Synechron

Designing for complex systems:

Traditional UX research methods are not enough

Authored by:

Taryn Subramoney
Senior User Experience Designer
Surrey, UK

Traditional UX research methods such as interviews and contextual inquiries are invaluable tools for designers. However, when it comes to researching users who work on domain specific complex applications, this is not the case! We need to be flexible and adapt these methods to uncover notable insights. A few simple tweaks to traditional UX research approaches will result in gaining the insights we require to reduce the learning curve and improve the speed and accuracy of users of complex systems.

What is meant by a complex system?

A complex system is any application supporting the broad, unstructured goals or nonlinear workflows of highly trained users in specialized domains. Complex apps certainly vary in the type of workflows and end users they support — from research scientists to military professionals to financial analysts, for example. But they often share similar qualities. For example, complex apps frequently:

- **Support highly trained users** with specialized knowledge
- **Help users navigate and manage** large underlying data sets and enable advanced sensemaking or data analysis
- **Support problem solving** or end goals with unknown or variable underlying tasks
- **Require handoff or collaboration** among multiple roles, tools, or platforms
- **Mitigate the risks** of executing high-impact (or high-value) tasks, where high loss (e.g., revenue or even lives) is at stake

Why traditional UX research methods don't work for complex systems

Before we even start the research process, we need to first critically analyze our assumptions as designers, and ensure that they are appropriate for the demands of complex systems. Furthermore, traditional UX research methods are indispensable when doing research on generalist apps for everyday domains.

However, when it comes to complex systems, interviews for example, fall short since skilled behaviors are automated and not easily verbalized. Lastly, as designers we lack expert domain knowledge to fully understand the intricate nuances involved in complex problem solving. We need to be able to bridge this gap in order to successfully research a complex system.

How to adapt traditional UX methods for researching complex systems

Here are three ways to adapt your UX research approach for analyzing complex systems:

- Adapt your mindset as a designer
- Employ a strategic direction when collecting and interpreting data on observed work
- Partner with domain experts

Adapting your mindset

Designers rarely examine their assumptions. In fact, many may go years without believing they have reason to reflect on them. However, when teams apply assumptions about well-structured tasks to applications for users' complex analytical investigations these assumptions often undercut usefulness.

Here is a list of new assumptions that need to replace the old:

Parts to whole

Old: The whole of users' work is equal to the sum of its parts.

New: Complex work is synergistic.

Design teams must not prematurely sever related patterns from one another or examine sole actions in task landscapes in isolation. Relationships - the connections between procedural, spatial, and structural elements - are what give users' work coherence.

Who leads and who follows?

Old: Users must adapt to the logic of a program to improve their task performance.

New: Programs have to move closer to users and adapt to them.

In contrast to users adapting to the systems constraints, user adaptability is the prime emphasis. User adaptability means that users have a reasonable way in an application to initiate any kind of intelligent user support; decide on its type, amount, and point in time of initiation; and refine and adjust it.

Customized tools - yes, or no?

Old: Systems are more versatile when they are generic.

New: Systems are more adaptable when they are domain specific.

Domain-specific systems represent professionals' patterns of inquiry and helps them maintain a connection between their goals and available actions. Domain-based applications for complex problem solving also know when to leave the way open to provide optimal user control and adaptability. In addition, designs centered on domains rely on a familiar, domain- based vocabulary.

Keep it simple and smart ('KISS')

Old: Support should simplify work.

New: Support should be simple to operate, but should promote intellectual nuance and sophistication.

For complex problem solving, simplicity means a program eases users' investigative burdens without reducing the layering and nuances of their work.

Dramatic interaction

Old: Problem solvers are fairly predictable rational agents.

New: Problem solvers are extemporaneous actors in an open and unfolding plot.

If designers hope to capture the nuances of people who tackle perplexing problems in their jobs amidst uncertain conditions and competing interests, they must actually see these people at work. Inquiries are structured by drama as much as by logic, and the best way for designers to get the drama right is to witness or be part of it.

Employing a strategic direction when collecting and interpreting data on observed work

As designers, we need to look for opportunities for improving users' work when analyzing and interpreting our research data. Identifying opportunities involves finding indications of lapses in experts' problem solving and decision-making. Many of these lapses are common to experts across domains.

In collecting and interpreting data on observed work, user-experience analysts should pose the following questions that highlight these common lapses:

- **Where do users fall back** on 'default' strategies and give them more weight than they deserve for a given situation?
- **Where do users flounder** in attempts to integrate and synthesize multiple factors?
- **Where do they neglect mundane information** that seems trivial in light of a single case but is important for many cases combined?
- **Where do they focus** on the wrong level of detail for a given purpose?
- **Where do they factor** the idiosyncrasies of a particular case into a solution and lose track of other variables?
- **Where do they decide** on a solution without justification?
- **Where do they fail** to transfer relevant abstract rules, principles, strategies, or 'automatic' procedures from one situation to another?
- **Where do they encounter** obstructions that keep them from verifying their intuitions and instead force them into making complex inferences?

Software solutions that address these lapses, depending on why they occur, may take the form of feedback, cues, workspace structures, reminders, more integrated functions, and easy access to data from divergent sources.

Findings can reveal how much control users need analytically, for what purpose, and when. Findings from observations need to highlight the points at which user freedom is pivotal to the success of complex problem solving and when it is not.

Why should you partner with domain experts?

There are certain challenges that come with working with complex systems. Specifically, learning domain-specific concepts and terms can be demanding. It is, therefore, critical to incorporate domain experts into your research process. You can leverage the knowledge from domain experts by holding interviews with them or even through co-design sessions. The key is to form long-term relationships with internal stakeholders to ensure both domain expertise and UX expertise are well integrated into the design lifecycle.

Conclusion:

Designing for complex systems requires a mindset and toolset shift to accurately capture the nuances in this area. By employing the strategies discussed in this article, you are on your way to creating useful solutions for users of complex systems.

References

1 https://www.nngroup.com/articles/complex-application-design/

2 https://www.youtube.com/watch?v=NHevdlqoOaQ

3 Source: Interaction Design for Complex Problem Solving. Developing Useful and Usable Software, book by Barbara Mirel

4 Source: Interaction Design for Complex Problem Solving. Developing Useful and Usable Software, book by Barbara Mirel

5 https://www.nngroup.com/videos/complex-app-101/

Authored by:

Taryn Subramoney
Senior User Experience Designer
Surrey, UK

To learn more about Taryn
see her profile

Reach out to:
taryn.subramoney@synechron.com

Synechron

www.synechron.com