

Software Management Plan

Quick definitions

- Software Management Plan: refers to activities during the project
- Software Sustainability Plan: refers to activities beyond the end of the project
- Software Stewardship Plan: refers to the entire lifecycle

Software stewardship is the management and oversight of an organization's software assets to help provide business users with high-quality software that is easily accessible in a consistent manner.

The SMP is NOT for editors.

One+ line of code is software.

Survey / List of Questions

Where can the source code be found? Or how can it be obtained?

1. Documentation			
1) What type of documentation is available for the software?		checklist	
	User documentation		
	README file		
	Release notes		
	Docstring/comments		
	CHANGELOG		

	Other		
	None		
Please provide the URL of the documentation website (e.g. read the docs), if available			
2) Is the purpose of the software stated in the documentation?		Yes/No	
3) Does the documentation describe how to	test the software	Yes/No	
	use the software	Yes/No	
	build the software	Yes/No	
	deploy the software	Yes/No	
	install the software	Yes/No	

2. Testing			
Do you test your software?		Yes	No
What type of testing are you using?			
	Unit	Yes	No
	Integration	Yes	No
	Regression	Yes	No
	End-to-end	Yes	No
	Other (e.g. linting)		

Do you use any testing methodology? (e.g. Continuous Integration, Bug-Driver testing etc)	Yes	No
If yes, please name the used methodologies		
Are the tests for the software automated?	Yes	No
Are the tests available with the source code?	Yes	No
Do you provide example parameters and input/output data for testing purposes?	Yes	No

3. Interoperability		
Do you use existing and standard input / output formats?	Yes	No
If yes, please list the standards you are using (e.g. standards defined by GA4GH project)		
Whenever possible please also provide a URL pointing to the description of the standard(s)		

4. Community, contributing & governance		
Does your software have a license?	Yes	No
What is the license of your software?	Name	NA

5. Reproducibility		
Do you use a version control system?	Yes	No
What version control system do you use?	Multiple selection with Other as Text	
	GIT	
	Mercurial	
	Subversions	
	VCS	
	Other	
Do you assign a version to each release of your software?	Yes	No
Do you use semantic versioning?	Yes	No
How do you define dependencies of your software and their version? For instance Maven (for Java), requirements.txt or environment.yml (Python), package.json (JS)	Text	NA
Do you provide input and output examples? Where can they be found?	Yes (+URL)	No

6. Recognition		
Do you include citation information (i.e. how to cite your software in the form of citation.cff, codemeta.json or bibtex)?	Yes	No
Do the releases have a persistent global unique identifier (such as release on Zenodo with DOI, snapshot or release referenced on Software Heritage with SWH-ID)?	Yes	No
Does the citation information contain ORCIDs of the authors	Yes	No

