1. Out of Industry, Into the Classroom: UX as Proactive Academic Practice

Kate Crane
Eastern Washington University

Abstract: User experience (UX) is an established practice in industry and is taught as a necessary skill for future technical and professional communicators. However, its use in technical and professional communication (TPC) pedagogy and program development is notably absent in TPC literature. This chapter contextualizes and situates UX in TPC, documenting its evolution from user advocacy to usability, to user-centered design, and then to UX. UX is a broadly used term that, at its core, values design processes that are focused on the user experience (notably, user-centered design, human-centered design, participatory design, and design thinking). Further, UX promotes the collection of user data to understand how the user experiences products and processes in a way that promotes iterative design and assessment. UX is contextually bound, non-linear, and often not generalizable; yet, to build and assess programs and curricula, TPC needs to take a UX approach to our work to best serve our student users and our field.

Keywords: student users, user experience, pedagogy, technical and professional communication

Key Takeaways:

- User advocacy and UX is a cornerstone of all technical and professional communication work.
- Localized UX and small-scale studies best apply to pedagogical and programmatic design.
- Proactive UX work strengthens curricula and programs.
- UX and design processes (user-centered design, human-centered design, participatory design, and design thinking) are imperative for successful curricula and programs.

I began my user experience (UX) journey by accident. My personal statement to my future Ph.D. program stated that I wanted to learn how technology changed the ways students wrote and engaged with texts. I had never heard the terms "user experience," "user-centered design," or "usability." Technical and professional communication (TPC), even then, was on the periphery of my writing studies experience as I had mostly taught first-year composition up to that point. That was in 2010. Four years later, I was iteratively testing how students interact-

DOI: https://doi.org/10.37514/TPC-B.2022.1367.2.01

ed with syllabi. I was also playing with eye-tracking technology as a metric for usability studies. Ten years later, I had co-written a book on user-centered design, collaborated with colleagues on designing research projects with UX methods, collaborated to design curriculum using UX methods, and was even approached to begin this book collection on how UX methods could be applied to curriculum design. I'm not bragging; rather, upon reflection, I see how UX has permeated my everyday work as an educator, program director, researcher, and designer.

It was my syllabus study, however, that really showed me the potential of UX for curriculum and program design. In 2014, I developed a usability study to determine if student users would use a PDF syllabus or a non-linear web syllabus more effectively. I observed 21 students per syllabus mode (42 total) perform five tasks each. The syllabus usability test findings were as follows:

1. The PDF's efficiency errors were often the result of users locating incorrect information for a task.
2. Users used print navigation cues (such as headings) to stop in locations where the correct information was not available to complete a task.
3. Term confusion led to using navigation cues that did not help users locate information.
4. Users' previous experience with syllabi led to generic expectations about how they would use new syllabi (Crane, 2015).

However, the more significant conclusion that I took from this usability study was that the usability test was not generalizable beyond the specific context of this particular syllabus evaluation. What I mean by this is that once I revised the syllabi to be more usable, or a new population of students was tested, or I moved to a new institution with different policies and procedures, or new technologies were better suited to deliver syllabi, the test conducted in 2014 would be insignificant beyond a historical perspective of usability using two texts and technologies in 2014. That is not to say that usability testing, or this study, is insignificant; rather, it's an acknowledgement of usability testing's limitations. Usability evaluates products in a very specific context that changes based on the stage of a design process, the users tested, and the environment where the product is being tested. Usability, therefore, should only be used iteratively to understand how a design works for users at any given time or environment. Second to this conclusion was my understanding that course documents are (or should be) student-user centered and that it is an instructor's responsibility as an information designer to understand the student-user experience while using these documents. In this chapter, I use the syllabus usability study to illustrate the complexity of curricular materials and the importance of UX as we approach designing these materials for students.

Now that I am a faculty member and program director, I tend to see most tasks through the lens of UX. Claire Lauer and Eva Brumberger (2016) analyzed job postings for technical communicators and found that "the core competencies" of user experience (UX) and information design were placed directly in the realm

of technical communicators' work and skill sets (p. 262). Historically, TPC as a discipline has centered its identity around user advocacy, which evolved from usability to UX. Unfortunately, this user focus has mostly been discussed in TPC research as an application in industry rather than pedagogy or program design.

While user research has focused on industry applications, academic programs have evolved in UX education at institutions, such as the University of Washington's Human-Centered Design and Engineering, Michigan State's Experience Architecture, and the Milwaukee School of Engineering's User Experience degrees. Nevertheless, UX as a philosophical principle and methodological tool for the work we do in designing TPC programs and curriculum has been absent. Yet this work is imperative to designing innovative programs that attract students, prepare them for the field, and adapt to students' ever-changing needs. In other words, TPC program and curriculum designers ought to model the work their students will be expected to do post graduation. Offering UX instruction is different than programs developing curricula through UX. Based on the lack of published research on UX in program development, I claim that TPC program designers ought to demonstrate UX thinking while designing and assessing their programs and courses; this is an argument that the collection makes cumulatively throughout each chapter.

This collection derives its focus from this assumption: UX is a vital theory and principle that should guide TPC academic work. *To frame this assumption, this chapter contextualizes UX in TPC, and, in particular, attempts to tease out the differences between various design and research methods that are related, yet often conflated.* Further, it shows the interrelated nature of UX design methods and the various user research methods that inform UX designed products. Although by no means exhaustive in its discussion of design and research methods, this chapter attempts to show the hierarchy of UX and its relationship to design and research methods. At the same time, using illustrative examples from my own syllabus research, I discuss the various opportunities and challenges of UX work.

From Usability to User Experience in TPC

To understand TPC's emphasis on UX, a brief history of its evolution is necessary. UX is a fairly new concept in technical and professional communication studies. However, its emergence in design fields, especially in TPC, follows a longer path of TPC professionals trying to situate users in such a way that they are part of the design process for any system.

Before UX, and even usability studies, existed, designers focused on creating products through system-centered design (also referred to as the waterfall method). System-centered design focused more on the needs of the system to function as the designers intended. The problem with this approach is that systems, even well-built systems, are not always usable for the people those systems were designed for. Usability evaluation (through testing or heuristics) was used to ensure that end-users (or those who would use the designed product) could actually use

the product to complete intended tasks. However, usability was applied at the end of a design process once a product was built and nearly ready for distribution (i.e., being implemented or sold). The problem with this model is that users were only part of the design process once the design was complete, and their usability data only applied when tweaking final designs before being sent to the workplace, marketplace, or classroom.

Usability first and foremost references how usable a product is for completing a task. The International Organization for Standardization (ISO, 2018) defines usability as the "extent to which a system, product, or service can be used by specified users to achieve specified goals with effectiveness, efficiency, and satisfaction" (sec. 3.1.1). Usability testing was often (and sometimes still is) conducted in an experimental fashion to locate bugs in the system or to identify other major concerns that would create problems for the user. Usability evaluations focused on narrowly defined tasks that users would complete while using the product. These experiments tested for usability, but their mission was to ensure products were used for the tasks designers anticipated and to ensure those tasks were completed in the simplest way. This focus did not inform designers how users were interacting with the product in their real-world environment (Mirel, 1991; Redish 1995). Usability, usually under human-computer interaction (HCI) fields, limited itself to traditional experimental methods that focused on comparing interfaces and users' interactions to "long lists of guidelines of good practice" (Dumas, 2007, p. 55). This experimental method limited the amount of data specific to users' environments; users were tested in labs under strict testing protocols rather than in their own working environments or while performing tasks pertaining to their actual work. Joe Dumas (1989) argued for usability professionals to act as change agents "to have a long-range impact on the way an organization develops products" (p. 37). He advocated for involving usability early in the design process to understand how users interact with products.

As Dumas (1989) advocated, usability evolved. The usability profession became a movement away from a systems-based approach that used experimental methods to evaluate the usability of products. Usability professionals' new approach ultimately led to a focus on smaller-scale studies that provided an easier, more cost-effective method for incorporating usability iteratively in a product's design process (Barnum, 2019; Krug, 2014; Nielsen 1993). Further, as Janice Redish (1995) notes, usability took on a more integral role as computing became more common, users began using computers for various tasks, and new features provided more opportunity but also more usability problems. TPC, in particular, brought new methods of research beyond quantitative experimental research and a new focus on user advocacy.

Though much of the work in usability studies has focused on software and hardware, TPC professionals applied usability methods to test documentation long before usability testing helped in the evaluation of computers and software (Redish, 2010). TPC practitioners saw themselves as user advocates, and usability testing was one method TPC practitioners used to ensure that documentation

could effectively assist its end users in completing their tasks (Redish & Barnum, 2011). Further, Robert Johnson (1998) discussed the importance of listening to users, whom he calls "the better judges" of technology use (p. 11). As user advocates (Redish, 2010), TPC's role was to ensure that the user was at the center of designing content or products.

In 1993, Jakob Nielsen published *Usability Engineering*, in which he claimed that usability ought to be used throughout the design process by performing small tests iteratively between design stages. This strategy emphasized that, although usability professionals would only test a handful of users (he suggested five per iteration), these tests would inform the overall usability of the product for users. Using Nielsen's model, five users per iteration could potentially lead to dozens of users throughout the design's lifecycle. Further, this strategy suggests that users are agents within the design of the system, rather than novices experiencing the system only as end users. This aggregate user data and the design improvements upon incorporating user observations would ensure that the product's usability was well tested before the final distribution of the design.

Nielsen also proposed five dimensions of usability: learnability, efficiency, errors, memorability, and satisfaction. Building on Nielsen, Whitney Quesenbery (2003) provides a model for usability, the five E's: effective, efficient, engaging, error tolerant, and easy to learn. Table 1.1 outlines these two models.

Table 1.1. Nielsen's and Quesenbery's Usability Characteristics

Nielsen's Five Usability Attributes (1993)	**Quesenbery's Five Usability Dimensions (2003)**
Efficiency: The system should be efficient to use, so that once the user has learned the system, a high level of productivity is possible (p. 26).	Efficient: The speed (with accuracy) with which users can complete their tasks (p. 84).
Memorability: The system should be easy to remember, so that the casual user is able to return to the system after some period of not having used it, without having to learn everything all over again (p. 26).	Effective: The completeness and accuracy with which users achieved their goals (p. 83).
Satisfaction: The system should be pleasant to use, so that users are subjectively satisfied when using it; they like it (p. 26).	Engaging: The degree to which the tone and style of the interface makes the product pleasant or satisfying to use (p. 86).
Errors: The system should have a low error rate so that users make few errors during the use of the system, and so that if they do make errors, they can easily recover from them. Further, catastrophic errors must not occur (p. 26).	Error Tolerant: How well the design prevents errors, or helps with recovery from those that do occur (p. 87).
Learnability: The system should be easy to learn so that the user can rapidly start getting some work done with the system (p. 26).	Easy to Learn: How well the product supports both initial orientation and deepening understanding of its capabilities (p. 88).

Four of Nielsen's and Quesenbery's components for usability are very similar. For instance, users need to be able to complete tasks efficiently, learn a system in a reasonable amount of time, and recover from errors when they are made. Perhaps the first glimpse of user experience in usability was Nielsen's "satisfaction" and Quesenbery's "engaging" components. Both of these components indicate that users should feel good about using a product. These dimensions acknowledge that a workable product does not determine adoptability. Rather, emotional reactions to the product are also a factor for how users determine what product is best for completing a task (consider, for instance, how many people switch between PC and Mac platforms or iPhones and Androids). Users choose their preferences based on logic and emotion. The only differences in usability components between these frameworks are Nielsen's "memorability" dimension (the ability to remember the system enough to easily complete future tasks) and Quesenbery's "effective" aspect (which indicates that the users could successfully complete their goal). Both of these usability qualities deserve a place in our consideration of how usable a product or process should be. But, more importantly than comparing the differences in models, using models such as Nielsen's and Quesenbery's provided us with frameworks to understand data in relation to users' experiences with a system.

In the early 2000s, technical and professional communicators were still concerned with the usability of content, but they also took a turn in thinking about how their role as user advocates needed to begin even earlier and, on a more basic level, than what usability affords. In usability, there must be a product to test. Thus, the more dynamic the product, the more complex the problems that would arise. Barbara Mirel (2002) describes this problem as a lack of understanding usability in terms of "users' complex work-in-context" and notes that "if usefulness is to take center stage, a shift is needed in analyzing and designing for complex tasks" (p. 167). In other words, not only should we be concerned with how well users can complete tasks, but researchers should not assume that their (or a designer's) preconceived ideas about users' work is a fair representation of the complexity of users' work beyond a usability lab (or any testing situation).

Breaking down usability further, Michael Albers (2004) delineates three types of usability that are dependent on the product's level of complexity: simple, complicated, and complex. Simple usability employs a single path to complete a task, provides predefined answers to questions, predicts systematic changes and effects, and exists within a closed system. An example of simple usability is having students log in to a learning management system (LMS). My institution uses Canvas, and students only have one pathway for logging in to the LMS. First, they need to use a link (or correct URL) to navigate to Canvas. Students are then directed to the institution's single sign-on page, where they input their username and password. If their credentials are accepted, the Canvas dashboard appears. Thus, there is one pathway for signing in, and students know they have completed the task if they see their Canvas dashboard after they sign in.

Complicated usability is still part of a closed system, has predefined answers to questions, and a "description can be given," but there are multiple pathways for completing a task (Albers, 2004, p. 16). To distinguish between simple and complicated, consider how students use an LMS to turn in an assignment. There is one task to complete—submit the assignment. However, students may have multiple pathways to turn in that assignment. For instance, in Canvas, students have at least three ways to find the assignment page and submit their assignment files: 1) they can click on the assignment in their to-do list, which takes them to the assignment page; 2) they can go to the "Assignment" link on the course's Canvas menu; or 3) they can find the assignment link in the class modules. They can choose any of these options to turn in the assignment and, if navigated correctly, they should be able to submit the assignment correctly, and I can view their assignment in my grading queue. Thus, the user has more options for how to complete the task.

Complex usability, the final of the three types, does not have a clear description of its tasks, solutions, or questions. It is "part of an open system," and tasks may require leaving a system to be completed (Albers, 2004, p. 16). A complex system takes away more of the designer's control, and the user's actions are more unpredictable. The work done in program and curriculum design falls more directly under the realm of complex usability. When studying syllabus use, for example, it was impossible for me to predict how, where, when, and for what purpose students would use a syllabus. Student users could stay in Adobe Reader to complete tasks using the PDF syllabus, but they could also click on a hyperlink that led to a university code of conduct policy with additional navigation options. Thus, creating this system may seem simple (it's on the syllabus, right?), but the syllabus document relies not only on instructor design but also on the design of university webpages to inform its construction. As TPC instructors and program designers consider how student users will use course or program materials, we must all consider how the materials we create work in conjunction with other factors such as websites, learning management systems, or even email platforms students use.

As these descriptions illustrate, Mirel and Albers forecasted the shift to UX before they began using the language, but this move further distanced usability (as a concept and evaluation method) as the sole attribute needed to advocate for users. Relying on usability could be seen as one of the major flaws in my syllabus design. Though I designed a syllabus based on program and university needs, I didn't really know what students needed or how they used the syllabus. I had been encouraged, for years at many different institutions, to include as much information in my syllabus as possible; doing so would make it bulletproof and, therefore, effective. This, of course, was not the case. The syllabus worked for me and administrators, but not for students. A usability test was a start in understanding the nuances of student use but was not enough for understanding the students' user experience while trying to find and use information—such as the

attendance policy or major assignment descriptions—on the syllabus. In other words, the priorities of instructors, programs, and the institution at large did not represent the priorities of students.

Users need to be considered before designs are drawn, programmed, or outlined. Michael Salvo (2001) noted that, though a popular method, usability focuses on the end product rather than on collaborative design where users are participants in design instead of just the observed. Since this time, models such as user-centered design, human-centered design, participatory design, and design thinking have influenced TPC's work. The need for a clear distinction between researching users' success with tasks and the need to design the experience of users became the exigency needed for user experience to emerge as an overarching theory where design processes and user research methods (including usability testing) contribute to understanding a users' holistic experience.

User Experience for TPC

After analyzing my syllabus usability test's results, I learned that usability testing alone could not answer the questions I posed for my study. The syllabus and the students who use it are part of a complex academic system with multiple factors, stakeholders, tasks, environments, and functions. Thus, looking at usability alone, though a good starting point, led to more questions than it could answer. In essence, syllabus designers need a UX approach to understand how students interact with a syllabus within the context of their myriad uses. This is one example of why TPC instructors and program designers need to understand how UX functions within an interconnected web of design processes (user-centered design [UCD], human-centered design [HCD], participatory design, and design thinking) and research methods (observation, self-reporting, affinity diagramming, usability testing, etc.).

User experience is a theory and practice that emphasizes the need for functional products that integrate the users' needs and experiences. The ISO (2018) defines user experience as "users' perceptions and responses that result from the use and/or anticipated use of a system, product, or service" (sec. 3.2.3). These perceptions may include "users' emotions, beliefs, preferences, perceptions, comfort, behaviors, and accomplishments that occur before, during, and after use" (sec. 3.2.3). Thus, user experience goes beyond asking *how well a user can complete a task* to *how the user feels as they prepare and actively interact with a product*.

Given its historical development, user experience is an ambiguous term often confused or conflated with various research methods and theories. Don Norman and Jakob Nielsen (n.d.) state that "'User Experience' encompasses all aspects of the end-user's interaction with the company, its services, and its products" (para. 1). For this reason, user experience focuses on a more holistic understanding of the user, the user's context of use, and the user's experience as they interact with a given product. It is a theory or philosophy, supported by design processes that put the human user in the center of design processes, whether these processes are labeled as user-centered design (UCD), human-centered design (HCD),

participatory design, or design thinking; these design processes are then enacted through four iterative stages:

1. collecting information about human users (those most likely to use products upon design completion),
2. designing prototypes that can be used by these human users to collect additional data about their use,
3. redesigning products in response to the first two methods, and
4. testing and retesting products during and after distribution.

Whichever stage is enacted, UX researchers should be most concerned with their users' feelings about and interactions with the product, which are gauged through user research, collaborative activities or design, and even usability testing. But user experience is primarily concerned with how a product or process is built and maintained to support all its users. In other words, designers study and incorporate data about users' experiences with the design to continue adapting the product or process to best fit users' productive and emotional needs.

Another emerging theory related to UX is *experience architecture* (XA). XA, first used in customer relations, was introduced to technical and professional communication by Liza Potts. Discussing XA and its focus on collaborative teams, Potts (2014) explains that XA "focuses on architecting the end-to-end experiences of the participants" (p. 3). Further, XA focuses on how participants interact with systems as part of a larger ecosystem. In other words, participants do not work in a vacuum; they use systems that are influenced by multiple factors, including factors that are not work related. XA focuses on how the human experience is affected (for better or worse) by using technological systems (Potts & Salvo, 2017).

XA builds on the principles of UX, and its name certainly focuses on the idea of building (or architecting) the experience for and with participants. UX ought to take note of this metaphor, not only because it is a valuable theory from which to draw new, innovative ideas but also because UX designers are building experiences for other people. Whether they are navigating new software, an interactive museum exhibit, or an academic program, designers of these products have created the map for which users can navigate through their experience with the system. Thus, when undertaking a new design endeavor, designers should never underestimate the impact their building has on the people that use their product. Further, like UX, "experience architects need to hear from users, participants, researchers, designers, developers, business analysts and various other stakeholders" (Potts, 2014, p. 4).

XA considers UX as a design method that helps support the XA mission. However, in this collection, we use UX in a similar way to how Potts and Salvo (2014) use XA: as a theory and practice that aims to understand the holistic experience of users and design according to users' needs and feelings. I acknowledge and appreciate the use of different terms as TPC, user needs and interaction, and marketplaces

evolve; however, I place particular value in work that supports users as they navigate through various resources (electronic technologies, print resources, and organizational processes) more than claiming a new name for this work. We use UX in this collection to acknowledge the history and emphasis of user advocacy in TPC. That emphasis has paved the way for scholars and practitioners to continue pushing the boundaries for how user advocacy is enacted as new problems arise in the development of technologies, processes, and services. In other words, UX and XA can coexist as related umbrella concepts which design theories and research methods fall under as long as they use design processes and research methods that center, support, and include users in constructing new products.

UX and XA are not tidy or simple theories to enact. Rather, they are non-linear, highly contextual, and often not generalizable. Their use throughout the design process often reveals new UX challenges through each iteration and over the entire lifecycle of the product. Whereas a design iteration may end, UX work never does. Though UX professionals may begin thinking about their product from a UX process before any prototype is created, this is not always the case. Sometimes, UX does not become a conscious act until designers are ready for their idea to be seen or used. Alternatively, designers may inherit a product or process that needs revamping to better serve users. This is one reason why evaluating the usefulness of a product may be the first step in researching user experience.

Further, UX research, whether applied in industry or the academy, has its limitations. One of these limitations is reaching a level of generalizability. Recent articles about research in TPC have stressed the importance of conducting and reporting on research that can be replicated to test the researchers' findings (Lauer, 2016; Meloncon & St.Amant, 2019). UX research is highly contextual, and solutions are often quickly deployed to create a better user experience. Thus, conducting UX research means using very specific methodologies that are appropriate for the local context of the users' needs, rather than aiming for generalizability. This does not mean that UX research eschews *empirical research*, which, according to Mary Sue MacNealy, "describes and/or measures observable phenomena in a systematic way that has been planned in advance of the observation" (as cited in Meloncon & St.Amant, 2019, p. 130). The value of systematic research in UX is clearly seen in the design and research models discussed here and in other TPC literature. However, localized research for curriculum, community projects, or industry interactions with customers may not be as easily replicated based on the specific context they inhabit.

The specific, localized context is one that UX scholars have already noted. The small-scale studies such as guerilla usability or discount usability are highly valued in agile processes, and they are valuable in these localized pedagogical contexts as well, as we see in this volume's chapters. UX studies, according to Brian Still (2010), "allow for experimentation, for proactive research that enables practitioners to stay a step ahead" (p. 104). Carol Barnum (2019) further advocates for the small-scale study in "development scenarios" that work "particularly well

in Agile development methodologies" (p. 2). She emphasizes that "the goal of UX research is to identify or uncover *valid* findings, fix them, and iterate the process to discover more" (p. 3). Thus, the methods necessary to create rigorous UX research certainly fall within the parameters of TPC's empirical research focus. Moreover, the limited number of participants helps create solutions to UX problems by prioritizing iterative methods without depleting resources.

Though Still and Barnum both focus on small-scale studies for industry, this volume's authors claim that these small-scale studies are equally important to programmatic and curriculum design. TPC program designers can not only use scholar/instructor expertise or industry expertise to answer their questions. Another important stakeholder needs to be considered: students. For instance, one decision we all must come to when creating a program is how many course credits we should include in our major/minor/certificate. Will students want to spend one quarter of their college credits to get a degree in a TPC program, or will they feel this does not give them enough experience with TPC curricula? If we create a major that is one half of students' college credits, will they feel the content being taught is too repetitive between classes? Will they feel rushed to finish their degree on time? To answer these questions, we need to understand our student population, based on what they say their priorities and goals are. In this case, the UX process ought to start by talking to all stakeholders, including students. Then, based on the data collected, program designers can weigh the needs of all stakeholders to best design a program that ultimately can be marketed to students (see Cargile Cook and Masters-Wheeler & Fillenwarth in this collection).

UX Design Processes

UX is often conflated with other design processes because UX design was (and still is) used in place of user-centered design and human-centered design. This is a mistake. UX is the umbrella theory and principle—a methodology, if you will—that designers and researchers should adopt. But there are many means for achieving the UX end goal. *User-centered design* (UCD), *human-centered design* (HCD), *participatory design*, and *design thinking* can all be used to support a UX project. These four design methods have more in common than not. This is a strength for UX as a concept, practice, and field. UX is supported by design methods that integrate user research and feedback. Instead of designing for users, UX-focused designers build and assess products and processes with users and understand that each new user, new product or process, or new environment irrevocably changes the user-product system. Designing for UX is an act of humility and empathy. Choosing the best design process for a strong UX product is part of architecting that product. It is with this in mind that I describe four design methods that lend themselves to UX design. I limited my discussion to these four design processes because they have been the most widely discussed in TPC literature.

UCD focuses on designing products and processes in a way that, from planning to distribution, places users at the center of the design process. The goal of centering users is making sure that designers understand the system requirements needed for users to work effectively (Baek et al., 2008; Cosgrove, 2018; Still & Crane, 2016). UCD keeps users at the center of the design process, encouraging an iterative design model that incorporates user data and feedback throughout the design process.

However, some scholars believe we should move away from labeling people "users" as this term can dehumanize those who work with the systems designers create. For instance, Don Norman advocated for UCD in the 1980s, but by the time he released the revised and expanded version of *The Design of Everyday Things* in 2013, he changed his focus to HCD. He defines HCD as "an approach that puts human needs, capabilities, and behaviors first, then designs to accommodate those needs, capabilities, and ways of behaving" (2013, p. 8). One of the main differences between UCD and HCD, is the shift in nomenclature from "users" to "humans." This is not to minimize either process; rather, it acknowledges that some UX scholars and designers feel that the term "users" is not the best way to refer to people. The use of this term has been questioned for its perceived focus on efficiency (Opel & Rhodes, 2018). However, "human" clearly centers the work of HCD as a humane endeavor—one that ultimately is rooted in humanity and working for and with people while designing. Mark Zachry and Jan Spyridakis (2016) explain HCD as a perspective that "acknowledges the role of humans in actively constructing artifacts" (p. 394). Emma Rose (2016) furthers this distinction, stating that "human-centered design should look more broadly and provide a way to consider how design can support or constrain the needs of people whose lives are impacted by both the systems and policies that are created by a more digitized world" (p. 428). Thus, HCD emphasizes the important role designers play in creating systems that influence how people live, their ability to interact with and within these systems, and the consequences when designs do not enable people to interface with these systems effectively.

We all have our own guiding principles for choosing language, and that does not mean one method, UCD or HCD, is more or less valuable. In fact, they mimic each other in many ways. TPC literature often uses UCD and HCD interchangeably (sometimes even acknowledging the choice to use one or the other terms). What is important to note is that they each follow a process that begins with understanding users/humans and consulting with them throughout the lifecycle of the design (See Figure 1.1).

Participatory design, as compared to UCD and HCD, focuses on a more democratic process for creating new products (whether for the workplace or commercial purposes). Used as a design process, participatory design focuses on designing *with* people, rather than designing *for* people. They are, as Michael Muller (2009) explains, "full participants in activities" (p. 166). Participants become members of the design team rather than mere observers or subjects for data collection.

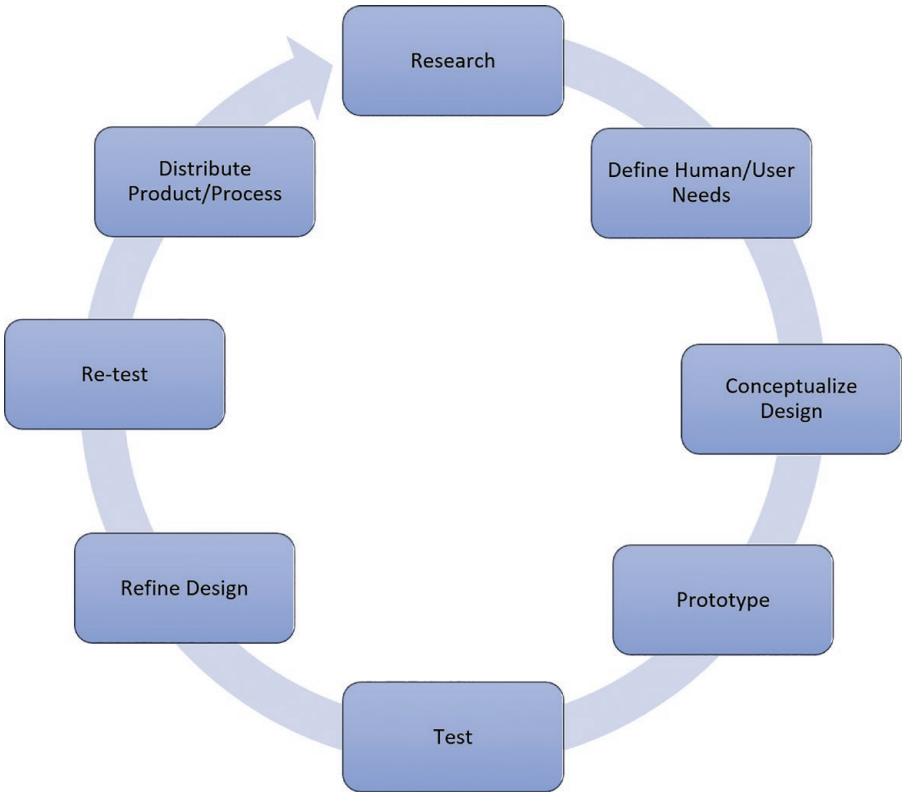

Figure 1.1. UCD/HCD process.

As seen in Figure 1.2, participatory design is about co-production rather than one designer making decisions in isolation and asking for user feedback at specific points in the process. The user is a *co-designer*, one that participates throughout the design. Co-designers are actively involved in the planning, research, design, delivery, and evaluation of a product (e.g., a syllabus) or process (e.g., developing a mentoring program for TPC majors—see Lee-Ann Kastman Breuch, Ann Hill Duin, and Emily Gresbrink's chapter in this collection). The participatory design process is as concerned with building relationships as it is with delivering appropriate, usable products to its users. This emphasis on a design's relationship with its participants is at the heart of participatory design. It is democratic in the way it approaches the participant as an expert and co-designer.

Design thinking, a relatively new theory introduced to TPC, distinguishes itself as a way of problem-solving that emphasizes empathizing with users, defining the problem, ideating solutions, prototyping designs, and testing the designs' effectiveness (Dam & Siang, n.d.; Pope-Ruark et al., 2019).

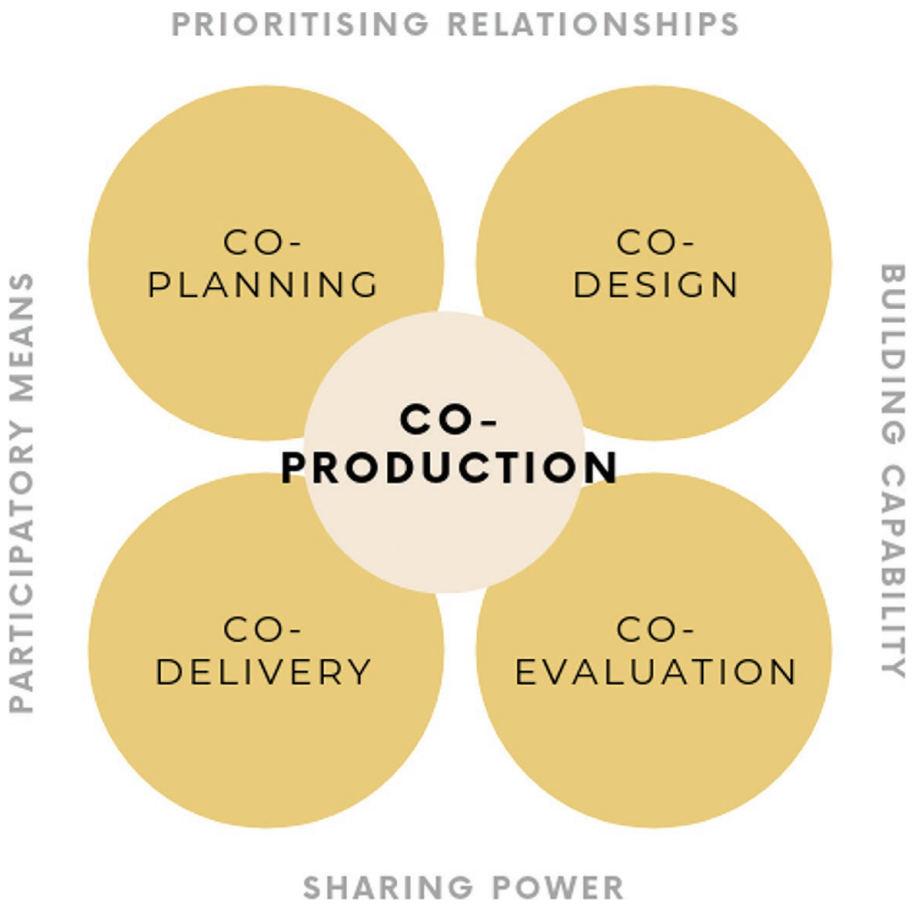

Figure 1.2. Co-production by Kelly Ann McKercher (as adapted by Stephens, 2019).

Design thinking's values (see Figure 1.3) mimic the processes and values of user-centered, human-centered, and participatory design. As Tharon Howard states, "Many UXers see design thinking as just a repackaging of what they've always done" (as cited in Pope-Ruark, 2019, p. 444). However, Howard and Melonçon acknowledge design theories' commitment to UX and TPC's user advocacy mission (Pope-Ruark, 2019).

These design methods provide a guide for how to develop new products and processes; more important, the design methods all include stages where user feedback is necessary. In fact, not one of these design methods is described as a linear process because when user research or feedback is collected and analyzed, designers may have to revert back to an earlier stage in the design. For instance, my syllabi were complete, but my research showed flaws in the design. Students wanted relevant information (such as grading and attendance policies) closer to the beginning of the syllabus, and my use of academic discourse made navigating to the correct informa-

tion confusing for students. Reverting back to collecting user research and designing new prototypes with student users may be the best, next iteration for the syllabus study. When our designs don't work out, we need to go back to user research.

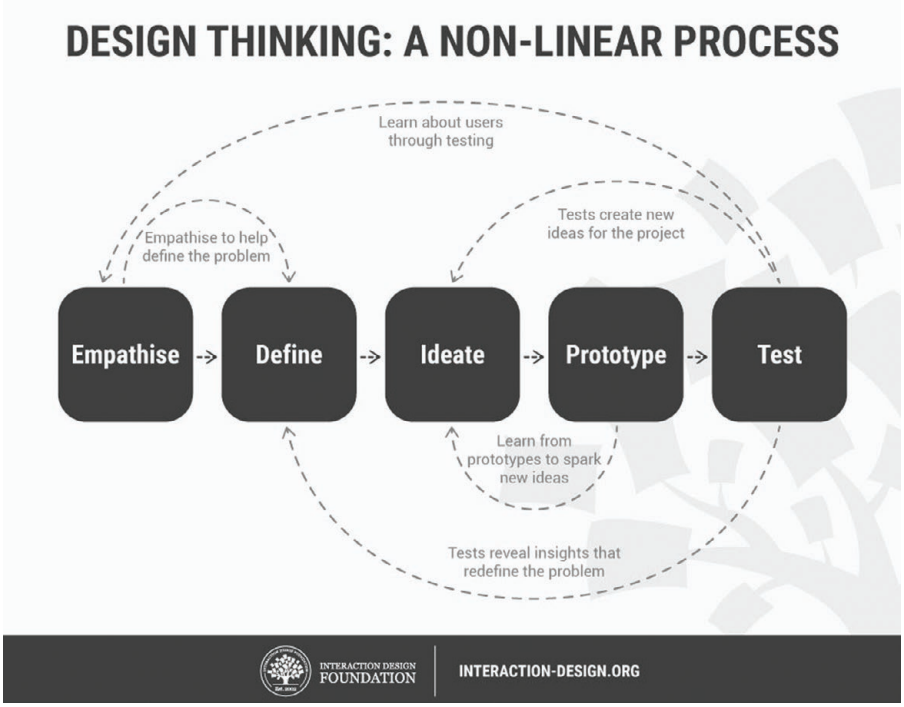

Figure 1.3. Design Thinking Process by Teo Yu Siang CC BY-NC-SA 3.0 (Dam & Siang, n.d).

To demonstrate how these design methods work within UX and XA, Table 1.2 summarizes the relationship between these theories, the design processes, and the methods that might be used in these processes. It illustrates a hierarchy of methods where design processes/methods support developing a positive user experience; the research methods inform designers throughout the design process. Worth noting, however, is that user research methods continue to evolve and expand; those methods listed in Table 1.2 are far from a complete list.

Table 1.2. UX/XA Processes and Methods

Theory	User experience and experience architecture			
Design Methods & Processes	User-centered design	Human-centered design	Participatory design	Design thinking
User Research Methods	Contextual inquiry	Self-reporting	Collaborative design studio	
	Affinity diagramming	Prototype feedback	Usability testing	

UX Research Methods

As Table 1.2 illustrates, UX research methods support the design process. These methods can be used in various design stages. The goal of the design method is to determine the type of data collection needed. If the design process is starting from the beginning of a new design, the first step is to understand users. The two research methods from Table 1.2 that align with this step are contextual inquiry and self-reporting.

Contextual inquiry is a method where researchers observe users working in their natural environment to understand how they complete tasks (Beyer & Holtzblatt, 1999; Still & Crane, 2016). Contextual inquiry (also called a *site visit*) includes time for the researcher to consult with the user being observed to ask questions about how tasks were completed, get clarification about why a user chose one action over another while completing their task(s), and, more generally, discuss how the user felt while completing the task. Contextual inquiry is one of the best ways to understand a user's context of use and how a user's environment and tools affect their work.

In addition to contextual inquiry, *self-reporting methods* (such as surveys, interviews, and focus groups) can help researchers understand the behaviors of their users toward a task or product. Self-reporting can collect a broad spectrum of user data (in the case of surveys) or data from a narrow user group that is most likely to be impacted by new designs (this is where interviews are very relevant). Although I had already studied syllabi and had a syllabus model ready to test, I still administered a survey to student users in the specific program I was studying (Crane, 2015). Doing this not only gave me important demographic data to determine who my representative user was, but it also provided me with information about what these students prioritized when using syllabi. I learned, for instance, that students were more interested in the information about what they had to do (such as assignments, absence policies, and grading policies) than they were in my teaching philosophy or university boiler plate language. I was then able to use this data to determine which tasks were "representative" for my user population. This preliminary data was imperative to understanding not only who my student users were but, more importantly, what their motivation and priorities were for using a syllabus.

When researchers have enough information about users that they understand user needs, priorities, and beliefs, they proceed to the next design stage: conceptualizing or ideating. Here is one place where the design process may affect methodological choices. For some designers, the data collected to understand the user would be enough to begin ideating, building, and prototyping a product. However, other design processes, such as participatory design, would advocate for ideating with users via a collaborative design studio or affinity diagramming exercise.

A *collaborative design studio* pairs designers and users together to collaborate on possible designs to address a problem. This collaboration ought to focus on

making design a more democratic process and, ideally, lead to clear design directions, such as a list of features that are necessary to address users' needs or a rapid prototype (such as a paper prototype) of the new product (see Elizarova et al., 2017 for more participatory design methods). *Affinity diagramming* is another tool that involves designers collaborating with users to understand what attitudes and/or values are shared amongst the user and designer group. Often, this work is done on a whiteboard or by using sticky notes. For instance, if I were redesigning syllabi with students, I may use affinity diagramming to discover the most valuable information for students using the syllabus. In smaller groups, students would be asked, "What are the most important elements of the syllabus for you?" and then I would ask them to work through the following process:

1. Each person writes an attitude, value, or idea on a sticky note (one idea per sticky note).
2. All participants' sticky notes are examined for patterns.
3. Patterns are identified by clustering sticky notes.
4. Patterns are compared between groups.
5. All students discuss how/if these patterns should become part of the course syllabus.

Affinity diagramming provides all participants the opportunity to make their values and attitudes known without succumbing to group thinking. This practice can also lead to low-fidelity co-designed prototypes where users can construct their own syllabus, in this case, using the values discovered from affinity diagramming and program and university syllabus requirements to create their own student-as-user-centered syllabus. There are many other collaborative research methods—these two represent a couple methods that I have worked with. The point of this research grouping is that it invites users/participants/students to be part of the design process instead of being seen only as end users. This is also an exercise that introduces students to work with/as participants, and teaches instructors how to practice participatory design in the classroom. (See Beau Pihlaja's chapter in this collection for more examples of class co-design activities and Luke Thominet's chapter for information on affinity diagramming to create program student learning outcomes.)

The final two user research methods I will address here are opportunities for prototype feedback and usability testing. In many ways, prototype feedback and usability testing try to achieve similar ends: How will users engage with the product? *Prototyping feedback* may include asking users to complete a task using a prototype to gauge how well the prototype allows users to accomplish their goals, asking users to create a journey map as they are working with the prototype to understand how the users navigate a product and how they feel while doing so, or asking users to help construct or improve a prototype while they are working through a problem. Prototypes, especially those that are low-fidelity (using less technical means for construction, such as paper prototypes), provide a low-stakes

opportunity to see a design used before more resources (such as money and time) are spent on perfecting a product.

I leave *usability testing* as the last method of discussion not because I believe usability testing should only be executed at the end of a design but because it is one of the most time-intensive methods. Usability testing involves researchers evaluating a product by observing representative users completing representative tasks. Researchers collect various quantitative data (time on task, types of errors, number of inputs—such as mouse clicks or keyboard entries—and dwell time between inputs) and qualitative data (observations, pre- and post-test surveys or interview, etc.). Examining these data together help researchers to decide how well the user can work with the product. Whereas many usability tests are still facilitated in usability labs, more usability testing options have emerged, such as remote testing using videoconferencing software (e.g., Zoom or other usability-specific tools), mobile usability equipment that can be used in users' environments, and even testing that can be done by outside administrators who then share the data with researchers (e.g., UserTesting).

What is important to understand about all of these methods is that they can be used in different ways than what I have presented above. Ultimately, we use research methods to help answer questions wherever we are in the design process. It is possible to have a complete, finished product that you want to evaluate by conducting a usability test first. This was the case of my syllabus test. My design was already complete and had been used for a long time. However, I was interested in understanding how well this design worked for students. Starting with usability testing was an important step in understanding the user experience of students—more so than understanding the deficiencies of my syllabus. It would be just as appropriate to observe students (through contextual inquiry) to see how, or if, they use a syllabus differently in a natural environment rather than a testing situation.

UX and This Collection

This collection emphasizes the importance of UX not only as an area of research in TPC, but as a necessary practice for our work as curriculum and program designers. Though the field acknowledges the importance of UX in industry, the work TPC instructors and program administrators perform while designing courses and programs (and the artifacts associated with each) is not acknowledged as UX work. It should be. This UX work is some of the most important work we do for our field because future UX and TPC professionals interact directly with this work. Therefore, UX needs to be part of our arsenal for conducting innovative academic work.

As UX professionals, TPC instructors and program administrators must choose from the UX toolkit as they create, assess, and revise their work. Again, I use Still's (2010) emphasis on using small-scale studies to be proactive, and TPC,

as a discipline, ought to focus on being proactive in research and pedagogy. Being proactive means starting with small, localized studies that provide designers with enough information to start building, retrofitting, or rebuilding new courses and programs. Being proactive means collecting data to understand how students succeed with the artifacts or processes we build for them and how those artifacts or processes hinder their work. Being proactive means understanding students' ecosystems, which our classes and programs are only a small portion of, to design programs that work within those ecosystems. To be proactive, we need to work with the resources and participants that we have and strive to create better student experiences.

This collection claims, promotes, and demonstrates how UX is an academic practice in myriad ways. The authors of the following chapters describe various impetuses for employing UX work to answer questions about teaching and program development. Authors chose diverse methods to collect and analyze data to understand student users. Though authors may use similar methods, their reasons for using these methods are unique and specific to their particular research and design goals. They demonstrate how to use the small-scale study in ways that produce rigorous, yet localized data to continue iterating their course and program designs. These studies further demonstrate the versatility of UX, the design methods that support UX as a practice, and the research methods used to collect data. This collection is a starting point for discussing a UX pedagogy that we hope will continue far beyond the work we have done here.

This chapter is only the first of a two-part introduction. In Chapter 2, Kelli Cargile Cook and I continue to situate UX as an academic and pedagogical practice, one that fills an important role in TPC's mission to be student-user advocates. In doing so, we argue that UX methods help us go beyond anecdotes to inform classroom and program administration practices. Further, we provide several journey maps illustrating how this collection can be read. This is our humble attempt at making UX a proactive academic practice.

References

Albers, M. J. (2004). *Communication in complex information: User goals and information needs for dynamic web information*. Lawrence Erlbaum.

Baek, E., Cagiltay, K., Boling, E., & Frick, T. (2008). User-centered design and development. In J. M. Spector, M. D. Merrill, J. van Merriënboer, & M. P. Driscoll (Eds.), *Handbook of research on educational communication and technology* (3rd ed., pp. 659-670). Lawrence Erlbaum.

Barnum, C. (2019). The state of UX research. *Journal of Usability Studies*, *15*(1), 1-7.

Beyer, H., & Holtzblatt, K. (1999). Contextual design. *Interactions*, *6*(1), 32-42. https://dl.acm.org/doi/10.1145/291224.291229

Cosgrove, S. (2018). Exploring usability and user-centered design through emergency management websites: Advocating responsive web design. *Communication Design Quarterly*, *6*(2), 93-102.

Crane, K. (2015). *The usability of the course syllabus: Testing syllabi modality and comprehension* [Doctoral dissertation, Texas Tech University]. Texas Tech University Libraries. https://ttu-ir.tdl.org/handle/2346/64572

Dam, R., & Siang, T. (n.d.). *5 stages in the design thinking process*. Interaction Design Foundation. https://www.interaction-design.org/literature/article/5-stages-in-the-design-thinking-process

Dumas, J. S. (1989). Stimulating change through usability testing. *SIGCHI Bulletin*, *21*(1), 37-44.

Dumas, J. S. (2007). The great leap forward: The birth of the usability profession (1988-1993). *Journal of Usability Studies*, *2*(2), 54-60.

Elizarova, O., Briselli, J., & Dowd, K. (2017, December). Participatory design in practice. *UX Magazine*. https://uxmag.com/articles/participatory-design-in-practice

International Organization for Standardization. (2018). *Ergonomics of human-system interaction* (ISO 9241-11:2018). https://www.iso.org/standard/63500.html#:~:text=ISO%209241-11

Johnson, R. R. (1998). *User-centered technology: A rhetorical theory for computers and other mundane artifacts*. SUNY Press.

Krug, S. (2014). *Don't make me think revisited* (3rd ed). New Riders.

Lauer, C. (2016). Re-considering research: Why we need to adopt a mixed-methods approach to our work. *Communication Design Quarterly*, *4*(3), 46-50.

Lauer, C., & Brumberger, E. (2016). Technical communication as user experience in a broadening industry landscape. *Technical Communication*, *52*(3), 248-264.

Meloncon, L., & St.Amant, K. (2019). Empirical research in technical and professional communication: A 5-year examination of research methods and a call for research sustainability. *Journal of Technical Writing and Communication*, *49*(2), 128-155.

Mirel, B. (1991). Critical review of experimental research on the usability of hard copy documentation. *IEEE Transactions on Professional Communication*, *34*(2), 109-122.

Mirel, B. (2002). Advancing a vision of usability. In B. Mirel & R. Spilka (Eds.), *Reshaping technical communication* (pp. 165-187). Lawrence Erlbaum.

Muller, M. J. (2009). Participatory design: The third space in HCI. In A. Sears & J. A. Sacko (Eds.), *Human-computer interaction* (pp. 165-185). CRC Press.

Nielsen, J. (1993). *Usability engineering*. Morgan Kauffman.

Norman, D. (2013). *The design of everyday things*. (Revised and expanded ed.). Basic Books.

Norman, D., & Nielsen, J. (n.d.). *The definition of user experience (UX)*. Nielsen Norman Group. https://www.nngroup.com/articles/definition-user-experience/

Opel, D., & Rhodes, J. (2018). Beyond student as user: Rhetoric, multimodality, and user-centered design. *Computers and Composition*, *49*, 71-81.

Pope-Ruark, R. (2019). Design thinking in Technical and Professional Communication: Four perspectives. *Journal of Business and Technical Communication*, *33*(4), 437-455.

Pope-Ruark, R., Tham, J., Moses, J., & Conner, T. (2019). Introduction to special issue: Design-thinking approaches in Technical and Professional Communication. *Journal of Business and Technical Communication*, *33*(4), 370-375.

Potts, L. (2014). *Social media and disaster response: How experience architects can build for participation*. Routledge.

Potts, L., & Salvo, M. J. (2017). Introduction. In L. Potts & M. J. Salvo (Eds.), *Rhetoric and experience architecture* (pp. 3-13). Parlor Press.

Quesenbery, W. (2003). The five dimensions of usability. In M. J. Albers & B. Mazur (Eds.), *Content and complexity: Information design in Technical Communication* (pp. 81-102). Lawrence Erlbaum.

Redish, J. (1995). Are we really entering a post-usability era? *ACM SIGDOC Asterisk Journal of Computer Documentation*, *19*(1), 18-24.

Redish, J. (2010). Technical communication and usability: Intertwined strands and mutual influences commentary. *IEEE Transactions on Professional Communication*, *53*(3), 191-201.

Redish, J., & Barnum, C. (2011). Overlap, influence, intertwining: The interplay of UX and technical communication. *Journal of Usability Studies*, *6*(3), 90-101.

Rose, E. (2016). Design as advocacy: Using a human-centered approach to investigate the needs of vulnerable populations. *Journal of Technical Writing and Communication*, *46*(4), 427-445.

Salvo, M. J. (2001). Ethics of engagement: User-centered design and rhetorical methodology. *Technical Communication Quarterly*, *10*(3), 273-290.

Stephen, D. (2019, July 27). *What is co-design—Kelli Ann McKercher—Beyond Sticky Notes*. Duncan Stephen. https://duncanstephen.net/what-is-co-design-kelly-ann-mckercher-beyond-sticky-notes/

Still, B. (2010). Mapping usability: An ecological framework for analyzing user experience. In M. Albers & B. Still (Eds.), *Usability of complex information: User goals and information needs for dynamic web information* (pp. 90-108). Lawrence Erlbaum.

Still, B., & Crane, K. (2016). *Fundamentals of user-centered design*. CRC Press.

Zachry, M., & Spyridakis, J. (2016). Human-centered design and the field of Technical Communication. *Journal of Technical Writing and Communication*, *46*(4), 392-401.