□□ Link collection design, UI and user research

Accessibility

See our link collection on accessibility

Visual design

- Open Source Design, *Resources*: https://opensourcedesign.net/resources/
- Senongo Akpem, *Cross-Cultural Design*: https://abookapart.com/products/cross-cultural-design

Color

- Read:
  - Superbloom, *Don't Let Color Drown Out Your Message*: https://superbloom.design/learning/resources/color-message/
  - Danny Sapio, *10 principles for Color Usage in UI Design*: https://uxdesign.cc/10-principles-for-color-usage-in-ui-design-65174b213004
- Color palette tools:
  - *Popular color palettes*: https://coolors.co/palettes/popular
  - Color combinations: https://www.colourlovers.com/
  - Color picker with HTML Hex codes: htmlcolorcodes.com
- Color System: https://m3.material.io/styles/color/system/overview
- Use *Contrast Grid* for checking color contrast for accessibility: https://contrast-grid.eightshapes.com/

Fonts

- Sustainable Open Source Design (Podcast), *Jérémy Landes on Type Designing*: https://sosdesign.sustainoss.org/16
- Finding fonts
  - https://www.velvetyne.fr/about/

    - "Researching and disseminating typography and typeface creation"
  - https://www.fontsquirrel.com/
    - Font squirrel has a useful Open Font License filter option

Icons & Mascots

- Finding icons: thenounproject.com
- Graphéine, *A history of mascots*: https://www.grapheine.com/en/graphic-design-en/history-of-mascots

Branding

- Read:
  - Superbloom, *Open source & branding – a contradiction?* https://superbloom.design/learning/blog/branding-tails/
  - Superbloom, *How to name your app*: https://superbloom.design/learning/resources/how-to-name-your-app/
  - Superbloom, *Communications Toolkit for Civic & Nonprofit Tech*: https://superbloom.design/learning/resources/comms-toolkit/
  - New York Times, *The Weird Science of Naming New Products*: https://www.nytimes.com/2015/01/18/magazine/the-weird-science-of-naming-new-products.html
- Working with the community:
  - Open Food Network as an example of an OSS community doing a community led project regarding vision and mission: https://community.openfoodnetwork.org/t/ofn-organizational-vision-request-for-feedback/2090
  - An article on how Simple organised their open source brand design: https://medium.com/simple-dot-org/open-source-brand-design-for-simple-474f03a7fc15
- A list of helpful (OSS) brand standards/guidelines:
  - Red Hat brand standards: https://www.redhat.com/en/about/brand/standards
  - OpenSSF brand guidelines: https://openssf.org/about/brand-guidelines/
  - Beyond Gravity brand portal: https://brand.beyondgravity.com/d/jbxsqV491EeP
  - Sonos brand identity: https://www.sonos.com/en-us/blog/sonos-brand-design-refresh
  - OpenAI brand guidelines: https://openai.com/brand/

Logo design

- AntennaPod, *Getting to a new icon*: https://antennapod.org/it/blog/2020/06/getting-to-a-new-icon
  - A good example about how to write down your requirements
- Fordes, *Designing a logo for your software project*: https://www.fordes.de/posts/logodesign_softwareproject.html
  - A worked example for a logo creation process

Working with designers

Finding a designer

- Open Source Design Jobboard: https://opensourcedesign.net/jobs/
- User Experience & Discovery Lab: https://www.opentech.fund/labs/uxd-lab/
  - The lab provide free user-experience research and design support to internet freedom projects.
- Hylke Bons is designing app icons & icon sets - for free, financed by the community: https://planetpeanut.studio/services

Redesign & preparations for working with designers

- *Design Project Scoping Guide*: https://dschool.stanford.edu/resources/design-project-scoping-guide
- DesignKit, *Align on Your Impact Goals*: https://www.designkit.org/methods/align-on-your-impact-goals.html
- DesignKit, *Define Your Audience*: http://www.designkit.org/methods/define-your-audience.html

UX/UI design

Design systems and consolidation

- Figma, *Name and organize components*: https://help.figma.com/hc/en-us/articles/360038663994-Name-and-organize-component
- Example of a big team of OSS designers working on a single OSS tool: https://www.figma.com/design/GLg0zPNnwMEGIb9TytCrxQ/Resilience-App?node-id=1257-553&node-type=canvas
- *Design for Code* on Figma: https://www.figma.com/community/file/967069265207476318/design-for-code-config-2021

- FOSDEM talk on *Documenting a Design System. Lessons learned from open sourcing the Orbit docs*:

  https://archive.fosdem.org/2022/schedule/event/osd_lessons_learned_from_open_sourcing_the_orbit_docs/
- Joplin, *Design patterns, systems, guidelines*:

  https://notes.saneux.ei8fdb.org/shares/oSoPeKIFSz3SNyZ9c44gf1
- Women-Centric Design: https://www.womencentricdesign.org/

Usability principles and rules-of-thumb

- Jan Dittrich, Advice *I give when helping teams with UX*:

  https://www.fordes.de/posts/TipsUXSmallProjects.html
- J. Raskin, *Intuitive equals familiar*: https://www.asktog.com/papers/raskinintuit.html
  - “how to make something can be used as something that is familiar to users” can usually be answered more productively than "how to make this intuitive"
  - "*Don’t make me think*" (Steve Krug) is an excellent, short introduction:

    https://sensible.com/dont-make-me-think/
- AskTog, *First Principles of Interaction Design*: https://asktog.com/atc/principles-of-interaction-design/
- Icons and graphical control design:
  - Icons are often harder to understand than labels, see *Icon Usability*:

    https://www.nngroup.com/articles/icon-usability/
  - *Write on the button: the grammar of interactivity*:

    https://www.theguardian.com/info/developer-blog/2013/jan/02/interactive-button-text-grammar
  - *Progressive Disclosure* as a way to manage complexity:

    https://en.wikipedia.org/wiki/Disclosure_widget
    - The typical example is a print dialog: Show essentials (“Print”-button, Printer selection if multiple are connected) and offer detailed settings on demand (page margins, dpi selector , duplex etc.). The settings have “good” defaults, but they do not *need* to be changed to get the text printed on a page.
- Web form design:
  - *Website Forms Usability: Top 10 Recommendations*:

    https://www.nngroup.com/articles/web-form-design/
  - Luke Wroblewski, *Web Form Design. Filling in the blanks* (Thema):

    https://rosenfeldmedia.com/Themen/web-form-design/
  - Caroline Jarret/Gerry Gaffney, *Forms that Work. Designing Web Forms for Usability* (Thema): https://www.sciencedirect.com/Thema/9781558607101/forms-that-work
- *Macintosh Human Interface Guidelines*:

  http://interface.free.fr/Archives/Apple_HIGuidelines.pdf

    - The 1995 *Machintosh Interface Guidelines* do not help you to up-to-date interfaces, but contain a lot of useful principles.
- Divio, *The Documentation System*: https://docs.divio.com/documentation-system/

Tools

- *Penpot* is a FOSS-alternative for Figma, a UI design tool: https://penpot.app/
- Tools for wireframes: Excalidraw seems to be a favorite: https://excalidraw.com/

User research & journeys

- Visualize user activities over time with *user journey mapping*:
  https://www.usability.de/leistungen/methoden/user-journey-mapping.html
- Superbloom, *Qualitative Interviewing*:
  https://simplysecure.org/resources/qualitative_interviewing.pdf
- Jan Dittrich, *A Beginner's Guide to Finding User Needs*: https://urbook.fordes.de/
- *Empathy map*: Start with an assumption for user group/s and see if the empathy map indicates that your idea is going to support their goals or resolve their problems:
  https://commons.wikimedia.org/wiki/File:Empathy_map.jpg
- Superbloom, *Using Personas in Open-Source Projects*:
  https://superbloom.design/learning/blog/personas/

Evaluation methods

On finding out if your ideas work. Very often this is done too late – with paper prototyping you can run a simple evaluation within a few hours, so do test early and often!

Prototyping

You do not need an almost finished product to test (the earlier the better, actually): You can evaluate a single feature (see *tracer bullet development*), a mockup (see *penpot* or *figma*) or a *paper prototype*.

- *Tracer Bullet Development*: https://growsmethod.com/practices/TracerBullets.html
- Paper prototyping, a way to try ideas with users without needing to write any code.
    - *Paper prototyping* on Wikipedia: https://en.wikipedia.org/wiki/Paper_prototyping
    - Alistapart, *Paper Prototyping*: https://alistapart.com/article/paperprototyping/
- Concept prototype: Testing your idea when it's still at the conceptual stage: Keep it basic, keep it at the big picture level (so users won't focus on the finer details, like the color, design etc), use an exisiting product that is similar to get the users talking.

Analytics

- Privacy-preserving tracking of user actions: https://cleaninsights.org/
- Google analytics alternative: https://matomo.org/

Heuristic evaluation

Using rules to spot usability problems in a usage scenario; ideally done by several people.

- *10 Usability Heuristics for User Interface Design* the most common set of rules: https://www.nngroup.com/articles/ten-usability-heuristics/
- *How to conduct a heuristic evaluation*: https://www.nngroup.com/articles/how-to-conduct-a-heuristic-evaluation/

Usability tests & user testing

In a usability test, potential users try to use an early version of your product to find potential problems that you need to fix. User testing is not meant to sell the idea to the users, it's to gather data/feedback that supports you to continue product development, so use any resources available as early as possible to communicate the idea.

To read, listen and watch

- Superbloom, *How To Test Software For Usability And Usefulness When You Haven't Finished Writing The Code*: https://superbloom.design/learning/blog/how-to-test-software-for-usability-and-usefulness-when-you-havent-finished-writing-the-code/
- When is the right time to do user testing? As early as possible is often not ideal, since projects first need to figure out where they want to go. See Christoph Lattemann, Stefan Stieglitz, F*ramework for Governance in Open Source Communities*: https://publishup.uni-potsdam.de/opus4-ubp/frontdoor/deliver/index/docId/1824/file/stieglitz_lattemann_2005.pdf
- Before testing or having a software, maybe you want to document user context and processes: https://jdittrich.github.io/userNeedResearchBook/#coDocumenting
- *A Beginner's Guide to Finding User's needs* by Jan Dittrich: https://jdittrich.github.io/userNeedResearchBook/#dataGathering
- In general, it is better to do several smaller usability tests than a one or two big ones. Even small numbers of users can help to find problems. A classic article on this is *Why you only need to test with 5 users*: https://www.nngroup.com/articles/why-you-only-need-to-test-with-5-users/
- Know what power/significance are and how they are related to sample size, what you know from academia may be way too conservative for your needs. See also *Improving your statistical inferences* by Daniël Lakens: https://lakens.github.io/statistical_inferences/

- The Thema *Rocket surgery made easy* (Steve Krug) describes usability testing (and recruiting for- and communication of it): https://sensible.com/rocket-surgery-made-easy/
- Superbloom, *A Guide to Gathering Feedback*: https://superbloom.design/learning/blog/feedback-gathering-guide/
- Superbloom, *User Testing Cheat Sheet*: https://superbloom.design/learning/blog/user-testing-cheatsheet/
- *A Dev's guide to User testing*, made by Superbloom, has resources that can help OSS tool teams start to do user testing and usability testing in their OSS tool projects either with or without a designer helping out: https://sprblm.github.io/devs-guide-to/
- *Usability Testing can be fun! An FAQ for small open source teams*: https://usable.itch.io/user-testing-can-be-fun-a-guide-for-oss-developers-and-tool-teams-on-how-to-user
- User testing examples:
  - for the process: https://docs.ushahidi.com/platform-developer-documentation/dev-legacy-v3/design/user-testing-process
  - for design documentation: https://github.com/ushahidi/platform/issues/3681
  - Here is also an example video for a usability test: https://www.youtube.com/watch?v=QckIzHC99Xc

Methods & tools

- *Cardsort* helps UX researchers quickly analyse data from open card sorting exercises using hierarchical cluster analysis. This task helps to understand how people organize information, and is frequently used to develop information architectures for websites: https://pypi.org/project/cardsort/
- *CRC Cards* are a collaboration tool to find out which concepts the code needs and how these concepts should be defined: https://de.wikipedia.org/wiki/Class-Responsibility-Collaboration-Karten
- Role play workshop content for learning how to do user testing, for example through wireframing, *Illustrated Quick-Start Intro To Wireframing*: https://superbloom.design/learning/blog/wireframing-intro/
- The *System Usability Scale (SUS)* provides a “quick and dirty”, reliable tool for *measuring* the usability (measuring sounds cool, but it only makes sense if you have something to compare it to, like a previous measurement!): https://digital.gov/topics/usability/
- *The Rainbow Spreadsheet: A Collaborative Lean UX Research Tool*: https://www.smashingmagazine.com/2013/04/rainbow-spreadsheet-collaborative-ux-research-tool/
- Get in touch with why you're doing this via *Empathy map:* https://web.archive.org/web/20160711191804/http://thetoolkitproject.com/tool/empathy-

map

- *EventStorming*: Post it workshops for software development: https://www.eventstorming.com
- A method for beta testing documentation: *the highlighter test:* https://userresearch.blog.gov.uk/2014/09/02/a-simple-technique-for-evaluating-content/
- Writing *unit tests* can also help to divide your code in meaningful parts (Since big thing where everything ins somehow connected is hard to test: https://en.wikipedia.org/wiki/Unit_testing
  - Caveat: Some parts are easier to test, usually the ones that are not related to the user interface. Keep using tracer bullet dev!
- For making it possible to change the UI while keeping the rest constant or vice versa: Use the *command pattern*: https://refactoring.guru/design-patterns/command

Sourcing testing users

- You can get free access to an otherwise expensive suite of user testing tools through UserTesting.com's social impact program, *OneWorld*: https://www.usertesting.com/company/programs-initiatives/usertesting-oneworld
  - (Note for usertesting.com (and similar products): You won't get a lot of people from marginalized communities e.g. because payment needs a bank account or credit card etc.)
- "Citizen User Testing" groups concept, a formal model to identify more diverse feedback from communities you want to serve w/ your product or service- Examples and free e-Thema on the method: https://directory.civictech.guide/listing-category/user-research-groups
- The Mom Test - frequently cited Thema on this, essentially people will lie to you to be nice, how to talk to customers to get honest feedback (note: as a mom, I really hate this name. - Molly): https://www.momtestbook.com/

Revision #14
Created 30 August 2022 10:21:23 by Claudia
Updated 2 July 2026 09:06:17 by Marie-Lena Wiese