# CHANGELOG description

## CHANGELOG

### PURPOSE

A changelog is a record of what has been done and how that has changed over time. It should provide a list that would quickly inform another user (or remind yourself) what is different in a new version of a project, or software and allow someone to pin an analysis, presentation, manuscript, etc to a particular and traceable form of the project.

See keep a changelog for more information about changelogs.

### FORMAT

The CHANGELOG should be written in a plain, markdown, or restructured text document. These formats are compatible with any software and, importantly, are machine-readable. Markdown is the preferred format as it is capable of being in its read in raw format while also providing symbols that indicate formatting.

For questions about markdown, see the Markdown Guide

### STRUCTURE

Every entry should begin with the new version followed by the date that the changes are being added. Version numbers should roughly follow semantic versioning. In brief there are three parts to a version in the form of `MAJOR.MINOR.PATCH`:

- MAJOR: Breaking changes. Alters results, potentially changing conclusions. Removal or addition of samples.
- MINOR: Bug fix, additional analysis. Change in software, provided it doesn't drastically change results

- PATCH: Small bug fix. For example, a change in capitalization or punctuation. Splitting a source code file into two.

Changes should be added as bullet lists organized by the kind of change. Sections include

- Added
- Changed
- Fixed
- Removed
- Updated

Keep lines limited to 120 characters (including spaces). Save file as `CHANGELOG.md`

**Example**

```
# Changelog guidelines

## [1.0.0] - 2024/07/15

### Added

- Purpose, format, structure, and example sections

## [0.0.1]

### Added

- Created file

[1.0.0]: https://github.com/guthridge-informatics/changelog/releases/tag/0.0.1..1.0.0
[0.0.1]: https://github.com/olivierlacan/keep-a-changelog/releases/tag/v0.0.1
```