

USER: Executive Summary

The Usable Software Ecosystem Research project (USER) is a research initiative that explores how open source scientific and research software (SROSS) teams understand, consider, and undertake usability and design opportunities in their projects. Supported by the Sloan Foundation, our research followed an iterative Human-Centered Design approach using multiple methods of inquiry, including desk research, surveys, interviews, and observation at community events and gatherings between November 2022 and June 2023.

This project illuminated countless insights, from a close look into how a diverse set of SROSS projects and their contributors consider their users, to how they approach the usability and broader design aspects of their projects, to the complex and often contradictory ways in which they understand and build SROSS tools.

Below we've crafted a summary of our key findings, which, though significant, fails to contain the true breadth and complexity captured in our [complete set of findings](#). To understand these takeaways in context, please navigate to the linked associated chapters. You'll find after each finding there's suggested sections associated with the findings should you want to discover more about a particular finding and then see [associated recommendations](#) that we suggest implementing into your SROSS projects in order to address a challenge present in the finding.

Common language and respect

SROSS development often needs cross-disciplinary work. This can be challenging, as norms and common language need to be negotiated between different work cultures. In SROSS projects, there is often a disciplinary exchange between programmers and scientists. Add designers, and there is a third 'language' and working style. Effective cross-disciplinary collaboration ranges in strategy and process. Some projects prioritize mutual understanding and a state of continuous knowledge transfer between disciplines, while others take a more synergistic approach where each expert is able to excel in their domain without the burden of needing to know the 'other' discipline well. Regardless of the approach, mutual respect, communication and knowing the value of each role's contribution is vital to the success of design and usability in SROSS.

Find out more in sections: [Perceptions of design](#), [Governance](#), [Cross-disciplinary collaboration in SROSS](#), [Designers' experiences working with SROSS projects](#), [Design work in the academic context](#), [Design Challenges and Barriers](#).

Openness and values

SROSS projects users claim values like openness, access and progress of science. Though 'design' itself is not stated as a central value, SROSS has an idea that usability, accessibility

and user-centered improvements and practices enable the ethical values of understandable, reproducible and more robust scientific progress. The Open Source way of working is essential for the development and practice of better science.

Find out more in sections: [Feedback and knowing your users](#), [Usability](#), [Values and Attitudes of Open Science and Open Source](#), [Design Challenges and Barriers](#).

Openness, design and academia

Similarly to openness being an ethical development choice that SROSS builders believe makes for better science, they also believe openness challenges some of the difficult dynamics present in academia around ownership and competition for academic prowess. This doesn't erase these challenges, but pushes against the assumed opposition of the individual vs. the collective. Academic bureaucracy and norms can also impede work, especially community- or user-driven processes. Academia also rarely recognises design and usability work (as is often the case with code too) and roles and often will name them differently or leverage students to produce design work as an alternative to design practitioners.

Find out more in sections: [Values and Attitudes of Open Science and Open Source](#), [Design work in the academic context](#).

Resourcing Design and Usability

Non-designer participants found prioritizing design as a key project value difficult. The participants say they believe design will help, but how to make space for design work was unknown at best or scary at worst. Congruously, design and usability work is seen as less vital than the coding of new features and the fixing of bugs (even if design and usability improvements might solve these).

Find out more in sections: [Values and Attitudes of Open Science and Open Source](#), [Design Challenges and Barriers](#), [Prioritization](#), [Funding and Sustainability](#).

Implementing Design practices

Whether projects establish design and usability practices depends on the perceived utility of design work; what non-designers define as design, usability, and accessibility; and the ease of implementing design changes. If design work is seen as time intensive, disruptive, requiring code overhaul, and/or frivolous, the implementation of design practices is unlikely. Even when design and usability are valued, many see it as optional, incidental, or aspirational. In this case, design implementation is only 'worth it' when it is clearly linked to the expansion of a user base. We observed this most potently in projects with diverse users. SROSS projects express that they want to improve tools and services. Dedicated design and usability work could help doing this, but SROSS projects do not often realize this. Good design and usability practices empower

users, make tools easier to use, and require less immediate support from maintainers, which ultimately helps fulfill the purpose of research and scientific discovery.

Find out more in sections: [Design Challenges and Barriers](#), [Values and Attitudes of Open Science and Open Source](#), [Designers' experiences working with SROSS projects](#), [Accessibility](#), [Funding and Sustainability](#), [Design practices and workflows](#).

The way Accessibility and Usability are understood

Accessibility was understood as the 'findability' of a project and how the project could be made useful for different scientific fields. Some described accessibility in terms that design professionals do – the practice of ensuring that tools are inclusive for users with physical or cognitive differences – but this was more often expressed by designers in SROSS. Usability similarly had differing definitions from multiple perspectives and connected into a users 'skill level' towards how they use/operate the technical functions of the SROSS such as whether a scientist/researcher can understand and use a string of code in a terminal program to run a command that executes a function in the SROSS.

Find out more in sections: [Accessibility](#), [Usability](#), [Design Challenges and Barriers](#), [Perceptions of design](#), [Designers' experiences working with SROSS projects](#).

Ongoing communication as means to learn about users

SROSS projects incorporate their communities as a source of feedback for improvements. Participants reported that many user-focused practices – onboarding, feedback, training tutorials and support – are often fulfilled, in large part, by user communities themselves. Onboarding in particular is a responsibility that is passed off to communities of users. For instance, the burden of teaching a tool is often on professors, researchers, or students to teach each other. Tool feedback and troubleshooting – key components of user research – happens organically in user community channels such as Slack, Discourse, community calls, etc. Projects teams that are active and responsive in user channels are participating in informal user research, whether they recognize these user research methods or not.

Find out more in sections: [Onboarding](#), [Documentation](#), [Feedback and knowing your users](#).

The Designer perspective

SROSS designers reported feeling isolated – often, they are the only designer in a project, field of science, or who contribute to the project ecosystem, and they have few opportunities to advocate for the value of design in governance processes. Currently, there are not many designers working in SROSS, so there is a lack of peer support and SROSS design-related resources. Designers often need to justify the value of design to their fellow contributors and educate them about how design and usability work. Design and designers are rarely part of SROSS governance conversations. This makes it challenging for contributors unfamiliar with

design to embrace or learn about design practices, because it is not seen as foundational. There are few pathways for non-code contributions to projects, and project hierarchies tend not to favor the design contributions.

Find out more in sections: [Designers' experiences working with SROSS projects](#), [Perceptions of design](#), [Design Practices and Workflows](#), [Design work in the academic context](#), [Design Challenges and Barriers](#).

Recommendations

Below you will find the list of recommendations that the USER research team has compiled. Many of these recommendations are broad to allow for the implementation of them to be context specific and maintainable by the specific SROSS project or team that wishes to make changes. If you'd like to discuss specific ways to implement the recommendations please reach out by raising an issue in our [Github repository that is focussed around discussion here](#).

Supporting SROSS Design Practice

Promote design and usability as a practice that cuts labor and time for maintainers.

Usability improvements can reduce the time that maintainers, developers and community contributors spend writing step-by-step user guides and answering customer support requests when users are having difficulties with the tool.

Find out more in sections: [Design Challenges and Barriers](#), [Onboarding](#).

Make the entry points to contribution beyond code clear and inclusive.

Clear contribution paths and entry points allow potential contributors to understand where they are needed and how they can be most useful. Inclusivity increases contributions, because designers and other non-code contributors will see their work as valued and prioritized.

Find out more in sections: [Onboarding](#), [Designers' experiences working with SROSS projects](#), [Contributors and Contributions](#).

Develop resources, community spaces, shared channels, and information repositories for SROSS design.

If OSS/SROSS designers share experiences and learn from each other, they can help shape best practices and reduce barriers for designers new to the ecosystem. Also, including designers in OSS/SROSS maintainer channels, and vice-versa, will help increase designer/developer collaboration and provide guidance to projects without a designer on the team. Institutions can support this by promoting reuse of science and research and usability.

Find out more in sections: [Design Challenges and Barriers](#), [Governance](#).

Demonstrate via examples that many design and usability improvements can be small, iterative, and compartmentalized.

SROSS project teams limited by time and resources might assume design and UX improvements will be a large undertaking. In many cases, design and usability improvements do not require overhauling the code.

Find out more in sections: [Designers' experiences working with SROSS projects](#), [Design Challenges and Barriers](#), [Prioritization](#).

In contrast to the above, ensure a project has sufficient capacity to scope and support the work that the designer undertakes.

Design is a non-trivial practice and requires space, time, and resources. Design and usability improvements may require development time. Collaboration will increase efficiency and reduce bottlenecks.

Find out more in sections: [Prioritization](#), [Perceptions of Design](#).

Celebrate the value of design in SROSS with active acknowledgement, accolades, and references.

Open source culture celebrates code contributions regularly (merged code, closed issues). By celebrating design contributions, the community can show that design and usability is essential for project success. Designers will feel recognized, which will allow for retention and growth of the SROSS designer community.

Find out more in sections: [Governance](#), [Values and Attitudes of Open Science and Open Source](#), [Usability](#), [Contributors and Contributions](#).

Create and grow knowledge and training for designers interested in SROSS and Scientists and Researchers growing their Design knowledge.

The gaps in knowledge between design, OSS and science and research are variable and often large. Creating and maintaining a shared learning space for all functions and roles building SROSS will ensure skills are shared among roles.

Find out more in sections: [Usability](#), [Cross-disciplinary collaboration in SROSS](#), [Design Practices and Workflows](#).

Collaboration

Establish a common language around design and usability.

Different contributors will have different understandings of what 'design' and 'usability' mean. Agreeing at the outset what the design/usability needs are and how they will be tackled can help shape a common vocabulary. It helps create a baseline of knowledge sharing, mutual respect, and curiosity.

Find out more in sections: [Perception of design](#), [Usability](#), [Cross-disciplinary collaboration in SROSS](#), [Governance](#).

Spend dedicated time at the outset to get to know the project team's different disciplines and workflows.

Any given SROSS project can include maintainers from varying disciplines, perspectives, and skill sets. While each party does not need to understand their colleagues' work in expert-level depth, understanding each others' approaches will lead to more efficient work and successful collaboration.

Find out more in sections: [Design Practices and Workflows](#), [Design Challenges and Barriers](#), [Governance](#), [Prioritization](#).

Incorporate design and usability language and processes in product planning.

Outlining design and usability activities as a critical part of the plan from the outset would reduce friction later in the process. This can also help shift perceptions and support project leads and contributors prioritize UX/UI. Remember that developing user-friendly features can cut time spent on documentation.

Find out more in sections: [Cross-disciplinary collaboration in SROSS](#), [Prioritization](#), [Governance](#).

Support and include designers and usability experts in governance and standards discussions.

Giving designers a seat at the table will legitimize design/UX as a core component of SROSS success. Via these discussions, designers can help shift all SROSS practices to be more user-centered.

Find out more in sections: [Governance](#), [Designers' experiences working with SROSS projects](#), [Design Challenges and Barriers](#).

Foster community-driven accessibility practices by forming a working group or community of people with impairments.

Involving people with impairments in the building, maintaining, coding and design of SROSS helps shape more effective platforms and reduces needs for changes later. Forming a working group or community for those who identify as such would enable a stronger platform for advocacy of needs.

Find out more in sections: [Accessibility](#), [Governance](#), [Feedback and knowing your users](#)

Implementing User Experience and Design

Participate in existing user communities as a lightweight way to better understand users.

Active users and developers will often communicate via mailing lists, community forums, and open community calls. Observing or participating in existing community channels helps maintainers understand how the user community is using the software, what their needs are, and how to reach them.

Find out more in sections: [Feedback and Knowing your Users](#), [Design Practices and Workflows](#).

Use UX methods to represent a wider community of users.

Direct communication in projects often represents only the most active community members' needs. Beginners, occasional users, and users who do not have time and resources to participate are often not represented.

Find out more in sections: [Feedback and Knowing your Users](#), [Onboarding](#), [Accessibility](#).

Prioritize clear, accessible documentation and ensure its discoverability.

Navigable tutorials and documentation explaining features and related concepts helps users and contributors get started on their own, promotes adoption, and saves valuable time. Documentation is most effective when easy to find and applicable to a broad spectrum of users.

Find out more in sections: [Documentation](#), [Accessibility](#).

Incorporate user-centered design practices as part of onboarding.

Onboarding is key to project growth in terms of more users and contributors. Projects may grow beyond their initial use case, so capturing user needs as part of the onboarding process can help inform user-centered improvements.

Find out more in sections: [Onboarding](#), [Accessibility](#).

Create models out of projects that have adopted accessibility, design, and usability best practices.

Case studies of projects that incorporate best practices could inspire the projects that believe these activities are complicated. Open, clear, and thorough documentation allows other projects to borrow the pieces that would work for them.

Find out more in sections: [Accessibility](#), [Usability](#), [Design Practices and Workflows](#), [Designers' experiences working with SROSS projects](#), [Governance](#), [Funding and Sustainability](#).

Create standards for terminology and processes around design, usability, and user research that can be adapted by others.

Common processes and terminology on design, usability and user research can be included in governance documents. These could serve as templates to be loaded into a repository or copy and pasted as needed for other projects.

Find out more in sections: [Design Challenges and Barriers](#), [Values and Attitudes of Open Science and Open Source](#).