# STARS Framework

## Checklist

|  |  |
| --- | --- |
| Item | Implementation |
| **Essential components** |  |
| Open license | *Describe your implementation of this component* |
| Dependency management |  |
| FOSS model |  |
| Minimum documentation |  |
| ORCID |  |
| Citation information |  |
| Remote code repository |  |
| Open science archive |  |
| **Optional components** |  |
| Enhanced documentation |  |
| Documentation hosting |  |
| Online coding environment |  |
| Model interface |  |
| Web app hosting |  |

## Description

### Essential components

#### Open license

Free and open-source software (FOSS) license (e.g. MIT, GNU Public License (GPL))

#### Dependency management

Specify software libraries, version numbers and sources (e.g. dependency management tools like virtualenv, conda, poetry)

#### FOSS model

Coded in FOSS language (e.g. R, Julia, Python)

#### Minimum documentation

Minimal instructions (e.g. in README) that overview (a) what model does, (b) how to install and run model to obtain results, and (c) how to vary parameters to run new experiments

#### ORCID

ORCID for each study author

#### Citation information

Instructions on how to cite the research artefact (e.g. CITATION.cff file)

#### Remote code repository

Code available in a remote code repository (e.g. GitHub, GitLab, BitBucket)

#### Open science archive

Code stored in an open science archive with FORCE11 compliant citation and guaranteed persistance of digital artefacts (e.g. Figshare, Zenodo, the Open Science Framework (OSF), and the Computational Modeling in the Social and Ecological Sciences Network (CoMSES Net))

### Optional components

#### Enhanced documentation

Open and high quality documentation on how the model is implemented and works (e.g. via notebooks and markdown files, brought together using software like Quarto and Jupyter Book). Suggested content includes:

* Plain english summary of project and model
* Clarifying license
* Citation instructions
* Contribution instructions
* Model installation instructions
* Structured code walk through of model
* Documentation of modelling cycle using TRACE
* Annotated simulation reporting guidelines
* Clear description of model validation including its intended purpose

#### Documentation hosting

Host documentation (e.g. with GitHub pages, GitLab pages, BitBucket Cloud, Quarto Pub)

#### Online coding environment

Provide an online environment where users can run and change code (e.g. BinderHub, Google Colaboratory, Deepnote)

#### Model interface

Provide web application interface to the model so it is accessible to less technical simulation users

#### Web app hosting

Host web app online (e.g. Streamlit Community Cloud, ShinyApps hosting)