

11. Usability Evaluation

In general it is difficult to access real users. It needs planning and sometimes incentives such as money to convince them to invest their time. Testing with real users might take 30-45 min. Hence, you need to be sure, your app is running and does what it should do, the users should not state obvious problems, you could have found and fixed already. Thus, start with methods that do not need real users and then continue with the methods with real users.

Every evaluation is a little research and should follow the steps and methods of research

1. Define your research question/goal.
2. Choose an appropriate method – know the strengths and limits of the methods, especially the chosen method, consider the criteria of good research (reliability, validity, ...).
 - [A Guide to Using User-Experience Research Methods](#)
3. Plan your research, make sure you will "measure" aspects of your research goal and no side effects dilute your findings.
4. Evaluate critically

11.1 Usability Inspection - no real users needed

There are two widely used and useful evaluation types, that do not require real users, but may be done by a member of the team – obviously, it would be better if a real usability expert outside the team would run the usability inspection.

11.1.1 Heuristic Evaluation

Check for [heuristics](#) and document how they are fulfilled or missed.

11.1.2 Cognitive Walkthrough

Basically, a usability expert emphasizes with a persona and goes through a list of tasks with the best way to solve the tasks and checks, if the user would do it in the suggested way. The expert should consider the questions

- "Will the user try and achieve the right outcome?"
- Will the user notice that the correct action is available to them?"
- Will the user associate the correct action with the outcome they expect to achieve?"
- If the correct action is performed, will the user see that progress is being made towards their intended outcome?"¹

"The focus of the cognitive walkthrough is on how easy the users will find it to learn, and how to use the system in an effective, efficient and satisfying way. It assesses each step a user is required to perform by the system in order to complete a task. Therefore, the role of the evaluator is to walkthrough each step in turn and assess whether it meets those users' needs."²


Example

Task scenario: Eva wants to log into the OBS-system.

Optimal Steps

- Open browser
- Navigate to site
- Click login button
- Enter the user name in the user name field
- Enter the password in the password field
- Click the login button

11.2 Usability Evaluation with real users

The most common known method is thinking aloud. However, as a first impression and to get quick feedback, you could start with  [Guerrilla Testing with Usability Cafe](#).

11.2.1 Thinking Aloud

Even if you do all the things mentioned before, a pilot test is mandatory. A pilot test gives you hints for

- ambiguous instructions
- unrealistic time estimates
- ambiguous task completion criteria
- misleading questionnaire questions
- dead battery in microphone
- bad order

Follow the steps below to run a thinking aloud evaluation

1. Develop the Test Plan
2. Select and Acquire Participants
3. Prepare Test Materials
4. Run a Pilot Test
5. Conduct the Real Test
6. Analysis and Final Report

11.2.1.1 Test Plan

First define a **task list**.

- Prioritize tasks by frequency and criticality.

- Choose most frequent and critical to test.
- For each task
 - Define any prerequisites.
 - Define successful completion criteria -- but not the click steps!
 - Specify maximum time to complete each task, after which help may be given.
 - Define what constitutes an error.
- Do not instruct the test user to return to the initial screen (home page) at the beginning of each task. If they do so of their own accord, that's fine.

11.2.1.2 Script

Introduce yourself and any observers by first name (no titles or job descriptions!).

"Hi, my name is Keith. I'll be working with you in today's session. [Frank and Thomas here will be observing]."

Explain that the purpose of the test is to collect input to help produce a better interface.

"We're here to test a new product, the Harmony 3D Information Landscape, and we'd like your help."

Emphasize that system is being tested not the user.

"I will ask you to perform some typical tasks with the system. Do your best, but don't be overly concerned with results -- the system is being tested, and not your performance."

Acknowledge software is new and may have problems.

"Since the system is a prototype, there are certainly numerous rough edges and bugs and things may not work exactly as you expect."

Do not mention any association you have with product (do mention if you are not associated with product).

"[I am an independent researcher hired to conduct this study, and have no affiliation with the system whatsoever]. My only role here today is to discover the flaws and advantages of this new system from your perspective. Don't act or say things based on what you think I might want to see or hear, I need to know what you really think."

Say user may ask questions at any time, but they may not be answered until after the test is completed.

"Please do ask questions at any time, but I may only answer them at the end of the session."

Explain any recording (reassure confidentiality).

"While you are working, I will be taking some notes. We will also be videotaping the session for the benefit of people who couldn't be here today. However, the material is used within our dev team only and everything will be anonymized."

Say user may stop at any time."If you feel uncomfortable, you may stop the test at any time. Do you have any questions?"

Ask users to tell you

- what they are trying to do
- things they read
- questions that arise in their mind
- things they find confusing
- decisions they make

Request questions be asked as they arise, but explain that you won't answer them until after the test. Provide concrete Tasks.

- Don't say "how would you do that..."
- Don't justify decisions in your design.
- If the user discovers something unusual or don't behave as expected – that is very valuable information that signals that the chosen design needs to be revised.
- Don't tell users where to click
- Say that it is ok if something breaks or stops
- It is not the users' fault
- Take notes of questions and discuss them after the thinking aloud test.
- Be careful when asking questions, they could influence your users.
- If the user stops talking aloud, encourage them to keep up the flowing commentary with neutral, unbiased prompts
 - non-committal "uh huh"
 - "Can you say more?"
 - "Please tell us what you are doing now?"
 - "I can't hear what you are saying"
 - "What are you thinking right now?"

Do not direct the user with specific questions like

- "Why did you do that?"
- "Why didn't you click here?"
- "What are you trying to decide between?"
- In general: Do not ask "why"-questions

After the test thank the user enthusiastically and ask them if they want to add or comment anything and if they are interested in the further progress of the product.

11.2.2 Questionnaires

If the users know your new features or app (e.g. alpha or beta testers, i.e. a privileged group of users with early access) or after a thinking aloud test, you could go for a questionnaire. Questionnaires give you quantitative comparable data, that could be analyzed statistically. The following questionnaires are widely used and some are available in different languages.

- SUS³ System Usability Scale
 - developed 1986, widely used, also in sciences
 - [practical guide and template](#)
- PSSUQ Post-Study System Usability Questionnaire⁴
 - developed 1995

- a small sample of 12 is sufficient⁵

Fraunhofer provides a good list of questionnaires, in German and [An Introduction to Usability Questionnaires](#), in English.

11.3 Report and Findings

The following metrics are useful to document your tests

- Completion rate
- **Time on task**
- Misclick rate
- Number of errors (running into the error criteria of a task)

Whatever method you use, you need to document the process and findings and rate its impact to decide how to continue with your app.

- [usability report template](#)

11.4 Norms

There are a couple of norms addressing usability: ISO 9241 family.

If you need to follow a more formal approach, see [Leitfaden Usability](#).

-
1. The Interaction Design Foundation. How to conduct a cognitive walkthrough. 2021. URL: <https://www.interaction-design.org/literature/article/how-to-conduct-a-cognitive-walkthrough> (visited on 16.02.2024). ←
 2. Christopher Reid Becker. *Learn human-computer interaction: Solve human problems and focus on rapid prototyping and validating solutions through user testing*. Packt, Birmingham and Mumbai, 2020. ISBN 9781838820329. URL: <https://learning.oreilly.com/library/view/learn-human-computer-interaction/9781838820329/>. ←
 3. John Brooke. Sus: a quick and dirty usability scale. *Usability Eval. Ind.*, 189; 11 1995. URL: https://www.researchgate.net/publication/228593520_SUS_A_quick_and_dirty_usability_scale. ←
 4. James Lewis and James R. Ibm computer usability satisfaction questionnaires: psychometric evaluation and instructions for use. *International Journal of Human-Computer Interaction*, 7:57–, 02 1995. URL: https://www.researchgate.net/publication/200085994_IBM_Computer_Usability_Satisfaction_Questionnaires_Psychometric_Evaluation_and_Instructions_for_Use, doi:10.1080/10447319509526110. ←
 5. Thomas Tullis and Jacqueline Stetson. A comparison of questionnaires for assessing website usability. 06 2006. URL: https://www.researchgate.net/publication/228609327_A_Comparison_of_Questionnaires_for_Assessing_Website_Usability. ←