Self-Assessment Add-On:

Open and FAIR-R Research Software in Practice

A. FAIR Software Principles[1]

Use a publicly accessible repository with version control

Making code publicly available ensures transparency, reproducibility, and accessibility. Version control systems help track changes, support collaboration, and provide a clear history of the software's development.

		Yes	No
R1	Is the code hosted in a public repository?		
R2	Is version control used to track changes?		
R3	Are releases tagged clearly for citation/reuse?		

Add a license

Including a license clarifies how others can use, modify, and redistribute your software. Without a license, reuse is legally restricted, even if the repository is public.

		Yes	No
L1	Is a LICENSE file included with clear open-source terms?		

Register your code in a community registry

Registering your software in a community registry increases its visibility and discoverability, helping other researchers find and reuse it. Keeping the registry entry updated ensures users have access to the latest version and information.

		Yes	No
CR1	Is the software registered in a relevant community registry?		
CR2	Is the registry entry kept up to date?		

[1] Lamprecht A-L, Garcia L, Kuzak M, et al. Towards FAIR principles for research software. *Data Science*. 2019;3(1):37-59. doi:10.3233/DS-190026

Honors Disciplinary Project (July 2025) - This tool has been designed to complement Utrecht University's Ethics and Privacy Scan and support ethical software development in practice.

Enable citation of the software

Providing clear citation instructions ensures your work is credited when used by others. This helps track impact, supports academic recognition, and encourages proper reuse.

		Yes	No
C1	Is a CITATION.cff file or clear citation instructions provided?		
C2	Are citation instructions visible in the README?		

Use a software quality checklist

A quality checklist helps ensure that key aspects of good software development practices are followed. It can improve reliability, usability, and maintainability of the software.

		Yes	No
QC1	Is a software quality checklist used?		

B. Stakeholder Involvement

Without input from users and experts, software can become irrelevant, unusable, or even harmful. Feedback channels are critical for improvement and early issue detection.

		Yes	No
SI1	Were end users or experts involved in the design or testing?		
SI2	Is there a feedback or contribution channel?		

C. Responsibility & Sustainability

Lack of clear responsibility or planning risks abandonment, misuse, and loss of trust. Ongoing support and ethical foresight are essential for long-term impact.

		Yes	No
RS1	Is there a named contact for questions or concerns?		
RS2	Is there a plan for maintenance, bug fixes, and updates?		
RS3	Is there a governance or hand-over plan?		

D. Impact Assessment[2]

Psychological impact

Unchecked psychological effects can harm both users and contributors, reducing well-being, motivation, and participation. Identifying and addressing these risks helps create healthier, more inclusive, and sustainable software communities.

		Yes	No
PS1	Have possible psychological harms to users (e.g., stress, compulsive usage, social isolation) been identified?		
PS2	Have mitigation strategies been proposed (e.g., contribution limits, recognition systems, usage recommendations)?		

Physical impact

Software use can affect physical health through prolonged screen time, poor posture, or disrupted sleep. Addressing physical risks improves user comfort and reduces health-related barriers to accessibility and use.

		Yes	No
PH1	Could use of this software lead to extended screen time or sedentary behavior?		
PH2	Have potential physical side effects (e.g., eye strain, disrupted sleep, posture issues) been acknowledged?		
PH3	Is there any guidance provided to reduce physical strain (e.g., rest reminders, accessibility settings)?		

Economic impact

Software can introduce hidden costs for users, such as paid dependencies or technical migration. Considering financial impact ensures broader accessibility, sustainability, and fair use across user groups.

		Yes	No
E1	Have you considered the software's impact on users' financial resources (e.g., expensive dependencies, migration costs)?		

[2] de Graaff F, van Amerongen K. Ethics in (Open) Software. Utrecht University. 2025.
Honors Disciplinary Project (July 2025) - This tool has been designed to complement Utrecht University's Ethics and Privacy Scan and support ethical software development in practice.

Environmental impact

Digital technologies contribute to global energy consumption and e-waste. Responsible design reduces environmental harm and aligns the project with sustainability goals and ethical innovation.

		Yes	No
EN1	Have you considered the energy usage or carbon footprint of running the software?		
EN2	Have optimizations been considered to reduce computational resource use?		
EN3	Does the project support reuse, modularity, or longevity to avoid digital waste?		