Testing with users

Topics that we'll cover today:

- Congitive walkthroughs
- Usability testing (UT)
- User acceptance testing (UAT)

But first, the tl;dr

Cognitive walkthroughs help us quickly learn how usable a design or software is for a new or infrequent user who is not familiar with it

Usability testing helps us understand how the intended users of a product interact with it, and whether it helps them in the way they need it to

User acceptance testing helps us test a design's functionality against clearly defined criteria to see whether a feature is complete

Cognitive walkthrough help us quickly learn how usable a design or software is for a new or infrequent user who is not familiar with it

These help us learn about users in order to get clear about what features will produce value

Usability testing helps us understand how the intended users of a product interact with it, and whether it helps them in the way they need it to

User acceptance testing helps us test a design's functionality against clearly defined criteria to see whether a feature is complete

Cognitive walkthrough help us quickly learn how usable a design or software is for a new or infrequent user who is not familiar with it

Usability testing helps us understand how the intended users of a product interact with it, and whether it helps them in the way they need it to

User acceptance testing helps us test a design's functionality against clearly defined criteria to see whether a feature is complete

This helps us understand functionality so that we know whether a story or feature is complete

18F

Cognitive Walkthrough

We perform cognitive walkthroughs to get quick and early feedback on whether a design solution is easy for a new or infrequent user to learn, and why it is or isn’t easy.

This method is useful for catching big issues at any stage in the design process when you don’t have access to real users. It is not a substitute for usability testing.

How to do it

1. Identify a use case for a new or infrequent user of your design or product.
2. Recruit someone who is not familiar with your design and introduce that use case to them. Ask them what they expect will happen if they were to use your design.
3. Then, ask them to try to accomplish that task with your design. Watch.
4. As they go, let them lead and encourage them to think out loud and share how they make decisions.
5. Pay attention to how quickly/easily participants are able to pick up a task. Also pay attention to whether their expectations match up with their experience
6. Analyze the walkthrough results to highlight where the user struggled and what needs improvement.

Usability testing

Usability testing helps us understand how the intended users of a product interact with it, and whether it helps them in the way they need it to.

For usability testing, you'll want to recruit participants who represent the product's intended users. You'll also want to do prep work beforehand to guide your participant through relevant scenarios.

How to do it

1. Identify what specifically you want to learn about and then draft a series of questions or prompts that will ensure that you capture the data points you are interested in.
2. Recruit participants who represent the intended users of the design solution (5 participants are sufficient).
3. Ask them what they expect will happen if they were to use your design.
4. Then, ask them to try to accomplish that task with your design. Watch .
5. As they go, let them lead and encourage them to think out loud and share how they make decisions.
6. Observers should contribute to a rolling issues log. Engage your team in a post-interview debrief after each test.
7. Discuss the results. Schedule a 90-minute collaborative synthesis meeting to discuss issues you observed, and any questions these tests raise concerning user needs. Conclude the meeting by determining how the team will use what it learned in service of future design decisions.

User Acceptance Testing

When we write user stories, we include acceptance criteria.

UAT helps us know whether a user story's acceptance criteria are met and whether it can be moved to the done column.

For example, during a sprint, your vendor might take on a story:

“As an agency compliance officer, I want the online submission form for emissions data to require the user to digitally sign an attestation (i.e., "Click to sign"), in a accordance with the Electronic Signature Act 15 U.S.C. 7001, Public Law 105-277 the Uniform Electronic Transaction Act.”

18F

And the acceptance criteria for that story might be:

- "click to sign" button appears on screen
- After “click to sign” button is clicked, “are you sure” button appears on screen (to meet Electronic Signature Act requirement for signer’s consent to do business electronically)
- Form can only be submitted after “click to sign” and “are you sure” buttons are clicked
- E-signature is connected to submitted form
- User can easily download copy of the signed form

How to do it

1. Identify what user story or particular functionality you will test.
2. Use the existing acceptance criteria to define your test. If you don’t have acceptance criteria set, come together as a team and align around what those acceptance criteria should be before conducting your test. Then use those criteria to develop your testing plan.
3. Run the tests with your anticipated user. Observers should contribute to a rolling issues log if not all the criteria are being met.
4. If criteria are met, document test results, move story to done.

Synthesis

Whenever you conduct testing, afterward, come together as a team and synthesis the results. What are the patterns that surfaced? What do you believe these outcomes indicate? Align and pave your path forward together.

Discussion