that we address later on in this paper), we recommend other instructors adapt this activity as needed.

Given the scant resources and time constraints that instructors frequently face, our activity can be used at any point in the IS curriculum from the Introduction to MIS course through Advanced Systems Analysis and Design – everywhere that students need to learn and practice usability concepts. Because the materials to set up this activity are commonly available (e.g., paper and pens), students do not need *a priori* technical skills, such as coding or wire-framing. As a result, the activity is widely accessible, and the materials are relatively inexpensive compared to hardware and software resources involved in building a functional, digital prototype. This exercise is easily adapted for any IS course, and could work for K-12 students to build excitement about computing. In this paper, we explain how it was implemented in an Introductory HCI course as a way to emphasize the importance of usability.

2. BACKGROUND CONCEPTS

This exercise is rooted in the pedagogical philosophy of Constructionism. Papert (1991, p. 2) writes that this approach "boils down to demanding that everything be understood by being constructed." The central idea is that as students construct artifacts, they apply theory, concepts, and ideas in a way that is relevant to them; and in doing so, they become active participants in their own learning. In other words, they build to understand. Through this pedagogical lens, the instructor becomes a facilitator working to consult, clarify, encourage, and support students in need, rather than playing "sage on the stage." Given the aforementioned shortage of HCI training in IS curriculum, and for those instructors wishing to brush up on this subject, we briefly provide background knowledge on paper prototyping, the concept of "fit," and a few suggested prerequisite concepts to prepare students to maximize their learning from this activity.

2.1 Paper Prototyping
Most IS faculty are familiar with the concept of prototypes but may be less familiar with ways to implement them in a classroom. A prototype can be conceived of as a hypothesis in the form of a preliminary design for a problem, tested by how users engage with this design (Pernice, 2016). In software development, prototypes can be used to receive user feedback and save money because "it's 100 times cheaper to make a change before any code has been written than it is to wait until after the implementation is complete" (Nielsen, 2003, para 6). Paper prototyping, also known as low-fidelity prototyping (Rettig, 1994) or throwaway prototyping (Vijayan and Raju, 2011), was popularized by IBM during the 1980s. In its simplest form, it is "building prototypes on paper and testing them with real users" (Rettig, 1994, p. 1). Although such prototypes may seem crude, research has shown that the feedback they enable is of nearly the same quality and quantity as computer-based prototypes (Sefelin, Tscheligi, and Giller, 2003). An example of a paper prototype created by one of our students is shown in Figure 1 as a point of reference.
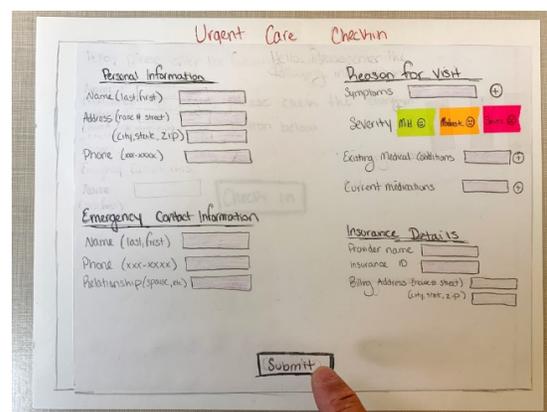
Figure 1. Sample Paper Prototype

Paper prototyping in the classroom allows students to express their understanding of basic concepts as they construct and evaluate quick, throw-away UIs. This method can be used to introduce good design concepts such as usability and "fit," as well as to test comprehension of these concepts. When learners engage in paper prototyping, they design an interface, test it quickly, and then reflect on what worked or didn't, and why. Further, applying paper prototyping at multiple points in a class provides an opportunity to observe students' comprehension and growth at different points in the semester.

2.2 Key Concepts in Human-Computer Interaction

Usability in HCI is often defined in terms of affordances and constraints. Affordances are "the design aspects of an object which suggest how the object should be used; a visual clue to its function and use" (Chamberlain, 2010, p.169; citing Norman (1988)) and constraints are the "limitations of the actions possible perceived from the object's appearance" (Norman, 1988). For example, an affordance of a keyhole is that the opening invites something to be inserted; while a constraint is that its small size and narrow width limits the range of what somethings could reasonably be inserted. Similarly, on a UI, a text field invites the user to input character data (affordance) while its size and meta-properties can limit input options to certain kinds and lengths of characters such as a 4-digit pin number (constraint). Students first need to know about affordances and constraints, at least minimally, in order to build an understanding of "fit."

"Fit" is a fundamental concept of usability that is defined based on three separate, but related, dimensions: physical fit, cognitive fit, and affective fit (Te'eni, Carey, and Zhang, 2005). *Physical fit* addresses the input/output mechanics of technology with respect to human physiology. This concept is roughly synonymous with ergonomics and, to some extent, accessibility. Ideally, good physical fit minimizes physical effort while maximizing productivity (Te'eni, Carey, and Zhang, 2005). *Cognitive fit* assumes that when the UI and its feedback mechanisms are consistent with users' previous experiences, skill sets, and mental models, users can accomplish tasks effectively and efficiently. In other words, "the problem representation and the task both emphasize the same type of information" (Vessey and Galletta, 1991, p. 67). Finally, *affective fit* considers how positive affect can be enhanced, negative affect can be minimized, or another desired affective state can be influenced through functional (an object that a user interacts with) or non-functional (non-interactable properties such as color, font, etc.) UI design characteristics (Avital and Te'eni, 2009). These three types of "fit" are often (but not exclusively) demonstrated through the intentional design of affordances and constraints.

As a Constructionist activity, paper prototyping allows students to practice designing affordances and constraints and then to play with the various dimensions of "fit." To strengthen this relationship of the activity to the three facets of "fit," we rooted our activity's problem within a context that would encourage learners to reflect, particularly, on users' potential physical, cognitive, and affective states.

3. PAPER PROTOTYPING ACTIVITY

We executed this activity twice per semester (the rationale for which we explain in the next section) for two semesters. For clarification of terminology, we designate each execution of the activity Exercise Iteration 1 (EI1) and Exercise Iteration 2 (EI2) and each semester Term 1 (T1) and Term 2 (T2). As each iteration resulted in a paper prototype, there were two prototypes per student per term. We refer to these as Prototype Version 1 (PV1) and Prototype Version 2 (PV2).

For each iteration, we prefaced the activity by introducing (or reintroducing) students to paper prototyping through a short video of a testing session. In the clip, one person tests the prototype (like a user) while another manipulates pieces of paper (like a computer) to demonstrate interaction (Yun, 2007). Students were then instructed to design an interface for a device using paper-based, hands-on materials. They were advised it would not be graded, but that they were expected to participate. Inspired by Snyder's (2001) guidelines, the following rules were given:

- Once prototypes were finished, students were to pair up
- One student role-played "computer" while their partner played the "user;" then they switched roles
- While in the "computer" role, a student presented his or her prototype to the "user"
- The "user" could click/tap on paper objects using his/her fingers while the "computer" manipulated the prototype accordingly to simulate interface behavior
- The "user" could simulate inputting character text however they wanted (pretending to type, speaking/voice, selecting an option, etc.)
- The "computer" was not allowed to speak or gesture hints about how to use the prototype; their role was simply to simulate or facilitate functionality

Students were then shown a PowerPoint slide (Figure 2) with instructions to design an urgent care check-in kiosk.

In-Class Activity

You're designing self check-in kiosks for an Urgent Care waiting room. When patients arrive, they can check in, and receive an estimated appointment time.

The prototype must allow (but is not limited to) the following inputs:
- Name, phone, and residential address
- Emergency contact details
- Insurance details (including billing address)
- Existing medical conditions and medications taken

The prototype must provide (but is not limited to) the following output:
- Estimated wait time
- Name of the health care professional the patient will see

Figure 2. Activity Instructions

For our purposes, we found the following supplies were sufficient, although quantities may differ based on class size: 11x17" cardstock paper, 8.5x11" printer paper, 3x5" notepad paper, sticky notes and/or labels, index cards (x100), no. 2 pencils with erasers (plus extra erasers), colored pencils, scotch tape, glue sticks, paper clips, binder clips, and scissors.

Although students were not told which materials to use to avoid unintentionally influencing their designs, some common practices emerged that instructors might consider. Card stock often represented main "screens" due to its sturdy nature. Sticky notes tended to work well for large buttons or dialog boxes, while scissors were helpful to customize sizes and shapes of screen items. Colored pencils proved useful for highlighting and differentiating objects on the "screen".

As the course periods in which our activities were conducted lasted only 80 minutes, students spent approximately 55 minutes working on their prototypes, during which time the instructor facilitated and observed how the activity played out, supporting and encouraging students as needed. Students then spent time alternating role-play of computer and user with multiple classmates. Each pair took about five minutes to test their prototypes before moving on to form new pairs to test them again. During EI1 of T1, role-play was limited to the remaining class time, which restricted opportunities for students to engage with a broad range of classmates. In T2, prototype testing implementations were moved to the following class period, allowing 40 to 45 minutes for testing. Table 1 summarizes the timeline for the revised activity. While our timeline assumes two 75-minute class periods, it could easily be modified for three 50-minute sessions.

Activity Steps	Approximate Duration
Class Period #1	
Step 1: Introduction Play sample video, explain the problem, and allow students to collect materials and tools (scissors, pencils, tape, etc.)	5-10 minutes
Step 2: Construct Prototype Students independently build prototypes with minimal intervention on behalf of the instructor	55-60 minutes
Step 3: Construction Wrap-Up Students finish prototypes and return tools and unused materials	5-10 minutes
Class Period #2	
Step 4: Computer/User Roleplay Students pair up several times with different classmates to play both "user" (testing a peer's prototype by mimicking how they'd interact with a digital version of the design) and "computer" (manipulating their own prototypes based on the interaction of the "user" to simulate how it would behave as a digital artifact).	40-45 minutes
Step 5: Class-Wide Discussion The instructor leads reflective discussion about the activity	30 minutes

Table 1. Lesson Schedule of Paper Prototyping Activity

After engaging in role-play, students were led through a 30-minute face-to-face discussion to reflect on and articulate what they learned and to recognize the value of their experience. Of note, we did not provide students with access to their PV1 before or during building PV2 to ensure they were continuing to harness only their current understandings of "fit." They were, however, allowed to review both PV1 and PV2 after the role-play part of EI2 to ensure they were adequately informed to converse about their experiences with paper prototyping.

Below are the questions we asked to facilitate this discussion. Questions 1, 2, and 5 are applicable for any iteration of the exercise, while 3 and 4 should be asked following EI2.

1. ***Which factors influenced your initial prototype? That is, why did you design your prototype the way that you did?*** Here, students typically point to previous experiences as (at least part of) the influence of their design. As a point for elaboration, instructors should probe students as to why those designs worked in other UIs and if/how they continue to be appropriate for this problem.
2. ***Did you learn anything of value from the "usability tests," either on your prototype or by exploring a peer's prototype?*** This highlights the role of feedback in designing UIs as students identify good practices that peers used that they didn't think of.
3. ***Describe any modifications that went into your revised prototype and specifically why you made any changes that you did.*** This emphasizes that "fit" may be improved and informed through knowledge, iterative design, and evaluation. Here the instructor should challenge them to articulate what they changed to make their design "fit" better.
4. ***Does anything come to mind now for a third prototype that you did not consider in your previous iterations?*** This allows students to consider and reflect on features and functionalities that they did not have the opportunity to implement due to time constraints, material constraints, or any other limitation(s).
5. ***What, if anything, did you learn from this activity?*** This is intended to help students step back and reflect on the degree to which they found meaning in the practice of prototyping, design, and evaluation. This promotes discussion on how designer intentions do not always match user expectations, highlighting the utility of prototyping and testing.

4. EVIDENCE

We conducted two exercise iterations (one at the start and one at the end of the semester) to determine the extent to which students' application (and therefore, presumed understanding) of "fit"-related concepts changed. Although this is not a research paper, we have evidence to suggest that paper-prototyping was effective. Therefore, in this section, we share our observations across both T1 and T2 and summarize relevant student feedback gained from focus groups and course evaluations.

4.1 Instructor Observations
As expected, most of the work produced for EI1 across both terms did not reflect much about the problem's context. Students did not generally account for potential physical, psychological, or emotional states of the user at an urgent care center, although a few considered potential physical limitations. For example, despite having an injury or condition that could impact the user's mobility, most PV1s required data to be input through touch-based means (such as a digital keyboard) or a

traditional keyboard and mouse. Meanwhile, affordances as functional clues (often considered a measure of cognitive "fit") were somewhat evidenced in PV1.

EI2 occurred during the penultimate week of class after students learned about the various types of "fit," design principles, and evaluation methods. Generally, students exhibited more awareness of the problem's context in their PV2. One significant common improvement was that multiple students' work reflected an expedited check-in process. For example, on some prototypes, users could provide key information later or request immediate, emergency assistance. Presumably, these options would reduce physical exertion, cognitive load, and/or anxiety by hastening the check-in process. Another common improvement was quick input to alleviate physical effort such as dialog boxes to select country and state (rather than having to type these out) or to swipe one's insurance card in an external reader (which would automatically populate related fields). Furthermore, in the EI1 discussion, students drew heavily on previous experiences to articulate what influenced their designs; in EI2, they still tended to do so but were much more likely to frame their responses within the three dimensions of "fit."

4.2 Focus Group and Course Survey

Following EI2 of T2, a 30-minute focus group was held during class time without the instructor present to garner honest, candid feedback from students to gain insight into their experiences. This was conducted by one of the author's colleagues from outside of his department in hopes that students would feel comfortable being authentic with her.

One student explained that after taking this course,

> every time that I use any sort of technology or interface, and I see something that I don't think works well or looks right, I'm always gonna think back to what we learned that I could do to fix it, or to make it better for somebody, any website, any operating system or anything. I'll always find little things that I don't like about it now, because I know the correct way to do things. So I feel like that will stick for a long time.

When asked "*which activities or assignments did you feel were the most useful or important?*" another student responded

> I thought that the prototyping was the most useful, especially at the beginning because it kind of set the tone for the class, giving us an idea for what we'd be learning about. And then also with doing the prototyping at the end, it gave us a physical example of the progress we made and what we learned throughout the course.

When asked what they would remember about the course in five years, one student commented, "cognitive fit, affective fit, and physical fit ...I don't think that's something that can easily be forgotten. Cuz like, you know, you think about those things but now that we know the technical terms for them and what actually to look for."

Finally, students' average rating of the course was 5.5 out of 6 (81.2% response rate; n=13) during the first semester and 5.6 (82.4% response rate; n=14) during the second. We point to these end-of-the-semester course evaluations as at least partial support of the utility and success of the exercise. This conclusion is based on the fact that paper prototyping was the most prominent activity conducted in the class, and it was designed to draw on every major theme of the course. For example, although the three dimensions of "fit" were not introduced until after the EI1 (about the third week of class), the first few lessons of the course were on usability in general; designed to introduce basic ideas such as *affordance* and *constraint* to establish that beginning vocabulary upon which to tether the forthcoming lessons on "fit." The following 6-7 weeks covered "fit," while the remaining weeks considered computer-supportive cooperative work, ethical design, and usability evaluation methods. "Fit" elements were continually emphasized throughout these lessons.

Given that our prototyping exercise was devised to elicit designs that account for "fit," given that "fit" is the foundation stone of our HCI course, and given that the exercise was employed as bookends around the majority of our classes, we maintain that the focus group and course evaluation results can provide complementary, albeit anecdotal, evidence to support the degree to which paper prototyping succeeded.

5. DISCUSSION

For those readers considering the application of paper prototyping in their classrooms, we have multiple suggestions not addressed above. First, we address other possible courses in which paper prototyping might be employed, then some general tips for implementing it, and, finally, additional student-oriented considerations.

5.1 Other Potential Subjects for Adaptation

We piloted this exercise in an Introduction to HCI elective course over two semesters. The students in this course were majors or minors in Computer Science and/or IS. Based on the evidence of success noted in the previous section and the teaching expertise of the authors, we believe that this exercise would be reasonably fruitful in any course that touches on usability or interface design. For example, we envision that this exercise could be implemented in a Systems Analysis and Design course when discussing Design and Implementation as a phase in the Systems Development Life Cycle (SDLC). In particular, prototypes are noted as one design technique to assess usability and make refinements following requirements gathering and organizing in the Planning and Analysis phases (Valacich and George, 2017).

As an activity, students could be provided with functional and non-functional requirements and then be asked to build and test paper prototypes. Not only might such an activity support practicing and learning usability concepts, but it could also lead to insights in respect to the relevance of non-conflicting and unambiguous requirements documentation. Furthermore, it could underscore the notion that the SDLC is an integrated process rather than separate and distinct phases.

As another example, a version of this activity could work in a Project Management (PM) course. Techniques to estimate benefits and risks through practices such as SWOT analyses and feasibility assessments are typical topics in PM. Prototyping might be used here as a technique to assess the strengths and weaknesses of a particular software-based solution or as a rough

draft to evaluate technical (and economic) feasibility of constructing and implementing the system.

Furthermore, we believe that this exercise also provides meaningful insight applicable across the entirety of an IS curriculum and might therefore be used to introduce HCI concepts in an Introduction to MIS course. Specifically, role-playing computer and user may lead students to recognize that human-based interactions (such as collaborative relationships, trust, and social capital (Kumar, van Dissel, and Bielli, 1998)) are generally integral to IS design and implementation. In other words, the exercise could illuminate the relevance of interaction between IS professionals and actual users.

5.2 Tips for Instructors

Although we designed and executed our prototyping activity with the intention that students work independently, we found that they did reap benefits from minor instructor interactions. Consistent with Constructionism, we suggest that although the instructor is meant to be a facilitator, he or she needn't be a silent observer. Points of praise or gently critical encouragement during design can motivate students. For example, offering comments such as "oh, you're not using colored pencils?" or "what does this mean?" can serve to stir enthusiasm or prod students to connect the activity with course material. Additionally, observations made by the instructor are valuable for the discussion that follows the exercise. By witnessing what students are doing throughout the activity, in both of the roles assigned, the instructor can identify common misunderstandings or opportunities to then address with the class. For example, if a key construct is not observed as a design element, this can be a clue that the construct needs to be clarified or refined in a subsequent lesson or exercise.

Finally, the instructor should remind students (and correct behavior) during the testing periods if the rules for the roles of "computer" and "user" are not properly executed. Because the "computer" represents and manipulates their own design, the "computer" may be tempted to provide hints to the "user" in the form of verbal explanations or non-verbal signs like sighs, glances, or gestures. This is natural because the student wants their prototype to be successful, and its functionality makes sense to them because they built it from their own perspective. However, giving hints goes against the spirit of the exercise as it prevents the "user" from authentic engagement; as a result, the "computer" will not receive genuine feedback.

Additionally, we didn't grade the exercise because we wanted students to be as creative as possible and to feel free to take risks rather than limiting themselves to the criteria of a rubric. Reminding the "computer" that this exercise was not being marked, we suspect, helped to put them more at ease and, therefore, not as tempted to give the "user" clues.

5.3 Considering Different Contexts, Skillsets, and Tools

We required students to build their prototypes in class because (1) we wanted to ensure that the materials students used were consistent, (2) that they did not work with partners, and (3) that their work was not influenced by searching for and borrowing from similar interfaces. The last two points, we believed, were important to make certain that students expressed their own ideas and understandings in the artifacts they created. Yet, requiring in-class design work meant that students who were absent during those sessions missed out on creating an artifact

and that students' work was constrained (or even unduly rushed) by limited class time.

Going forward, despite concerns that students may collude or utilize unauthorized resources for guidance, we plan to allow students to make their prototypes at home. We suspect that the potential benefits will outweigh the concerns because students will have more time to reflect on the problem and imagine viable solutions that flush out their creativity and interpretations of usability. Like any assignment, we will provide specific instructions to minimize our concerns as much as possible, knowing that any implementation will have specific benefits and drawbacks.

There were approximately 15 students each term we employed the exercise, almost entirely Juniors and Seniors in IS or Computer Science majors, with 2 or 3 students across both classes who were either Sophomores or minors in the aforementioned majors. This demographic was largely comfortable with computing technology fundamentals and expressed no concern over the physical requirements of utilizing paper-based materials.

Yet, we recognize that this activity assumes students are both physically capable and wholly comfortable with handwriting, drawing, and paper-based craftwork. Given our reconsideration that prototyping could happen at home, students with physical limitations or less comfort could design their UIs with digital tools (such as PowerPoint or wireframing software) and then create a printout to bring to class.

Regardless of the tools and constraints implemented for this activity, a key aim of paper prototyping is to create a worthwhile learning experience that is equally accessible to nearly everyone. Therefore, we encourage instructors to implement the activity through whichever means allows them to reach the widest range of students possible regardless of a priori technical skills or experience. Such "unplugged" activities hold the potential for creating democratized, meaningful opportunities in computing-oriented classes to build knowledge across a range of student skills sets, expectations, and goals irrespective of backgrounds such as major, race, and gender.

6. CONCLUSION

In this paper, we have written about and reflected on our use of a paper prototyping activity to reinforce the concept of "fit." While research has shown that paper prototypes can produce similarly critical feedback in terms of quantity and quality compared to computer-based prototypes (Sefelin, Tscheligi, and Giller, 2003), to the best of our knowledge, there is little, if any, empirical work to guide IS instructors in applying paper prototyping in the classroom. From our experiences of implementing and observing a paper prototyping activity across two terms of an HCI course, we believe it is a viable means for students to gain hands-on experience in applying concepts of "fit" to an artifact that can be shared and discussed with others. An understanding of "fit," along with the practice of designing UIs based on that understanding, may impress upon students how people matter when developing, implementing, and analyzing information systems. As computing becomes even more ubiquitous in the developed world, user considerations are paramount to effective and efficient interactions with hardware and software (Janicki, Cummings, and Healy, 2015). Therefore,

IS curricula need to continue reinforcing the concept of "fit" in good UI design via educational activities such as the one presented in this paper.

7. REFERENCES

Avital, M. & Te'eni, D. (2009). From Generative Fit to Generative Capacity: Exploring an Emerging Dimension of Information Systems Design and Task Performance. *Information Systems Journal,* 19(4), 345−367.

Burton, G. (2016). Top 10 Worst User Interfaces. *The Inquirer*. Retrieved August 6, 2020, from https://www.theinquirer.net/inquirer/feature/2459940/top-10-worst-user-interfaces.

Chan, S. S., Wolfe, R. J., & Fang, X. (2002). Teaching HCI in IS/EC Curriculum. In *Proceedings of the Eighth Americas Conference on Information Systems* (pp. 1011-1020). Dallas, Texas.

Chamberlain, P. (2010). Horses, Elephants and Camels. Challenges and Barriers to Interdisciplinary User-Centered Design Research. In *DS 60: Proceedings of DESIGN 2010, the 11th International Design Conference* (pp. 163-172). Dubrovnik, Croatia.

Churchill, E. F., Bowser, A., & Preece, J. (2016). The Future of HCI Education: A Flexible, Global, Living Curriculum. *Interactions,* 23(2), 70-73.

Cooper, A., Reimann, R., Cronin, D., & Noessel, C. (2014). *About Face: The Essentials of Interaction Design* (4th ed.). Indianapolis, Indiana: John Wiley & Sons.

Faiola, A. (2007). The Design Enterprise: Rethinking the HCI Education Paradigm. *Design Issues,* 23(3), 30-45.

Hartson, R. H. (1998). Human–Computer Interaction: Interdisciplinary Roots and Trends. *Journal of Systems and Software,* 43(2), 103-118.

International Organization for Standardization. (1998). *Ergonomic Requirements for Office Work with Visual Display Terminals (VDTs) — Part 11: Guidance on Usability* (ISO Standard No. 9241-11). Retrieved June 5, 2019, from https://www.iso.org/standard/16883.html.

Janicki, T., Cummings, J., & Healy, R. J. (2015). Incorporating a Human-Computer Interaction Course into Software Development Curriculums. *Information Systems Education Journal*, 13(3), 81.

Kumar, K., van Dissel, H. G., & Bielli, P. (1998). The Merchant of Prato-Revisited: Toward a Third Rationality of Information Systems. *MIS Quarterly,* 22(2), 199-226.

Nielsen, J. (2003). Paper Prototyping: Getting User Data before You Code. *Nielsen Norman Group.* Retrieved August 6, 2020, from https://www.nngroup.com/articles/paper-prototyping/.

Norman, D. A. (1988). *The Psychology of Everyday Things*. New York, New York: Basic Books.

Papert, S. (1991). Situating Constructionism. In S. Papert & I. Harel (Eds.), *Constructionism: Research Reports and Essays, 1985-1990* (pp. 1-11). Norwood, New Jersey: Ablex Publishing Corporation.

Park, K. (2015). PowerPoint Should be Banned. This PowerPoint Presentation Explains Why. *The Washington Post*. Retrieved June 3, 2019, from https://www.washingtonpost.com/posteverything/wp/2015/05/26/powerpoint-should-be-banned-this-powerpoint-presentation-explains-why/.

Pernice, K. (2016). UX Prototypes: Low Fidelity vs. High Fidelity. *Nielsen Norman Group*. Retrieved November 8, 2019, from http://www.nngroup.com/articles/ux-prototype-hi-lo-fidelity.

Pogue, D. (2016). 5 of the Worst User-Interface Disasters. *Scientific American*. Retrieved May 28, 2018, from https://www.scientificamerican.com/article/pogue-5-of-the-worst-user-interface-disasters/.

Rettig, M. (1994). Prototyping for Tiny Fingers. *Communications of the ACM*, 37(4), 21-27.

Sefelin, R., Tscheligi, M., & Giller, V. (2003). Paper Prototyping - What is it Good For? A Comparison of Paper- and Computer-Based Low-Fidelity Prototyping. In *CHI'03 Extended Abstracts on Human factors in Computing Systems* (pp. 778-779). Fort Lauderdale, Florida.

Snyder, C. (2001). Paper Prototyping: Sure, It's Low-Tech, But This Usability Testing Method Can Help You Sidestep Problems Before You Write Your Code. Retrieved November 9, 2019, from https://www.csee.umbc.edu/courses/undergraduate/345/spring12/mitchell/readings/paperPrototyping_Snyder.pdf.

Te'eni, D., Carey, J. M., & Zhang, P. (2005). *Human-Computer Interaction: Developing Effective Organizational Information Systems*. Hoboken, New Jersey: John Wiley & Sons.

Valacich J. S. & George J. F. (2017). *Modern Systems Analysis and Design* (8th Ed)*. Upper Saddle River, New Jersey:* Pearson Education, Inc.

Vessey, I. & Galletta, D. (1991). Cognitive Fit: An Empirical Study of Information Acquisition. *Information Systems Research,* 2(1), 63-84.

Vijayan, J. & Raju, G. (2011). A New Approach to Requirements Elicitation Using Paper Prototype. *International Journal of Advanced Science and Technology*, 28, 9-16.

Yun, C. (2007). Hanmail Paper Prototype. [Video] Retrieved July 30, 2018, from https://www.youtube.com/watch?v=GrV2SZuRPv0.

Zhang, P. & Li, N. (2005). The Intellectual Development of Human-Computer Interaction Research: A Critical Assessment of the MIS Literature (1990-2002). *Journal of the Association for Information Systems,* 6(11), 227-292.

AUTHOR BIOGRAPHIES

Michael J. Scialdone is a clinical associate professor of information and operations management in the Mays Business School at Texas A&M University. He holds a doctorate in information science and technology from Syracuse University. His research interests include the impacts of information technology on education, business, and self-organizing teams. He has been published in journals including *AIS Transactions on HCI* and *Journal of the Association for Information Science and Technology*.

Amy Connolly is an assistant professor of computer information systems and business analytics in the College of Business at James Madison University. Her doctorate is in management information systems from the University of South Florida. Her research interests include the role of social media in volunteer organizations and active learning and inclusion in information systems pedagogy. Her research has been published in journals including *European Journal of Information Systems*, *Information Systems Education Journal*, and *Informing Faculty*. She was recently awarded the Stafford Beer Medal 2019 by the Operational Research Society.

STATEMENT OF PEER REVIEW INTEGRITY

All papers published in the Journal of Information Systems Education have undergone rigorous peer review. This includes an initial editor screening and double-blind refereeing by three or more expert referees.