1

Methods of User Centered Design and Evaluation for Learning Designers

Matthew Schmidt, Yvonne Earnshaw, Andrew A. Tawfik, & Isa Jahnke

User Experience   User-centered Design   Human-computer Interaction   Learning Design   Evaluation Methods

Various theories and models have been published that guide the design and development of learning technologies. While these approaches can be useful for promoting cognitive or affective learning outcomes, user-centered design methods and processes from the field of human-computer interaction (HCI) can also be of value to those in the learning design and instructional design and technology (LIDT) community. In this chapter, we present user-centered design, development, evaluation methods, and processes derived from HCI that lead to highly usable technologies promoting positive user experiences. We begin by aligning these methods and processes with theories commonly referenced in the field of LIDT. We then outline specific methodologies that can be applied during the design and development of digital learning environments. The detailed descriptions outline the goals of the various methods and the ideal stage in which to apply the methods; theory and practice are also discussed. Multiple case examples illustrating how the methods can be used in practice are provided.

Editor's Note

A previous version of this chapter was included in a prior publication (cited below). The current version extends on the previous version by providing examples relevant to the field of Learning/Instructional Design, further clarifications, and additional illustrative figures.

Earnshaw, Y., Tawfik, A., & Schmidt, M. (2017). User experience design. In R. E. West (Ed.), *Foundations of learning and instructional design technology* (1st ed.). EdTech Books. https://edtechbooks.org/-ENoi

---

1. Introduction

Educators and learners are increasingly reliant on digital tools to facilitate learning. However, educators and learners often use technology in ways that are different than developers originally intended (Straub, 2017). For instance, educators may be faced with challenges trying to determine how to assess student learning in their learning management system (LMS), so they use a different tool and then copy/paste the results. Or they might spend time

determining workarounds to administer lesson plans because the LMS does not directly support a particular pedagogical approach. From the perspective of learners, experiencing challenges navigating an interface or finding homework details might result in frustration or even missed assignments. When an interface is not easy to use, users tend to develop alternative paths to complete a task to accomplish a learning goal. Long recognized in the field of human-computer interaction (HCI), such adjustments, accommodations, and improvisations are the result of design flaws (cf. Orlikowski, 1990; Grudin, 1988). These design flaws are often the result of the software development team failing to consider the user sufficiently in the design process. This extends to the field of learning design and instructional design and technology (collectively LIDT) and can create barriers to effective instruction (Jou et al., 2016; Rodríguez et al., 2017). Increasingly, user-centered approaches to design are being accepted as particularly useful in supporting positive user experience. User-centered design (UCD) emphasizes understanding users' needs and expectations throughout all phases of design (Norman, 1986).

Understanding how educators and learners interact with learning technologies is key to avoiding and remediating design flaws. HCI seeks to understand the interaction between technology and the people who use it from multiple perspectives (Rogers, 2012)—two of which are user experience (UX) and usability. UX describes the broader context of technology usage in terms of "a person's perceptions and responses that result from the use or anticipated use of a product, system, or service" (International Organization for Standardization, 2010, Terms and Definitions section, para 2.15). UX considers all aspects of a user's interaction with technology, including how pleasing and usable the technology is. More specifically, usability describes how easy or difficult it is for users to interact with a user interface in the manner intended by the software developer (Nielsen, 2012). Highly usable user interfaces are easy for users to become familiar with, support users achieving their goals, and are easy to remember. From the perspective of learning design, these design factors are used strategically to focus cognitive resources primarily on the task of learning.

The principles of HCI and UCD have implications for the design of learning environments. While the field of LIDT has focused historically on theories that guide learning design (e.g., scaffolding, sociocultural theory), less emphasis has been placed on learning technology design from the view of HCI and UCD (Okumuş et al., 2016). This chapter addresses this issue. We begin with a discussion of some of the theories used in the field of LIDT that align with UX. We then discuss the importance of iteration in design cycles and provide implications with details of UCD-specific methodologies that allow learning designers to approach design from both pedagogical and HCI perspectives. Multiple case examples drawn from the authors' real-world experiences are provided, illustrating how this can be enacted in practice. The intention of this chapter is to highlight how the fields of HCI and LIDT can intersect synergistically by aligning theories and design approaches of LIDT with methods and processes more commonly used in the field of HCI.

2. Theoretical Foundations

Usability and HCI are often situated in established theories such as cognitive load theory, distributed cognition, and activity theory. LIDT is a sister of these disciplines; hence, these theories also have ramifications for the design and development of learning technologies. In the following sections, we discuss each theory and the importance of conceptualizing UCD, usability, and UX from the LIDT perspective.

2.1. Cognitive Load Theory

Cognitive load theory (CLT) contends that learning is predicated on effective cognitive processing; however, an individual only has a limited number of resources needed to process the information (Mayer & Moreno, 2003; Paas & Ayres, 2014). The three categories of CLT include: (a) intrinsic load, (b) extraneous load, and (c) germane load (Sweller et al., 1998). Firstly, intrinsic load describes the active processing or holding of verbal and visual representations within working memory. Secondly, extraneous load includes the elements that are not essential for learning but are still present for learners to process (Korbach et al., 2017). Thirdly, germane load describes the relevant load imposed by the effective instructional/learning design of learning materials (hereafter referred to simply as learning design). Germane cognitive load is therefore relevant to schema construction as information is subsumed into long-term memory (Paas et al., 2003; Sweller et al., 1998; van Merriënboer & Ayres, 2005). It is important to note that the elements of CLT are

additive, meaning that, if learning is to occur, the total load cannot exceed available working memory resources (Paas et al., 2003).

Extraneous load is of particular importance for UCD. Extraneous cognitive load can be directly manipulated by a designer (van Merriënboer & Ayres, 2005) through improved usability. When an interface is not designed with usability in mind, the extraneous cognitive load is increased, which impedes meaningful learning. From a learning design perspective, poor usability might result in extraneous cognitive load in many forms. For instance, a poor navigation structure in an online course might require the learner to extend extra effort to click through the learning modules to find relevant information. Further, when an instructor uses unfamiliar terms in digital learning materials that do not align with a learners's mental model or the different web pages in a learning module are not consistently designed, the learner must exert additional effort toward understanding the materials. Another example of extraneous cognitive load is when a learner does not know how to progress in a digital learning environment, resulting in an interruption of learning flow. Although there are many other examples, each depicts how poor usability taxes finite cognitive resources. Creating highly usable digital environments for learning can help reduce extraneous cognitive load and allow mental resources to remain focused on germane cognitive load for building schemas (Sweller et al., 1998).

2.2. Distributed Cognition and Activity Theory

While cognitive load theory helps describe the individual experience of user actions and interactions, other theories and models focus on broader conceptualizations of HCI. Among the most prominent are distributed cognition and activity theory, which take into account the broader context of learning and introduce the role of collaboration between various individuals. Distributed cognition postulates that knowledge is present both within the mind of an individual and across artifacts (Hollan et al., 2000). The theory focuses on the understanding of the coordination "among individuals and artifacts, that is, to understand how individual agents align and share within a distributed process" (Nardi, 1996, p. 39). From the perspective of LIDT, individual agents (e.g., learners, instructors) operate within a distributed process of learning, as facilitated by various artifacts (such as content, messages, and media). The distributed process of learning is mediated by intentional interaction and communication with learning technologies (e.g., learning management systems, web conferencing platforms) in pursuit of learning objectives (Boland et al., 1994; Vasiliou et al., 2014). For example, two learners collaborating on a pair of programming problems might write pseudo-code and input comments into a text editor. In this case, distributed cognition is evident in collaborating on the programming problem and by conceptualizing various solutions mentally but also by using a tool (the text editor) to extend their memory. Cognition in this case is distributed between people and tools; distributed cognition, therefore, would focus on the function of the tool within the broader learning context (Michaelian & Sutton, 2013). In contrast with the more narrow perspective of cognitive load theory that considers the degree to which a specific learner's finite cognitive resources are affected when interacting with a technology system, distributed cognition adopts a broader cognitive, social, and organizational perspective (Rogers, 1997).

Activity theory is a systems-based, ecological framework that shares some similarities with distributed cognition but distinguishes itself in its specific focus on activity and the dynamic interplay of actors, artifacts, and sociocultural factors within an interconnected system. Given its ecological lens, activity theory can be a useful framework for describing and understanding how a variety of factors can influence human activity. Central to activity theory is the concept of mediation. In activity theory, activity is mediated by tools, also called artefacts (Kaptelenin, 1996). From a technological perspective the concept of tools is often in reference to digital tools or software. These technological tools mediate human activity within a goal-directed hierarchy of (a) activities, (b) actions, and (c) operations (Jonassen & Rohrer-Murphy, 1999). Firstly, activities describe the top-level objectives and fulfillment of motives (Kaptelinin et al., 1999). Secondly, actions are the more specific goal-directed processes and smaller tasks that must be completed in order to complete overarching activities. Thirdly, operations describe the automatic cognitive processes that group members complete (Engeström, 2000). However, they do not maintain their own goals but are rather the unconscious adjustment of actions to the situation at hand (Kaptelinin et al., 1999). Engström's (2000) sociocultural activity theory framework is commonly depicted as an interconnected system in the shape of a triangle, as depicted in Figure 1.
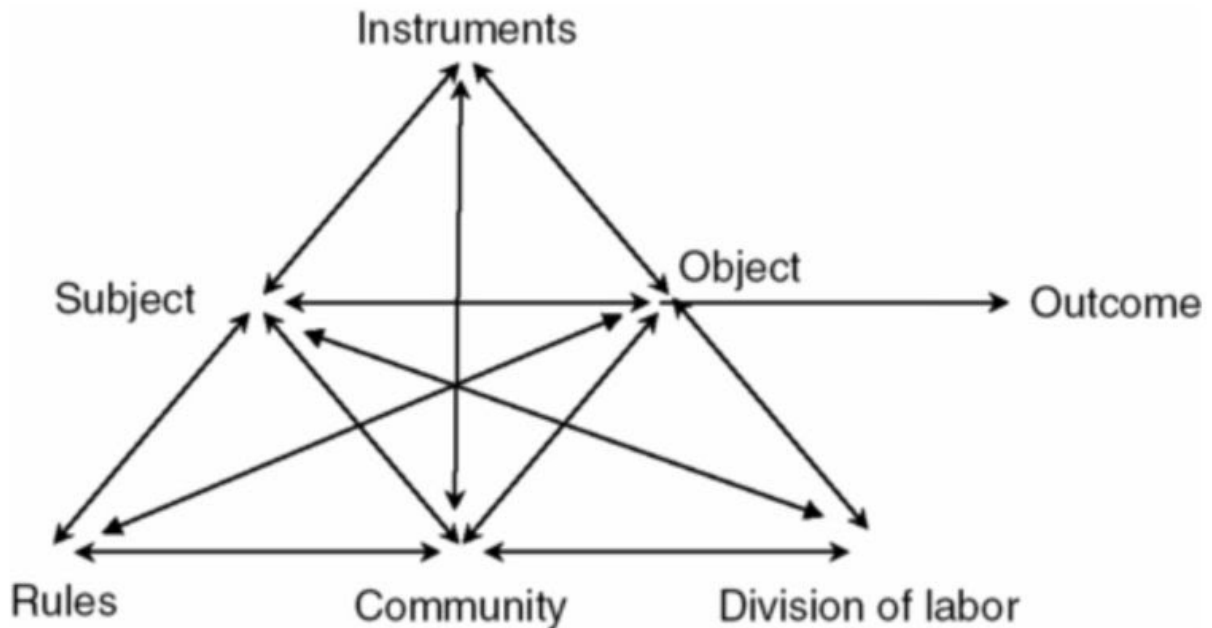
Figure 1

Activity System Diagram

Note. Adapted from Engeström (2000, p. 962).

Activity theory is especially helpful for learning design because it provides a framework to understand how objectives are completed within a learning context. Nardi (1996) highlights the centrality to activity theory of mediation via tools/artefacts. These artefacts are created by individuals to control their own behavior and can manifest in the form of instruments, languages, or technology. Each carries a particular culture and history that stretches across time and space (Kaptelinin et al., 1999) and serves to represent ways in which others have solved similar problems. As applied to learning contexts, activity theory suggests that tools not only mediate the learning experience but that learning processes are often altered to accommodate the new tools (Jonassen & Rohrer-Murphy, 1999). This underscores the importance of considering the influence of novel learning technologies (e.g., LMSs, educational video games) from within a broader context of social activity when implemented by schools and/or organizations (Ackerman, 2000). The technological tools instituted in a particular workgroup should not radically change work processes but should present solutions on the basis of needs, constraints, history, etc. of that workgroup (Barab et al., 2002; Yamagata-Lynch et al., 2015). As learning is increasingly collaborative through technology (particularly online learning), activity theory and distributed cognition can provide important insights for learning designers into the broader sociocultural aspects of human-computer interaction.

3. User-Centered Design

The brief overview of theoretical foundations provided in the above sections highlights how theories of cognition and human activity in sociocultural contexts can be useful in the design of digital environments for learning. However, the question remains as to how one designs highly usable, pleasing, and effective digital environments for learning on the basis of these theories. Answering this question is difficult because these theories are not prescriptive. Specific guidance for how they can be applied is lacking, meaning that how best to design theoretically inspired, highly usable and pleasing learning environments is ultimately the prerogative of the designer. Iterative design approaches can be useful for confronting this conundrum. While the field of LIDT has recently begun to shift its focus to more iterative design and user-driven development models, there is a need to more intentionally bridge learning design and user-

centered design approaches to support positive learner experience in digital environments. To this end, a number of existing learning design methods can be used or adapted to fit iterative approaches. For example, identifying learning needs has long been the focus of front-end analysis. Ideation and prototyping are frequently used methods from UX design and rapid prototyping. Evaluation in learning design has a rich history of formative and summative methods. By applying these specific design methods within iterative design processes, learning designers can advance their designs in such a way that they can focus not only on intended learning outcomes but also on the learner experience and usability of their designs. In the following sections, UCD is considered with a specific focus on techniques for incorporation into one's learning design processes through (a) identifying user needs, (b) requirements gathering, (c) prototyping, and (d) wireframing.

3.1. Developing Requirements Based on Learners' Needs

One potential pitfall of any design process occurs when designers create systems based on assumptions of what users want. Only after designers have begun to understand the user should they begin to identify what capabilities or conditions a system must be able to support to meet the identified needs. These capabilities or conditions are known as requirements. The process a designer undertakes to identify these requirements is known as requirements gathering. Generally, requirements gathering involves gathering and analyzing user data (e.g., surveys, focus groups, interviews, observations) and assessing user needs (Sleezer et al., 2014).

In the field of LIDT, assessing learner needs often begins with identification of a gap (the need) between actual performance and optimal performance (Rossett, 1987; Rossett & Sheldon, 2001). Needs and performance can then be further analyzed and learning interventions designed to address those needs. Assessing user (and learner) needs can yield important information about performance gaps and other problems. However, knowledge of needs alone is insufficient to design highly usable and pleasing learning environments. Further detail is needed regarding the specific context of use for a given tool or system. Context is defined by learners (and others who will use the tool or system such as administrators or instructors), tasks (what will learners do with the tool or system), and environment (the local context in which learners use the tool or system).

Based on identified learner needs, a set of requirements is generated to define what system capabilities must be developed to meet those needs. Requirements are not just obtained for one set of learners but for all learner types and personas (including instructors and administrators) that might utilize the system. Data-based requirements (a) help learning designers avoid the pitfall of applying ready-made solutions to assumed learner needs, (b) position the learner and their needs centrally in the design process, and (c) allow for creation of design guidelines targeting an array of various learner needs. Requirements based on learner data are therefore more promising in supporting a positive learner experience. However, given the iterative nature of UCD, requirements might change as a design evolves. Shifts in requirements vary depending on design and associated evaluation outcomes. Two methods commonly used in UCD for establishing requirements are persona and scenario development.

3.1.1. Personas

In UCD, a popular approach to understanding users is to create what is known as personas (Cooper, 2004). Personas provide a detailed description of a fictional user whose characteristics represent a specific user group. They serve as a methodological tool that helps designers approach design based on the perspective of the user rather than (often biased) assumptions. A persona typically includes information about a user's demographics, goals, needs, typical day, and experiences. In order to create a persona, interviews or observations should gather information from individual users and then place them into specific user categories. Personas should be updated if there are changes to technology, business needs, or other factors. These archetypes help designers obtain a deep understanding of the types of users for the system. Personas are especially helpful for learning designers in considering cultural diversity. Learning design teams tend to be small (2-3 members) or consist of an individual learning designer. Such teams can lack sufficient sociocultural perspective to design for a culturally sensitive and diverse learner experience. However, developing personas of, for example, a 25 year-old African-American woman who is a first generation college student or a 17 year-old, male Asian-American high school student athlete can provide context for designers to consider these

sociocultural perspectives more intentionally in their learning designs. Because learner personas should be developed based on data that have been gathered about those learners, implicit bias can be reduced.

Table 1 provides an example of a culturally-situated persona in the context of Hawaiian public schools that was created by novice designers in an introductory learning design course using a template. The design context was development of a parent-teacher communication portal for public schools throughout the state using the Hawaii Department of Education E-School course management system. This particular persona highlights the value that Hawaiian families tend to place on family and interpersonal relationships.

Table 1

Persona of Parent-Teacher Communication Portal Users

Note. Derived from: http://usabilitybok.org/persona

User Goals: What users are trying to achieve by using your site, such as tasks they want to perform
1. Parents seek advice on improving teacher/parent interactions 2. Parents seek to build and foster a positive partnership between teacher and parents to contribute to child's school success 3. Parents wish to find new ways or improve ways of parent-teacher communication
Behavior: Online and offline behavior patterns, helping to identify users' goals
1. Online behavior: "Googling" ways to improve teacher communication with parent or parent communication with teacher; parent searching parent/teacher communication sites for types of technology to improve communication; navigating through site to reach information 2. Offline behavior: Had ineffective or negative parent-teacher communication over multiple occurrences; parents seeking out other parents for advice or teachers asking colleagues for suggestions to improve communication with parents 3. Online/Offline behavior: Taking notes, practicing strategies or tips suggested, discussing with a colleague or friend.
Attitudes: Relevant attitudes that predict how users will behave
1. Looking for answers 2. Reflective 3. Curiosity-driven
Motivations: Why users want to achieve these goals
1. Wishing to avoid past unpleasant experiences of dealing with parent-teacher interaction 2. Looking to improve current or future parent-teacher relationships 3. Looking to avoid negative perceptions of their child by teacher
Design team objectives: What you ideally want users to accomplish in order to ensure your website is successful?

> 1. Have an interface that is easy to navigate
> 2. Inclusion of both parent and teacher in the page (no portal/splash page)
> 3. Grab interest and engage users to continue reading and exploring the site

3.1.2. Scenarios

A complimentary method to personas is scenarios. Scenarios provide a means to situate the user/learner persona and technology within a realistic context of usage while the learner attempts to achieve his or her goal. Scenarios are presented as narratives that describe user activity in an informal story format (Carroll, 2000). While scenarios are widely used in software development, there is little specific guidance on how they should be developed. Generally speaking, scenarios should be developed in such a way that they are able to provide the designer useful detail about contexts, needs, and goals, which can be used to highlight necessary requirements.

Table 2 provides an example scenario that was created in the context of a virtual reality (VR) learning intervention for youth with autism spectrum disorders. The design target of this scenario was a tool that would allow learners to compare snapshots of their own facial expressions with a standard model inside of the VR world. In this scenario, the learner persona "John" interacts with the teacher persona "Carla" to engage in the task. This scenario illustrates how a scenario helps to illustrate how a learner persona (in this case, John) engages with a learning technology.

Table 2

Scenario of Learner With Autism Using a Virtual Reality Tool to Learn Facial Expressions

Component	Component Description
Context	John is viewing images of faces showing emotions and states including happy, surprised, and disappointed in the collaborative virtual world. His teacher, Carla, has asked him to make a face showing he is sad and share it with the group.
Goal	John's goal is to take a webcam picture of himself using the tools provided in the VR interface and to discuss his picture with his teacher and the rest of his group.
Activity	John learned to use the camera when he was completing his orientation, so he knows how to do this. John tries to make a sad face and snaps his picture using the Live Images application on the heads-up display. His picture shows up automatically on a shared media board in the virtual world. John's picture takes up a large portion of the media board, since it is the only picture. Carla and John look at the picture, and then Carla makes a suggestion for how his expression could better show sadness. Carla says, "I'll remove this picture and would like you to try again?" She deletes the first image. John retakes the image and asks Carla, "Does this face look sad enough?" Carla provides positive praise, "I really like how you asked me about your picture!" and continues, "Let's ask the rest of the group."
Outcome	The whole group discusses John's picture and provides their input. After their discussion is over and John has some feedback, he asks if he can try again. Carla deletes his image and John takes another

image to share. After everyone praises John for getting it right this time, Carla deletes John's image and asks Mary to try to show a surprised look.

## 3.2. Prototyping Digital Environments for Learning

Gathering data to design and develop digital environments for learning is an iterative process. Based on personas and identified requirements, an initial prototype of the user interface or the online learning environment will be created. Prototypes tend to follow a trajectory of development over time from low fidelity to high fidelity (Walker et al., 2002). Fidelity refers to the degree of precision, attention to detail, and functionality of a prototype. Examples range from lower fidelity prototypes, which include the proverbial "sketch on a napkin" and paper prototypes, to higher fidelity prototypes, which include non-functional "dummy" graphical mockups of interfaces and interfaces with limited functionality that allow for evaluation. Typically, lower fidelity prototypes (lo-fi prototypes) do not take much time to develop and higher fidelity prototypes take longer because prototypes become more difficult to change as more details and features are added. Prototyping is a useful skill for all learning designers, including those who create online courses by arranging various content, media, and interactive experiences to those who develop educational software such as educational video games or mobile apps.

3.2.1. Rapid Prototyping

Rapid prototyping is an approach to design that emerged in the 1980s in engineering fields and began to gain traction in instructional design in the early 1990s (Desrosier, 2011; Tripp & Bichelmeyer, 1990; Wilson et al., 1993). Instead of traditional instructional design approaches with lengthy design and development phases, rapid prototyping focuses on fast, or "rapid," iterations. This allows instructional designers to quickly gather evaluative feedback on their early designs. Considered a feedback-driven approach, rapid prototyping is seen by many as a powerful tool for the early stages of a learning design project. The rapid prototyping approach relies on multiple, rapid cycles in which an artifact is designed, developed, tested, and revised. Actual users of the system participate during the testing phase. This cycle repeats until the artifact is deemed to be acceptable to users. Although high fidelity prototypes can emerge from the process of rapid prototyping, rapid prototypes themselves are usually lo-fi. An example of rapid prototyping applied in an instructional design context is the successive approximation model or SAM (Allen, 2014). The SAM (version 2) process model is provided in Figure 2.

For example, a learning designer developing a course in a LMS can benefit from rapid prototyping processes like SAM2 before a course is deployed. After gathering information and materials (preparation phase), he or she can quickly incorporate as many course elements and materials as are immediately available into the LMS (iterative design phase). For any materials or content that is missing, simple placeholders are used with relevant descriptions (e.g., an image with an "X" on it to designate a graphic or a screenshot of a video player to designate a video). These materials are then arranged to provide a rough estimation of how the course navigation, structure, sequence, and associated learning materials will be organized. This is then reviewed by students (who do not necessarily need to be students enrolled in the course) and iterated over two or three redesign and revision cycles. Once the organization has been refined, course materials can be developed (e.g., multimedia, text-based content) and evaluated (iterative development phase). These materials are often evaluated by subject matter experts in the form of expert review. After two or three rounds of revisions and refinements are completed, the course is ready to be rolled out. Due to the revisions the course has a far greater likelihood to promote a positive learner experience than a course that is organized based solely on an LMS template or designer intuition.
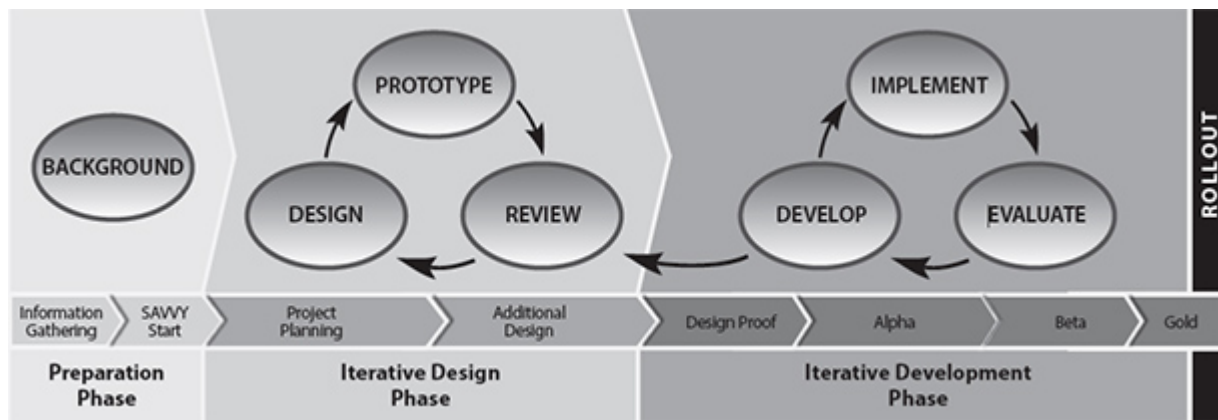
Figure 2

*Successive Approximation Model Version 2 (SAM2) Process Diagram*

Note. Adapted from Allen (2014). Copyright 2014 by the American Society for Training and Development.

3.2.2. Paper Prototyping

Paper prototyping is a lo-fi method of prototyping used to inform the design and development of many different kinds of interfaces, including web, mobile, and games. The focus of paper prototyping is not on layout or content but on navigation, workflow, terminology, and functionality. The purpose of creating these prototypes is to communicate designs among the design team, users, and stakeholders, as well as to gather user feedback on designs. A benefit of paper prototyping is that it is rapid and inexpensive—designers put only as much time into developing a design as is absolutely necessary. This makes it a robust tool at the early stages of design. As the name implies, designers use paper to create mockups of an interface. Using pencil and paper is the simplest approach to paper prototyping, but stencils, colored markers, and colored paper can also be used. These paper prototypes can be scanned and further elaborated using digital tools (Figure 3). The simplicity of paper prototyping allows for input from all members of a design team, as well as from users and other stakeholders. The speed of paper prototyping makes it particularly amenable to a rapid prototyping design approach. The process of creating paper prototypes can be individual, in which the designer puts together sketches on his or her own, or collaborative, in which a team provides input on a sketch while one facilitator draws it out. For further information on paper prototyping, refer to Snyder (2003) and UsabilityNet (2012).

For example, a learning designer planning to create a learning object using an authoring tool such as Articulate Storyline or Adobe Captivate can benefit from paper prototyping by establishing rough drafts of animations, interactions, or navigation before devoting time and effort to developing those things in the authoring environment. For example, Figure 3 illustrates a case vignette in which a child avatar with a behavior disorder gets into an intense verbal argument with a caregiver avatar. The scene sets up an interactive activity in which the learner selects from a variety of responses to the situation and receives specific feedback based on those decisions. The initial sketch considers visual design (sequence of scenes, positioning of the avatars, avatar facial expressions, placement of user interface elements, etc.), the tone of the language, potential animations (fade-in of "what could I do diff?"), how learners will interact with the learning object (e.g., should the scenes "autoplay" or should the user manually advance them?), and anticipates the following interactive activity. The design team has also added a design idea of potentially presenting the vignette in a comic book style. As the reader will note, there are deep and meaningful learning design considerations represented in this paper prototype that took less than three minutes to sketch, photograph, and digitally annotate. This then served as the basis for further discussions within the design team and to solicit feedback from a subject matter expert. These conclusions were then incorporated in another rapid prototype, and another, and so-on until the design was sufficiently developed to build out in a more robust authoring tool.
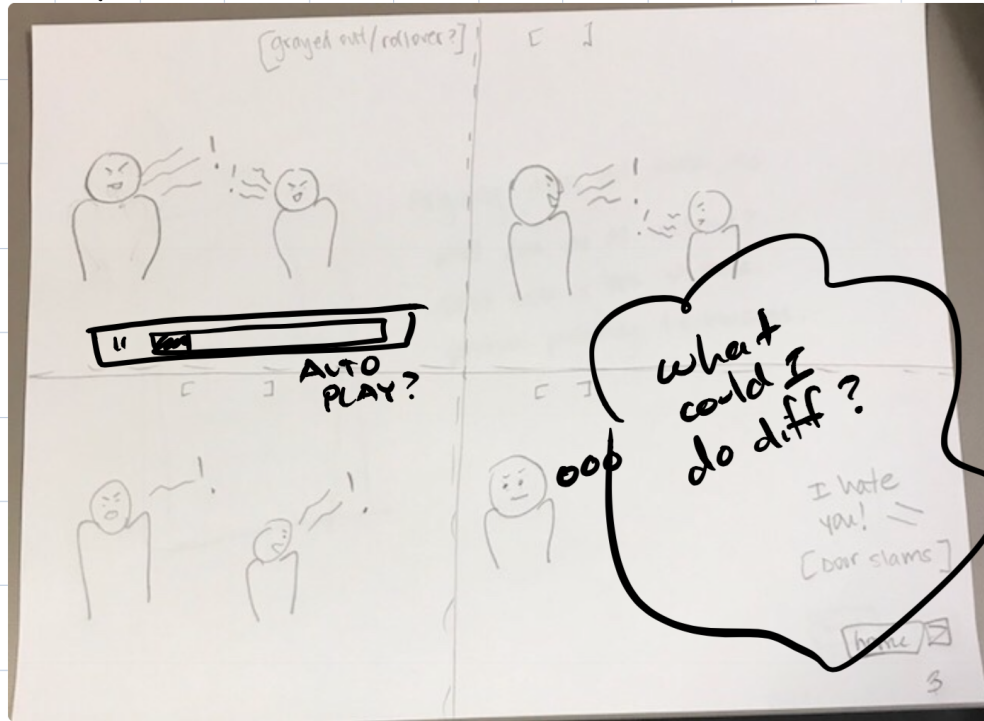
Figure 3

Example of a Paper Prototype That Has Been Scanned and Annotated Using Digital Tools

3.2.3. Wireframing

Wireframes are medium fidelity representations of interfaces that visually convey their structure (see Figure 4). Wireframing results in prototypes that are of higher fidelity than paper prototyping but lack the functionality and visual elements of high fidelity prototypes. Wireframing commonly occurs early in the design process after paper prototyping. It allows designers to focus on things that paper prototyping does not, such as layout of content, before more formal visual design and content creation occurs. Wireframing can be seen as an interim step that allows for fast mockups of an interface to be developed, tested, and refined, the results of which are then used to create higher fidelity, functional prototypes.
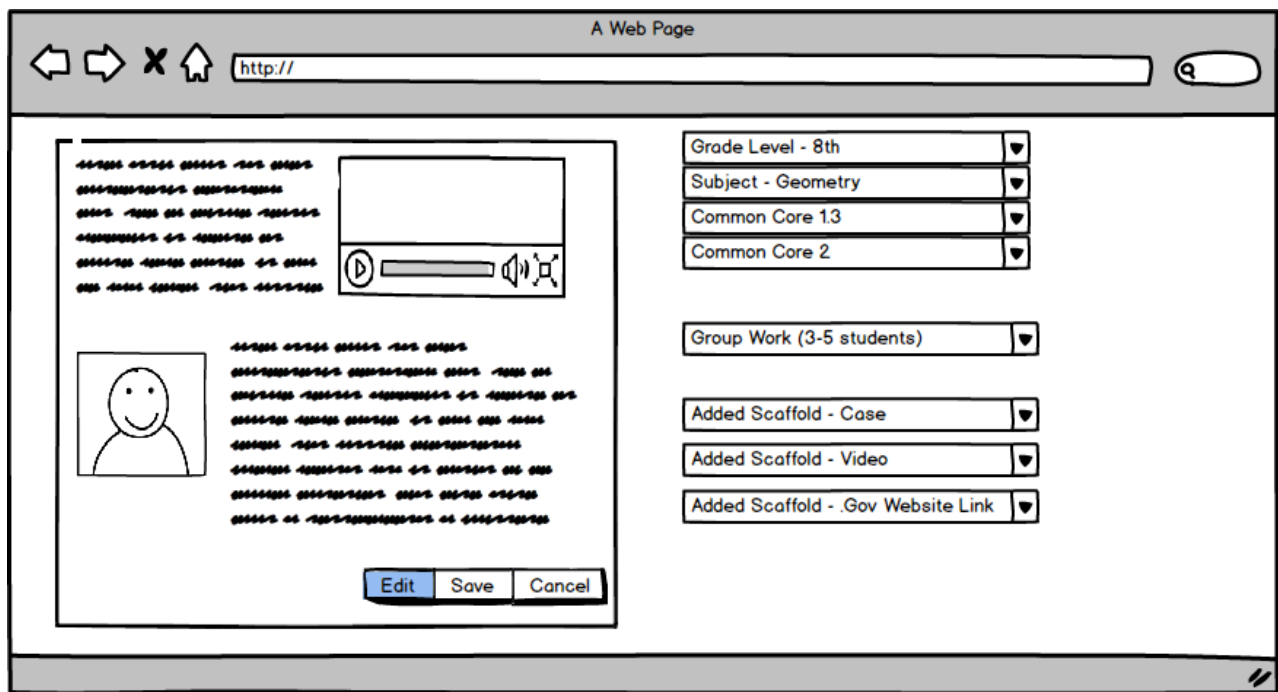
Figure 4

Example of a Wireframe

Wireframes consist of simple representations of an interface, with interface elements displayed as placeholders. Placeholders use a variety of visual conventions to convey their purpose. For example, a box with an "X" or other image might represent a graphic, or a box with horizontal lines might represent textual content. Wireframes can be created using common software such as PowerPoint or Google Drawings or with more specialized software such as OmniGraffle or Balsamiq. Wireframes are particularly amenable to revision, as revisions often consist of simple tweaks, such as moving interface elements, resizing, or removing them. A key benefit of wireframes is that they allow designers to present layouts to stakeholders, generate feedback, and quickly incorporate that feedback into revisions.

For example, learning designers developing a course in an LMS often incorporate multiple multimedia elements on a single LMS page. This could be a page consisting primarily of text interspersed with graphical illustrations or a page that presents three interactive three-dimensional models within a quiz. Learning designers can avoid unnecessary effort by developing wireframes for how content will be structured on these pages and then soliciting feedback. While creating wireframes for individual pages can increase designer efficiency, economies of scale can be achieved by wireframing entire learning modules and even entire course structures. These collections of wireframes provide a basis upon which to solicit feedback (i.e., from SMEs, students, etc.) and make subsequent improvements, thereby increasing the likelihood of a more positive learner experience. In addition, after designs are approved, the wireframe set can serve as a "punch list" for a learning design team, allowing the team to keep track of what content is needed, how it should be structured, and where it should be organized. As such, wireframes can be a tremendously useful communication and project management tool for a learning design team.

3.2.4. Functional Prototyping

Functional prototypes are higher-fidelity graphical representations of interfaces that have been visually designed such that they closely resemble the final version of the interface and that incorporate limited functionality. In some cases, content has been added to the prototype. A functional prototype might start out as a wireframe interface with links between screens. A visual design is conceived and added to the wireframe, after which graphical elements and content are added piece-by-piece. Then, simple functionality is added, typically by connecting different sections of the interface using hyperlinks. An advanced functional prototype might look like a real interface but lack full functionality. Functional prototypes can be created using PowerPoint or with more specialized software like InVision and UXPin. During

evaluation, functional prototypes allow for a learner to experience a mockup online course, mobile app, or educational software interface in a way that is very similar to the experience of using the actual product. However, because functionality is limited, development time can be reduced substantially. Functional prototypes provide a powerful way to generate feedback from learners in later stages of the learning design process, allowing for tweaks and refinements to be incorporated before time and effort are expended on development.

For example, imagine that a learning designer has received approval on a wireframe set for mobile microlearning materials for parents and caregivers of children with epilepsy (Figure 5). The designer imports the wireframes into InVision, a clickable prototyping tool, and sets up "hotspots" on the wireframe images. These hotspots are hyperlinks to other wireframes. By creating hotspots on all wireframes, the learning designer creates a simulation of how learners will interact with the mobile microlearning materials. The designer then sends this functional prototype to subject matter experts, who are attending an academic conference. These subject matter experts review the functional prototype and also share it with other academics in their discipline. By allowing other experts to actually experience how the mobile microlearning materials look and function, a wealth of informal feedback is generated that is then fed back to the learning designer. The learning designer then incorporates the expert feedback into the wireframes and creates a new clickable, functional prototype. This new functional prototype is then usability tested with a representative parent, and the process continues. In this way, content, visual design, and interaction design can all be tested before any actual learning materials are created or development takes place. This allows for continual, rapid, and targeted refinements, thereby increasing the likelihood for a positive learner experience.
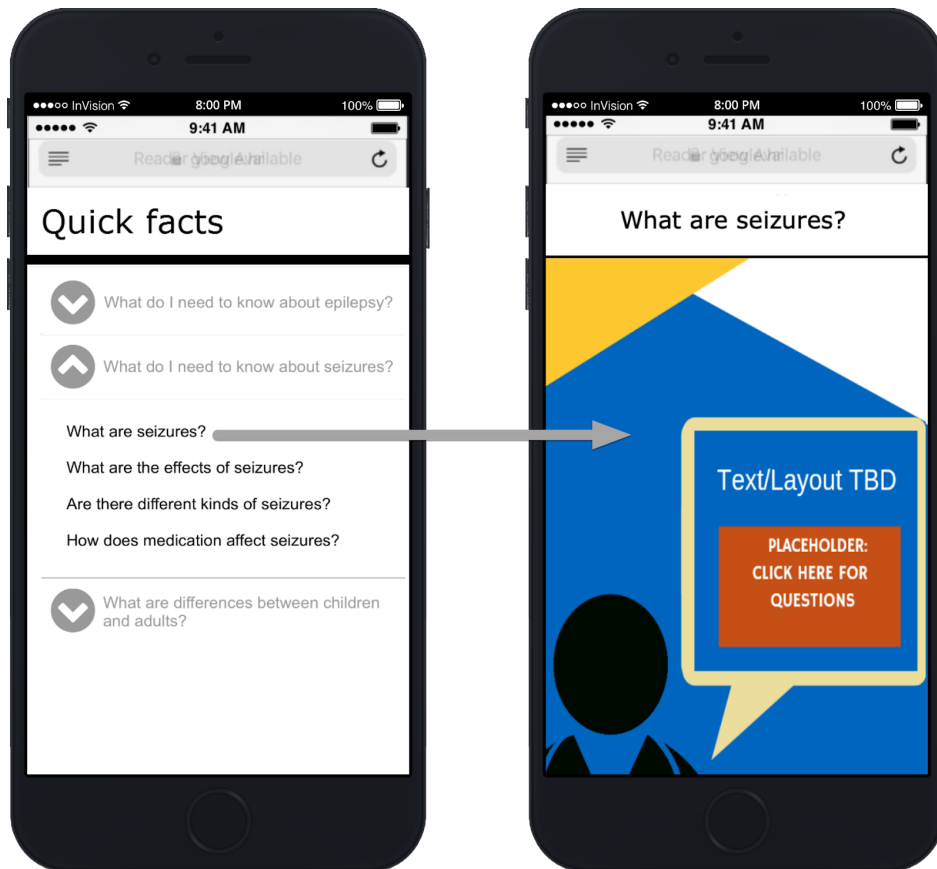
Figure 5

Functional Prototype of a Mobile Microlearning ASK System Developed for Parents of Children With Epilepsy Illustrating Clickable "Hotspots" That Allow Designers to Simulate How a Learning Environment Functions

To reiterate, the goal of UCD is to approach systems development from the perspective of the end-user. Using tools such as personas and prototypes, the learning design process becomes iterative, dynamic, and more responsive to

learner needs. Learning designers often use these tools in conjunction with a variety of evaluation methods to better align prototype interface designs with learners' mental models, thereby reducing cognitive load and improving usability. Evaluation methods are discussed in the following section.

4. Evaluation Methodologies for User-Centered Design

While UCD is important for creating usable interfaces, a challenge is knowing when and under what conditions to apply evaluation methodologies. In the following sections, several evaluation methodologies commonly used in UCD are described, with descriptions of how these evaluation methodologies can be used in a learning design context. These can be applied during various phases across the learning design and development process (i.e., front-end analysis, low-fidelity to high-fidelity prototyping). While a case can be made to apply any of the approaches outlined below in a given design phase, some evaluation methodologies are more appropriate to overall learner experience, while others focus more specifically on usability. Table 3 provides an overview of methods, in which design phase they can be best implemented, and associated data sources.

Table 3

Evaluation Methodologies, Design Phases, and Data Sources

Method	Design phase				Data source
	Front-end analysis	Prototyping			
		Paper (low fidelity)	Wireframe (medium fidelity)	Functional (high fidelity)	
Ethnography	•				Single user or users
Focus groups	•	•			Group of users
Card sorting	•	•			Single user, multiple users or group of users
Cognitive walkthrough		•	•	•	Expert
Heuristic evaluation		•	•	•	Experts
A/B testing		•	•	•	Multiple users
Think-aloud				•	Multiple users
EEG/ Eyetracking				•	Multiple users
Analytics				•	Multiple users

4.1. Ethnography

A method that is used early in the front-end analysis phase, especially for requirements gathering, is ethnography. Ethnography is a qualitative research method in which a researcher studies people in their native setting (not in a lab or controlled setting). During data collection, the researcher observes the group, gathers artifacts, records notes, and performs interviews. In this phase, the researcher is focused on unobtrusive observations to fully understand the

phenomenon in situ. For example, in an ethnographic interview, the researcher might ask open-ended questions but would ensure that the questions were not leading. The researcher would note the difference between what the user is doing versus what the user is saying and take care not introduce his or her own bias. Although this method has its roots in the field of cultural anthropology, UCD-focused ethnography can support thinking about design from activity theory and distributed cognition perspectives (Nardi, 1996). This allows the researcher to gather information about the users, their work environment, their culture, and how they interact with the device or website in context (Nardi, 1997). This information is particularly valuable when writing user personas and scenarios. Ethnography is also useful if the researcher cannot conduct user testing on systems or larger equipment due to size or security restrictions.

A specific example of how ethnography can be applied in learning design is in the development of learner personas. Representative learners can be recruited for key informant interviews with the purpose of gathering specific data on what a learner says, thinks, does, and feels, as well as what difficulties or notable accomplishments they describe. The number of participants needed depends on the particular design context but does not need to be large. Indeed, learning designers can glean critical insights from just a few participants, and there is little question that even small numbers of participants is better than none. For example, to develop online learning resources for parents of children with traumatic brain injuries, a learning designer might interview two or three parents and ask them to relay what their typical day looks like, to tell a story about a particular challenge they have encountered with parenting their child, or to describe how they use online resources to find information about traumatic brain injury. The interviews could then be transcribed, and the learning designer could use a variety of analysis techniques to categorize the interview data thematically. For an approachable method of thematic analysis, the reader is referred to Mortinsen (2020). This information from thematic categories could then be generalized into the development of learner personas that are illustrative of themes derived from the key informant interviews.

4.2. Focus Groups

Focus groups are often used during the front-end analysis phase. Rather than the researcher going into the field to study a larger group as in ethnography, a small group of participants (5-10) are recruited based on shared characteristics. Focus group sessions are led by a skilled moderator who has a semi-structured set of questions or plan. For instance, a moderator might ask what challenges a user faces in a work context (i.e., actuals vs. optimals gap), suggestions for how to resolve it, and feedback on present technologies. The participants are then asked to discuss their thoughts on products or concepts. The moderator may also present a lo-fidelity prototype and ask for feedback. The role of the researcher in a focus group is to ensure that no single person dominates the conversation in order to hear everyone's opinions, preferences, and reactions. This helps to determine what users want and keeps the conversation on track. It is preferred to have multiple focus group sessions to ensure various perspectives are heard in case a conversation gets side-tracked. Analyzing data from a focus group can be as simple as providing a short summary with a few illustrative quotes for each session. The length of the sessions (typically 1-2 hours) may include some extraneous information, so it is best to keep the report simple.

For example, a learning designer developing an undergraduate introduction to nuclear engineering course invited a group of nuclear engineers, radiation protection technicians, and nuclear engineering students to participate in a focus group. The learning designer had created a semi-structured set of questions to guide the session. These questions focused on issues the designer had gleaned from discussions with subject matter experts and from document analysis, such as the upcoming challenge facing the industry of an aging workforce on the brink of retirement and with no immediate replacements, the stigma of nuclear power, and the perceived difficulty of pursuing a career in nuclear engineering. These issues were then explored with the focus group participants, with the designer acting as facilitator. Sticky notes were used to document key ideas and posted around the room. Participants were asked to use sticky notes to provide brief responses to facilitator questions. The facilitator then asked the participants to find the sticky notes posted on the walls that best aligned with the responses they had provided and post their sticky notes near those others. These groups of notes were then reviewed by the groups, refined, and then named. The entire process took two hours. These named groups ultimately formed the basis of the content units of the online course, such as using nuclear medicine to diagnose and treat cancer and irradiation of food to increase shelf life.

4.3. Card Sorting

Aligning designs with users' mental models is important for effective UX design. A method used to achieve this is card sorting. Card sorting is used during front-end analysis and paper prototyping. Card sorting is commonly used in psychology to identify how people organize and categorize information (Hudson, 2012). In the early 1980s, card sorting was applied to organizing menuing systems (Tullis, 1985) and information spaces (Nielsen & Sano, 1995).

Card sorting can be conducted physically using tools like index cards and sticky notes or electronically using tools like Miro (https://miro.com/) or Lloyd Rieber's Q Sort (http://lrieber.coe.uga.edu/qsort/index.html). It can involve a single participant or a group of participants. With a single participant, he or she groups content (individual index cards) into categories, allowing the researcher to evaluate the information architecture or navigation structure of a website. For example, a participant might organize "Phone Number" and "Address" cards together. A set of cards placed together by multiple participants suggests to the designer distinct pages that can be created (e.g., "Contact Us"). When focusing on a group, the same method is employed, but the group negotiates how they will group content into categories. How participants arrange cards provides insight into mental models and how they group content.

In an open card sort, a participant will first group content (menu labels on separate notecards) into piles and then name the category. Participants can also place the notecards in an "I don't know" pile if the menu label is not clear or may not belong to a designated pile of cards. In a closed card sort, the categories will be pre-defined by the researcher. It is recommended to start with an open card sort and then follow-up with a closed card sort (Wood & Wood, 2008). As the arrangement of participants are compared, the designer iterates the early prototypes so the menu information and other features align with how the participants organize the information within their mind. For card sorting best practices, refer to the work of Righi et al (2013).

Card sorting is particularly useful for learning designers creating courses in learning management systems. After identifying the various units, content categories, content sections, etc., the learning designer can (a) write these down on cards (or use other methods discussed above); (b) present them to a SME, course instructor, or student; and (c) ask them to arrange the cards into what they perceive to be the most logical sequence or organization. This approach can be particularly educative when comparing how instructors feel a course should be organized with how a learner feels a course should be organized, which can sometimes be quite disparate. Findings can then be used to inform the organization of the online course.

4.4. Cognitive Walkthroughs

Cognitive walkthroughs (CW) can be used during all prototyping phases. CW is a hands-on inspection method in which an evaluator (not a user) evaluates the interface by walking through a series of realistic tasks (Lewis & Wharton, 1997). CW is not a user test based on data from users, but instead is based on the evaluator's judgments.

During a CW, a UX expert evaluates specific tasks and considers the user's mental processes while completing those tasks. For example, an evaluator might be given the following task: Recently you have been experiencing a technical problem with software on your laptop and you have been unable to find a solution to your problem online. Locate the place where you would go to send a request for assistance to the Customer Service Center. The evaluator identifies the correct paths to complete the task but does not make a prediction as to what a user will actually do. In order to assist designers, the evaluator also provides reasons for making errors (Wharton et al., 1994). The feedback received during the course of the CW provides insight into various aspects of the user experience including:

- first impressions of the interface,
- how easy it is for the user to determine the correct course of action,
- whether the organization of the tools or functions matches the ways that users think of their work,
- how well the application flow matches user expectations,
- whether the terminology used in the application is familiar to users, and
- whether all data needed for a task is present on screen.

In learning design, the CW is particularly valuable when working in teams that consist of senior and junior learning designers. Junior learning designers can develop prototype learning designs (e.g., learning modules, screencasts, infographics), which can then be presented to the senior designer to perform a cognitive walkthrough. For example, a junior designer creates a series of five videos and sequences them in the LMS logically so as to provide sufficient information for a learner to correctly answer a set of corresponding informal assessment questions (e.g., a knowledge check). The junior designer then presents this to the senior designer with the following scenario: You don't know the answer to the third question in the knowledge check, so you decide to review what you learned to find the answer. The senior designer then maps out the most efficient path to complete this task but finds that videos cannot be easily scrubbed by moving the playhead rapidly across the timeline. Instead, the playhead resets to the beginning of the video when it is moved. The senior designer explains to the junior designer that learners would have to completely rewatch each video to find the correct answer, and the junior designer then has specific feedback that can be used to improve the learner experience for this learning module.

4.5. Heuristic Evaluation

Heuristic evaluation is an inspection method that does not involve directly working with the user. In a heuristic evaluation, usability experts work independently to review the design of an interface against a predetermined set of usability principles (heuristics) before communicating their findings. Ideally, each usability expert will work through the interface at least twice: once for an overview of the interface and the second time to focus on specific interface elements (Nielsen, 1994). The experts then meet and reconcile their findings. This method can be used during any phase of the prototyping cycle.

Many heuristic lists exist that are commonly used in heuristic testing. The most well-known heuristic checklist was developed over 25 years ago by Jakob Nielsen and Rolf Molich (1990). This list was later simplified and reduced to 10 heuristics which were derived from 249 identified usability problems (Nielsen, 1994). In the field of instructional design, others have embraced and extended Nielsen's 10 heuristics to make them more applicable to the evaluation of eLearning systems (Mehlenbacher et al., 2005; Reeves et al., 2002). Not all heuristics are applicable in all evaluation scenarios, so UX designers tend to pull from existing lists to create customized heuristic lists that are most applicable and appropriate to their local context. Nielsen's 10 heuristics are:

1. Visibility of system status
2. Match between system and the real world
3. User control and freedom
4. Consistency and standards
5. Error prevention
6. Recognition rather than recall
7. Flexibility and efficiency of use
8. Aesthetic and minimalist design
9. Help users recognize, diagnose, and recover from errors
10. Help and documentation

An approach that bears similarities with a heuristic review is the expert review. This approach is similar in that an expert usability evaluator reviews a prototype but differs in that the expert does not use a set of heuristics. The review is less formal and the expert typically refers to personas to become informed about the users. Regardless of whether heuristic or expert review is selected as an evaluation method, data from a single expert evaluator is insufficient for making design inferences. Multiple experts should be involved, and data from all experts should be aggregated. This is because expert review is particularly vulnerable to an expert's implicit biases. Different experts will have different perspectives and biases and therefore will uncover different issues. Involving multiple experts helps ensure that implicit bias is reduced and that problems are not overlooked.

For learning designers developing online courses, established quality metrics such as Quality Matters (QM) can be used for guiding heuristic evaluations (MarylandOnline, Inc, 2018). QM provides evaluation rubrics for certified evaluators to

assess the degree to which an online course meets QM standards. The aggregate QM score can then be used as a quality benchmark for that course. However, when applied in the context of a heuristic evaluation, the QM materials should only be used to evaluate prototypes in the interest of making improvements and not for establishing a quality benchmark for a finalized course. A QM-guided heuristic evaluation performed by a skilled evaluator can provide tremendously valuable insights along the dimensions of learner experience outlined above. These can serve as the basis for subsequent design refinements to an online course, which promotes a more positive overall learner experience.

4.6. A/B Testing

A/B testing or split-testing compares two versions of a user interface and, because of this, all three prototyping phases can employ this method. The different interface versions might vary individual screen elements (such as the color or size of a button), typeface used, placement of a text box, or overall general layout. During A/B testing, it is important that the two versions are tested at the same time by the same user. For instance, Version A can be a control and Version B should only have one variable that is different (e.g., navigation structure). A randomized assignment, in which some participants receive Version A first and then Version B (versus receiving Version B and then Version A), should be used.

Learning designers do not frequently have access to large numbers of learners for A/B testing, and therefore need to consider how to adapt this approach to specific design contexts. For example, a design team building a case library for a case-based learning environment is struggling with the design of the cases themselves. One learning designer has created a set of cases that highlight the central theme of the different cases but are fairly text heavy. Another learning designer has taken a different design approach and created a comic-book layout for the cases, which has visual appeal, but the central theme of the cases is not emphasized. The design team asks six students to review the designs. Three students review the more thematically-focused cases and three review the comic-book cases. The students are then asked to create a concept map that shows the central themes of the cases and how those themes are connected. The design team learns that students who used the thematically-focused cases spent much less time reviewing the cases and their concept maps show a very shallow understanding of the topic, although they did appropriately identify thematic areas. The students who used the comic-book cases spent more time reviewing the cases, and their concept maps are richer and show a more nuanced understanding of the topic, despite missing the specific names of the thematic areas (although they describe the areas in their own words). With this information, the team decides to continue iterating prototypes of the comic-book design while better emphasizing the central themes within those cases. On this basis, a potentially more effective learner experience was uncovered.

4.7. Think-Aloud User Study

Unlike A/B testing, a think-aloud user study is only used during the functional prototyping phase. According to Jakob Nielsen (1993), "thinking aloud may be the single most valuable usability engineering method" (p. 195). In a think-aloud user study, a single participant is tested at any given time. The participant narrates what he or she is doing, feeling, and thinking while looking at a prototype (or fully functional system) or completing a task. This method can seem unnatural for participants, so it is important for the researcher to encourage the participant to continue verbalizing throughout a study session. To view an example of a think-aloud user study, please watch [Steve Krug's "Rocket Surgery Made Easy" video](#).

A great deal of valuable data can come from a think-aloud user study (Krug, 2010). Sometimes participants will mention things they liked or disliked about a user interface. This is important to capture because it may not be discovered in other methods. However, the researcher needs to also be cautious about changing an interface based on a single comment.

Users do not necessarily have to think-aloud while they are using the system. The retrospective think-aloud is an alternative approach that allows a participant to review the recorded testing session and talk to the researcher about what he or she was thinking during the process. This approach can provide additional helpful information, although it

may be difficult for some participants to remember what they were thinking after some time. Hence, it is important to conduct retrospective think-aloud user testing as soon after a recorded testing session as possible.

Think-aloud user testing is the most widely used method of usability evaluation in practice, including in the field of LIDT. Indeed, usability testing has long been recognized as a useful evaluation method in the design of interactive learning systems (cf. Reeves & Hedberg, 2003). Increasingly, usability testing is gaining acceptance in LIDT as a viable and valuable evaluation method for informing research related to advanced or novel learning technologies, for which existing research is neither substantial nor sufficient, such as 360-video based virtual reality (Schmidt et al., 2019) or digital badging (Stefaniak & Carey, 2019). Given the limited resources provided to learning designers, think-aloud user testing is particularly attractive because it can be conducted with relatively small numbers of participants (often only five participants are needed to assess the usability of an online course) and with open source or free-to-use tools. For a primer on how to conduct think-aloud user testing, readers are referred to the U.S. government's online resources for usability at https://www.usability.gov.

4.8. Eye-Tracking

Similar to the think-aloud user study, eye-tracking is an evaluation method that involves the user during the functional prototype phase. Eye-tracking is a psychophysiological method used to measure a participant's physical gaze behavior in responses to stimuli. Instead of relying on self-reported information from a user, these types of methods look at direct, objective measurements in the form of gaze behavior. Eye-tracking measures saccades, eye movements from one point to another, and fixations, areas where the participant stops to gaze at something. Saccades and fixations can be used to create heat maps and gaze plots, as shown in Figures 6-8, or for more sophisticated statistical analysis.
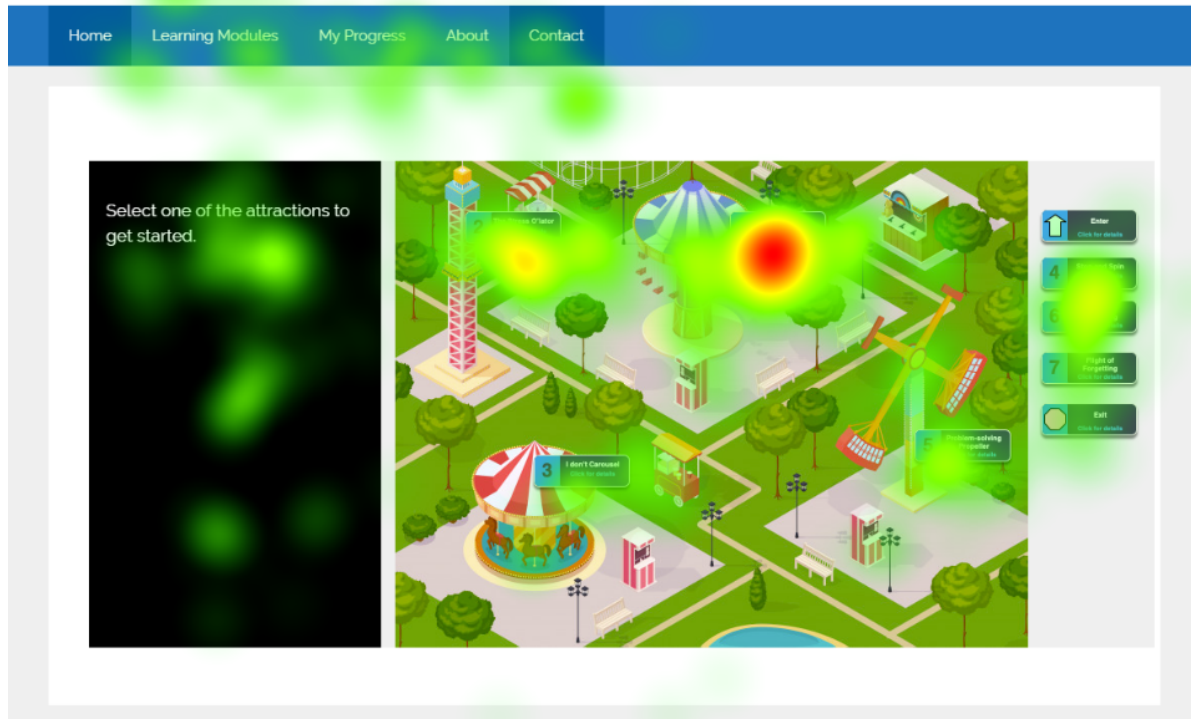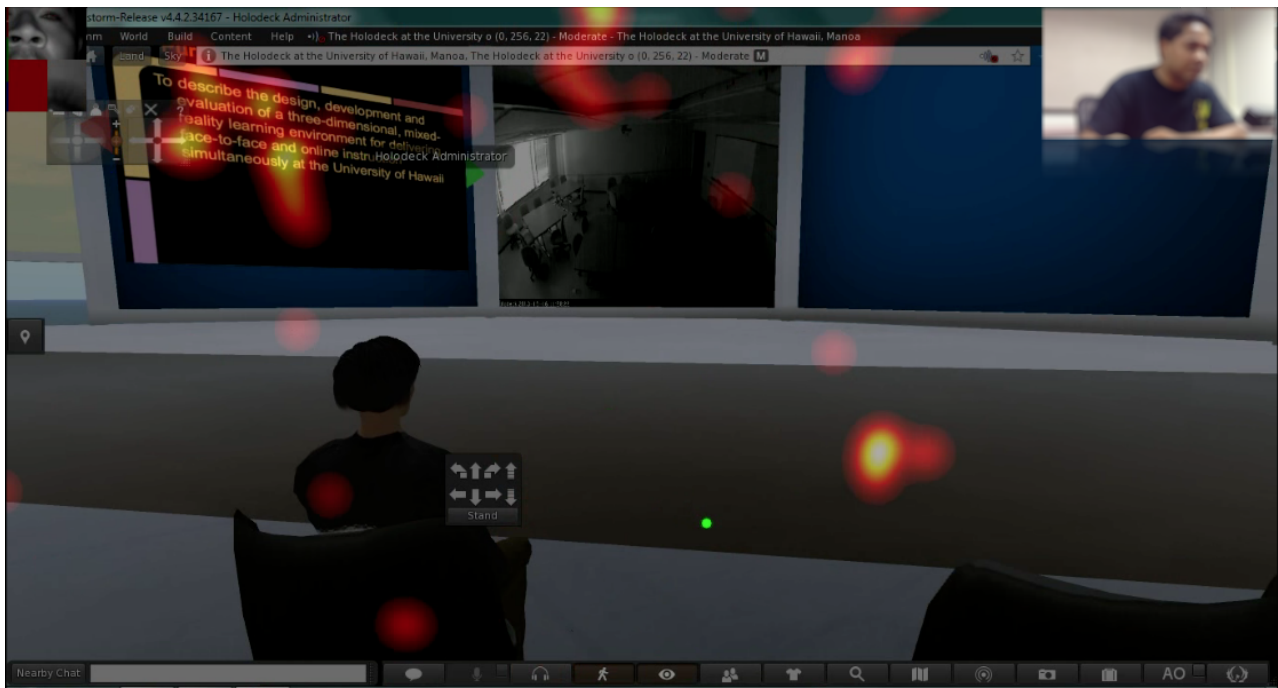
Figure 6

Heat Map of a Functional Prototype's Interface Designed to Help Learners With Type 1 Diabetes Learn to Better Manage Their Insulin Adherence

Note. Eye fixations are shown with red indicating longer dwell time and green indicating shorter dwell time. Photo courtesy of the Advanced Learning Technologies Studio at the University of Florida. Used with permission.

Figure 7

Heat Map of a Three-Dimensional Interface Showing Eye Fixations and Saccades in Real-Time, With Yellow Indicating Longer Dwell Time and Red Indicating Shorter Dwell Time

Note. Adapted from Schmidt et al. (2013). Reprinted with permission.
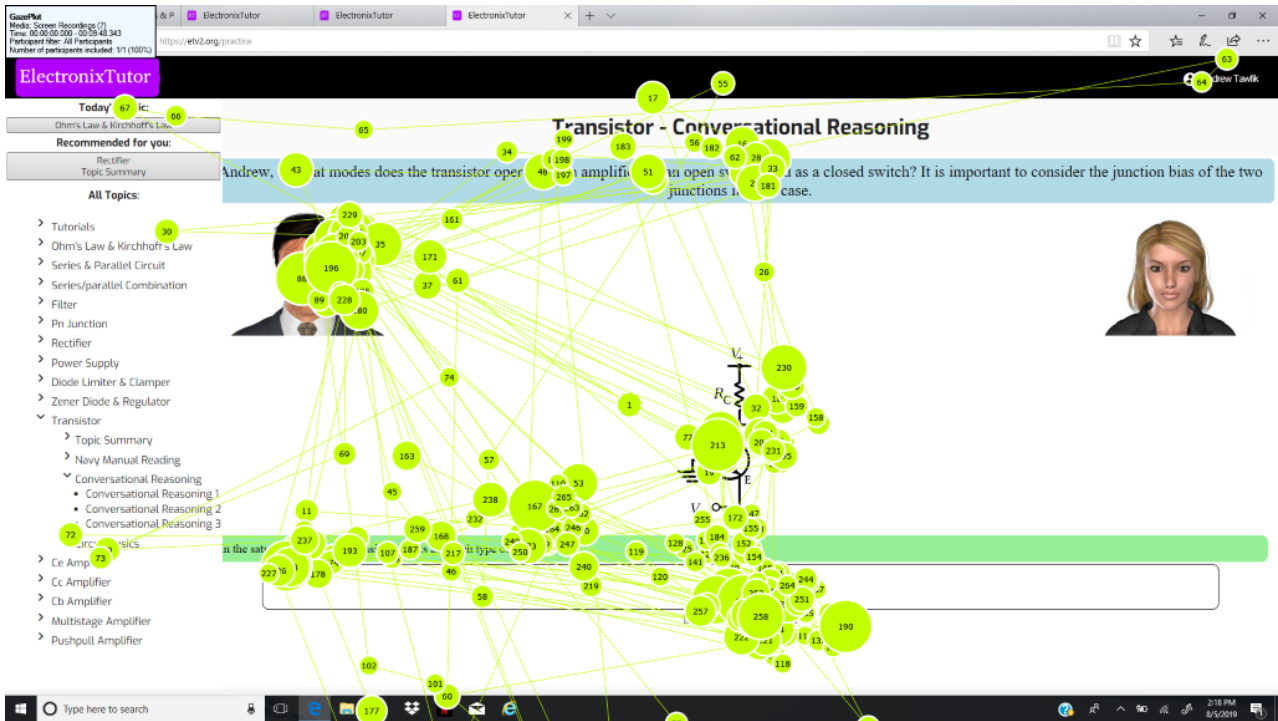
Figure 8

Gaze Plot of a Learner Engaged With the ElectronixTutor Learning Environment

Note. Photo courtesy of the Instructional Design Studio at the University of Memphis. Used with permission.

4.9. Electroencephalogy

Another psychophysiological method used to directly observe participant behavior is electroencephalogy (EEG). EEG measures participant responses to stimuli in the form of electrical activity in the brain. An EEG records changes in the brain's electrical signals in real-time. A participant wears a skull cap (Figure 9) with tiny electrodes attached to it. While viewing a prototype, EEG data such as illustrated in Figure 10 can show when a participant is frustrated or confused with the user interface (Bergstrom et al., 2014).

From the perspective of learning design, eye-tracking and EEG-based user testing are typically reserved for very large training programs (i.e., for large corporations like Apple or Facebook) or for learning designs that are more focused on research than on practical application. It is not very common for small learning design teams to have access to EEG and eye tracking resources. Nonetheless, these approaches can serve as a way to understand when learners find something important, distracting, disturbing, etc., thereby informing learning designers of factors that can impact extraneous cognitive load, arousal, stress, and other factors relevant to learning and cognition. A disadvantage of this type of data, for example, is that it might not be clear why a learner was fixated on a search field, why a learner showed evidence of stress when viewing a flower, or if a fixation on a 3D model of an isotope suggests learner interest or confusion. In these situations, a retrospective think-aloud can be beneficial. After the eye-tracking data have been collected, the learning designer can sit down with a participant and review the eye-tracking data while asking about eye movements and particular focus areas.
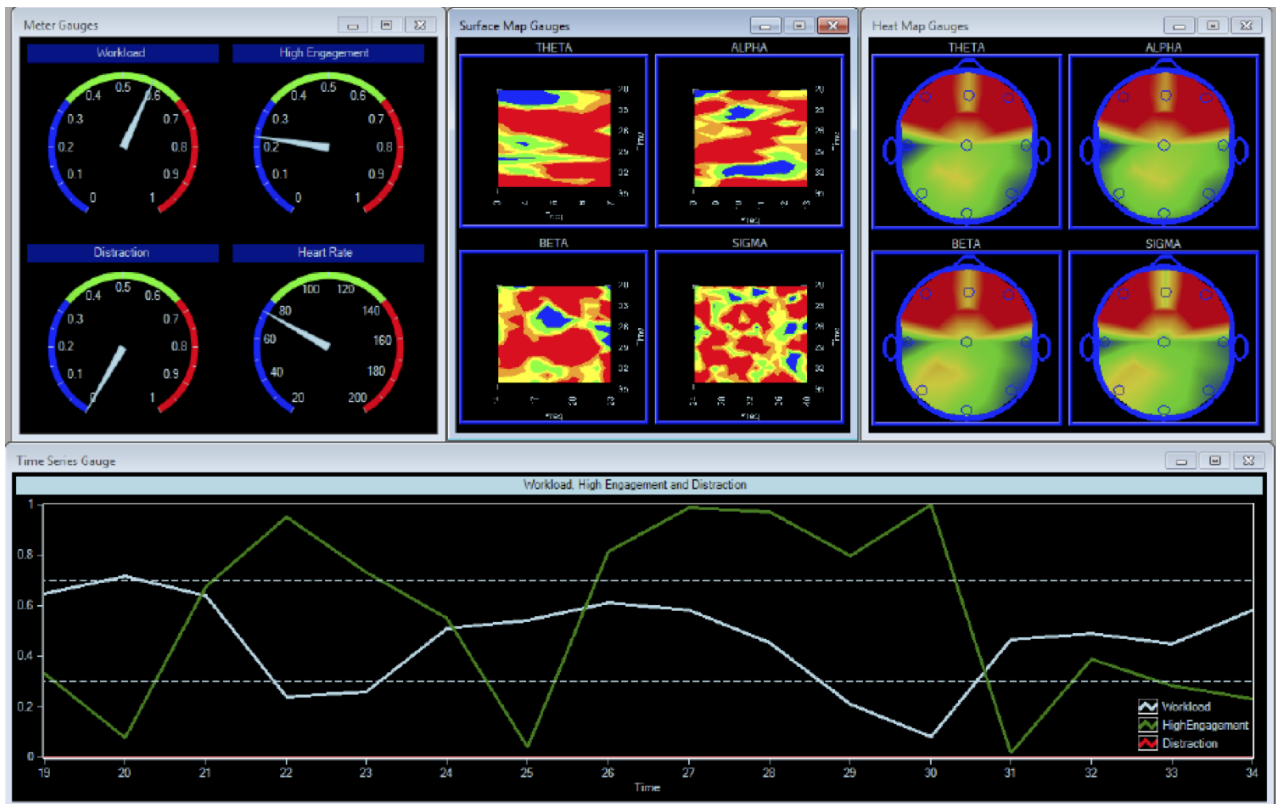

Figure 9

A Research Study Participant Wears an EEG While Viewing an Interface

Note. Photo courtesy of the Neuroscience Applications for Learning (NeurAL Lab) at the University of Florida. Used with permission.
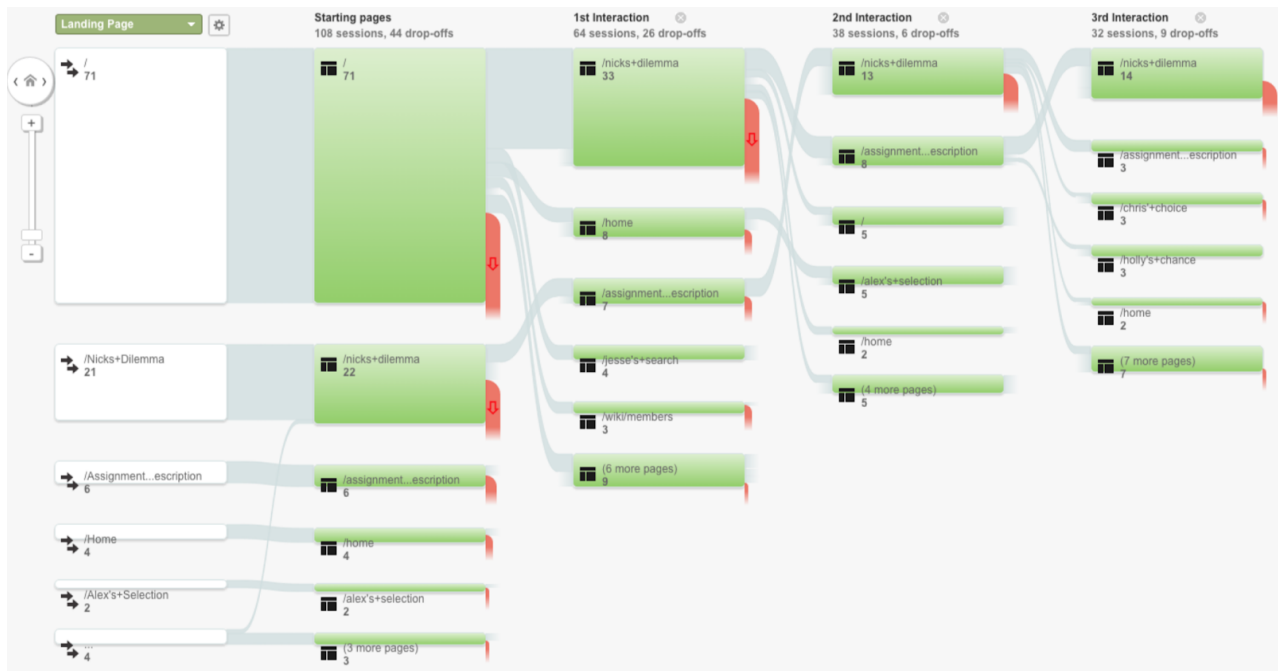
Figure 10

Output From EEG Device in a Data Dashboard Displaying a Variety of Psychophysiological Measures (e.g., Workload, Engagement, Distraction, Heart Rate)

Note. Photo courtesy of the Neuroscience Applications for Learning (NeurAL Lab) at the University of Florida. Used with permission.

4.10. Analytics

A type of evaluation method that is gaining significant traction in the field of learning design due to advances in machine learning and data science is analytics. Analytics are typically collected automatically in the background while a user is interfacing with a system and sometimes without the user even being aware the data are being collected. An example of analytics data is a clickstream analysis in which the participants' clicks are captured while browsing the web or using a software application (see Figure 11). This information can be beneficial because it can show the researcher the path the participant was taking while navigating a system. Typically, these data need to be triangulated with other data sources to paint a broader picture.

Figure 11

An Example of a Clickstream, Showing Users' Paths Through a System

Note. Adapted from Schmidt & Tawfik (2019). Reprinted with permission.

Increasingly, learning analytics and data dashboards are being incorporated into the tools of the learning design trade, including LMSs, video conferencing suites, video hosting providers, and a myriad of others. Indeed, the massive collection of learners' personal usage data has become so ubiquitous that it is taken for granted. However, analytics and data dashboards remain novel tools that learning designers do not necessarily have the training to use for making data-based decisions for improving learning designs. That said, data dashboards are maturing quickly. Less than a decade ago, only the most elite learning designers could incorporate learning analytics and data dashboards into their designs, whereas today these tools are built-in to most tools. Clearly, these tools have enormous potential for the field of LIDT, for example, for creating personalized learning environments, providing individualized feedback, improving motivation, and so-on. With advances in machine learning and artificial intelligence, learning analytics hold great promise; however, privacy concerns, questions of who owns and controls learner data, and other issues remain. Learning designers are encouraged to carefully review the data usage agreements of the software used for developing and deploying digital environments for learning. As mentioned previously in this chapter, LX considers the entire experience of the learner when using a technology, which includes their experiences with the collection of personal data. Carefully safeguarding this data and using it judiciously is paramount for a positive learner experience.

5. Conclusion

As digital tools for learning have gained in popularity, there is a rich body of literature that has focused on designing for learning with and through these tools. Indeed, a variety of principles and theories (e.g., cognitive load theory, distributed cognition, activity theory) provide valuable insight to situate the learning design process. While the design of learning technologies is not new, issues of how learners interact with the technology can sometimes become secondary to pedagogical concerns.

In this chapter, we have illustrated how the field of HCI intersects with the field of instructional design and provided specific examples of how to approach learning design using methods and processes commonly associated with UCD. Moreover, we have provided examples of iterative design processes and commonly used evaluation methodologies that

can be employed to advance usable and pleasing learning designs, along with illustrative examples of how these methods and processes can be used in practice. The concepts of HCI, UX, and UCD provide insight into how learning technologies are used by educators and learners. A design approach that connects the principles of UCD with theories and processes of learning design can help ensure that digital environments for learning are constructed in ways that best support learners' achievement of their learning goals.

References

Ackerman, M. S. (2000). The intellectual challenge of CSCW: The gap between social requirements and technical feasibility. *Human-Computer Interaction, 15*, 179–203.

Allen, M. (2014). *Leaving ADDIE for SAM: An agile model for developing the best learning experiences*. American Society for Training and Development.

Barab, S., Barnett, M., Yamagata-Lynch, L., Squire, K., & Keating, T. (2002). Using activity theory to understand the systemic tensions characterizing a technology-rich introductory astronomy course. *Mind, Culture, and Activity, 9*(2), 76–107.

Boland, R. J., Tenkasi, R. V., & Te'eni, D. (1994). Designing information technology to support distributed cognition. *Organization Science, 5*(3), 456–475.

Carroll, J. M. (2000). Introduction to this special issue on "Scenario-Based System Development." *Interacting with Computers, 13*(1), 41–42. https://doi.org/10/b446hz

Cooper, A. (2004). *The inmates are running the asylum: Why high-tech products drive us crazy and how to restore the sanity*. Sams Publishing.

Desrosier, J. (2011). Rapid prototyping reconsidered. *The Journal of Continuing Higher Education, 59*, 134–145.

Engeström, Y. (2000). Activity theory as a framework for analyzing and redesigning work. *Ergonomics, 43*(7), 960–974.

Goggins, S., Schmidt, M., Guajardo, J., & Moore, J. (2011). 3D virtual worlds: Assessing the experience and informing design. *International Journal of Social and Organizational Dynamics in Information Technology, 1*(1), 30–48.

Grudin, J. (1988). Why CSCW applications fail: Problems in the design and evaluation of organizational interfaces. In I. Greif (Ed.), *Proceedings of the 1988 ACM conference on computer-supported cooperative work* (pp. 85–93). ACM Press.

Hollan, J., Hutchins, E., & Kirsh, D. (2000). Distributed cognition: toward a new foundation for human-computer interaction research. *ACM Transactions on Computer Human Interaction, 7*(2), 174–196.

Hudson, W. (2012). Card sorting. In M. Soegaard, & R. F. Dam (Eds.), *The encyclopedia of human-computer interaction* (2nd ed.). https://edtechbooks.org/-FGmR

International Organization for Standardization. (2010). *Ergonomics of human-system interaction--Part 210: Human-centred design for interactive systems* (ISO Standard No. 9241). https://edtechbooks.org/-Zma

Jonassen, D., & Rohrer-Murphy, L. (1999). Activity theory as a framework for designing constructivist learning environments. *Educational Technology Research and Development, 47*(1), 61–79.

Jou, M., Tennyson, R. D., Wang, J., & Huang, S.-Y. (2016). A study on the usability of E-books and APP in engineering courses: A case study on mechanical drawing. *Computers & Education, 92*(Supplement C), 181–193.

Kaptelinin, V. (1996). Activity theory: Implications for human-computer interaction. In B.A. Nardi (Ed.), *Context and consciousness: Activity theory and human-computer interaction*. MIT Press.

Kaptelinin, V., Nardi, B., & Macaulay, C. (1999). The activity checklist: a tool for representing the "space" of context. *Interactions, 6*(4), 27–39.

Korbach, A., Brünken, R., & Park, B. (2017). Measurement of cognitive load in multimedia learning: a comparison of different objective measures. *Instructional Science, 45*(4), 515–536.

Krug, S. (2010). *Rocket surgery made easy: The do-it-yourself guide to finding and fixing usability problems*. New Riders.

Lewis, C., & Wharton, C. (1997). Cognitive walkthroughs. In M. Helander, T. K. Landauer, & P. Prabhu (Eds.), *Handbook of human-computer interaction* (2nd ed., pp. 717–732). Elsevier.

Mayer, R. E., & Moreno, R. (2003). Nine ways to reduce cognitive load in multimedia learning. *Educational Psychologist, 38*(1), 43–52.

Mehlenbacher, B., Bennett, L., Bird, T., Ivey, I., Lucas, J., Morton, J., & Whitman, L. (2005). Usable e-learning: A conceptual model for evaluation and design. *Proceedings of HCI international 2005: 11th international conference on human-computer interaction*, *volume 4 — Theories, models, and processes in HCI.* Lawrence Erlbaum Associates.

Michaelian, K., & Sutton, J. (2013). Distributed cognition and memory research: History and current directions. *Review of Philosophy and Psychology, 4*(1), 1–24.

Mortensen, D. H. (2020, June). *How to do a thematic analysis of user interviews*. Interaction Design Foundation. https://edtechbooks.org/-PRHn

Nardi, B. A. (1996). Studying context: A comparison of activity theory, situated action models, and distributed cognition. In B. A. Nardi (Ed.), *Context and consciousness: Activity theory and human-computer interaction* (pp. 69–102). MIT Press.

Nardi, B. (1997). The use of ethnographic methods in design and evaluation. In M. Helander, T. K. Landauer, & P. Prabhu (Eds.), *Handbook of human-computer interaction* (2nd ed., pp. 361–366). Elsevier.

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

Nielsen, J. (1994). Heuristic evaluation. In J. Nielsen & R. L. Mack (Eds.), *Usability inspection methods* (pp. 25–62). John Wiley & Sons.

Nielsen, J. (2012). *Usability 101: Introduction to usability*. Nielsen Norman Group. https://edtechbooks.org/-JMKq

Nielsen, J., & Molich, R. (1990). Heuristics evaluation of user interfaces. *CHI '90: Proceedings of the SIGCHI conference on human factors in computing systems* (pp. 249–256). ACM.

Nielsen, J., & Sano, D. (1995). SunWeb: User interface design for Sun Microsystem's internal Web. *Computer Networks and ISDN Systems, 28*(1), 179–188.

Norman, D. A. (1986). Cognitive engineering. In D. A. Norman & S. W. Draper (Eds.), *User-centered system design: New perspective on human computer interaction* (pp. 31–61). L. Erlbaum Associates.

Okumuş, S., Lewis, L., Wiebe, E., & Hollebrands, K. (2016). Utility and usability as factors influencing teacher decisions about software integration. *Educational Technology Research and Development, 64*(6), 1227–1249.

Orlikowski, W. J. (1996). Improvising organizational transformation over time: A situated change perspective. *Information Systems Research, 7*(1), 63–92.

Paas, F., & Ayres, P. (2014). Cognitive load theory: A broader view on the role of memory in learning and education. *Educational Psychology Review, 26*(2), 191–195.

Paas, F., & Renkl, A., & Sweller, J. (2003). Cognitive load theory and instructional design: Recent developments. *Educational Psychologist, 38*, 1–4.

Reeves, T. C., Benson, L., Elliott, D., Grant, M., Holschuh, D., Kim, B., Kim, H., Lauber, Erick & Loh, S. (2002). Usability and instructional design heuristics for e-learning evaluation. *ED-MEDIA 2002 world conference on educational multimedia, hypermedia & telecommunications proceedings* (pp. 1653–1659). Association for the Advancement of Computing in Education.

Reeves, T. C., & Hedberg, J. G. (2003). *Interactive learning systems evaluation*. Educational Technology Publications.

Righi, C., James, J., Beasley, M., Day, D. L., Fox, J. E., Gieber, J., Howe, C. & Ruby, L. (2013). Card sort analysis best practices. *Journal of Usability Studies, 8*(3), 69–89.

Rodríguez, G., Pérez, J., Cueva, S., & Torres, R. (2017). A framework for improving web accessibility and usability of Open CourseWare sites. *Computers & Education, 109*(Supplement C), 197–215.

Rogers, Y. (2012). HCI theory: Classical, modern, and contemporary. *Synthesis Lectures on Human-Centered Informatics, 5*(2), 1–129. https://edtechbooks.org/-VBxB

Romano Bergstrom, J. C., Duda, S., Hawkins, D., & McGill, M. (2014). Physiological response measurements. In J. Romano Bergstrom & A. Schall (Eds.), *Eye tracking in user experience design* (pp. 81–110). Morgan Kaufmann.

Rossett, A. (1987). *Training needs assessment*. Educational Technology Publications.

Rossett, A., & Sheldon, K. (2001). *Beyond the podium: Delivering training and performance to a digital world*. Jossey-Bass/Pfeiffer.

Schmidt, M., Kevan, J., McKimmy, P., & Fabel, S. (2013). The best way to predict the future is to create it: Introducing the Holodeck@UH mixed-reality teaching and learning environment. *36th annual proceedings: Selected papers on the practice of education communications and technology - Volume 2* (pp. 578–588). Research and Theory Division of AECT.

Schmidt, M., Schmidt, C., Glaser, N., Beck, D., Lim, M., & Palmer, H. (2019). Evaluation of a spherical video-based virtual reality intervention designed to teach adaptive skills for adults with autism: A preliminary report. *Interactive Learning Environments,* 1–20. https://doi.org/10/gf8rmf

Schmidt, M. & Tawfik, A. (2017). Transforming a problem-based case library through learning analytics and gaming principles: An educational design research approach. *Interdisciplinary Journal of Problem-Based Learning 12*(1), Article 5.

Sleezer, C. M., Russ-Eft, D. F., & Gupta, K. (2014). *A practical guide to needs assessment* (3rd ed.). Pfeiffer.

Snyder, C. (2003). *Paper prototyping: The fast and easy way to design and refine user interfaces*. Morgan Kaufmann.

Stefaniak, J., & Carey, K. (2019). Instilling purpose and value in the implementation of digital badges in higher education. *International Journal of Educational Technology in Higher Education*, (16), Article 44. https://doi.org/10/ggwnbj

Straub, E. T. (2017). Understanding technology adoption: Theory and future directions for informal learning. *Review of Educational Research*, *79*(2), 625–649.

Sweller, J., van Merriënboer, J. J. G., & Paas, F. G. W. C. (1998). Cognitive architecture and instructional design. *Educational Psychology Review*, (10), 251–296.

Tripp, S. D., & Bichelmeyer, B. (1990). Rapid prototyping: An alternative instructional design strategy. *Educational Technology Research and Development*, *38*(1), 31–44.

Tullis, T. S. (1985). Designing a menu-based interface to an operating system. In L. Borman and B. Curtis (Eds.), *Proceedings of the ACM CHI 85 human factors in computing systems conference* (pp. 79–84). ACM.

UsabilityTest (2006). *Paper prototyping.* https://edtechbooks.org/-WFzk

van Merriënboer, J. J. G., & Ayres, P. (2005). Research on cognitive load theory and its design implications for e-learning. *Educational Technology Research and Development*, *53*(3), 5–13.

Vasiliou, C., Ioannou, A., & Zaphiris, P. (2014). Understanding collaborative learning activities in an information ecology: A distributed cognition account. *Computers in Human Behavior, 41*(Supplement C), 544–553.

Walker, M., Takayama, L. & Landay, J.A. (2002). High-fidelity or low-fidelity, paper or computer? Choosing attributes when testing web prototypes. *Proceedings of the human factors and ergonomics society annual meeting, 46*(5), 661–665.

Wharton, C., Rieman, J., Lewis, C., & Polson, P. (1994). The cognitive walkthrough method: A practitioner's guide. In J. Nielsen & R. L. Mack (Eds.), *Usability inspection methods* (pp. 105–140). John Wiley & Sons.

Wilson, B. G., Jonassen, D. H., & Cole, P. (1993). Cognitive approaches to instructional design. In G. M. Piskurich (Ed.), *The ASTD handbook of instructional technology* (pp. 11.1–21.22). McGraw-Hill.

Wood, J. R., & Wood, L. E. (2008). Card sorting: Current practice and beyond. *Journal of Usability Studies, 4*(1), 1–6.

Yamagata–Lynch, L. C., Cowan, J., & Luetkehans, L. M. (2015). Transforming disruptive technology into sustainable technology: Understanding the front-end design of an online program at a brick-and-mortar university. *The Internet and Higher Education, 26*(Supplement C), 10–18.

Matthew Schmidt

University of Florida

Matthew Schmidt, Ph.D., is Associate Professor at the University of Florida (UF) in the Educational Technology program, where he also directs the Advanced Learning Technologies Studio (ALT Studio). His research interests include design and development of innovative educational courseware and computer software with a particular focus on individuals with disabilities and their families/caregivers, virtual reality and educational gaming, and learning experience design.
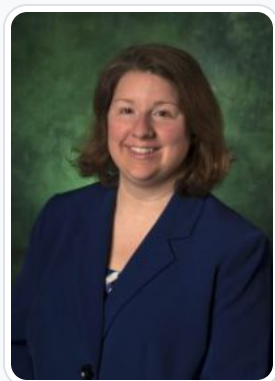
Yvonne Earnshaw

Kennesaw State University

Yvonne Earnshaw, PhD is an Assistant Professor of Instructional Design and Technology in the School of Instructional Technology and Innovation at Kennesaw State University. Dr. Earnshaw has an extensive industry background in technical writing, instructional design, and usability consulting. Her research interests include user/learner experience, online teaching and learning, and workplace preparation.

Andrew A. Tawfik

University of Memphis

Andrew A. Tawfik, Ph.D., is an Associate Professor of Instructional Design & Technology at the University of Memphis. Dr. Tawfik also serves as the the director of the Instructional Design & Technology studio at the University of Memphis. His research interests include problem-based learning, case-based reasoning, usability, and computer supported collaborative learning.

Isa Jahnke

University of Technology Nuremberg

Isa Jahnke, Ph.D., is Founding Vice President for Academic and International Affairs (digital learning) and Full Professor at the University of Technology Nuremberg. Past 6 years, she was Associate Professor at the University of Missouri's iSchool, and Director of the Information Experience Lab, a usability and user experience research, service and educational lab (2015-2021). She was Professor at Umeå University in Sweden (2011-2015) and Assistant Professor at TU Dortmund university in Germany (2008-2011) . Her expertise focuses on digital learning, sociotechnical-pedagogical integration for learning and work processes. Her work contributes to an understanding and development of teaching and learning designs-in-practices, and creative and meaningful learning experiences with digital technologies. Further information and list of publications can be found here: http://www.isa-jahnke.com