Releases

Release

Releases are a way to mark or point to a particular milestone in software development. This is useful for users and collaborators, e.g. I found a bug running version x. For publications that refer to software, refering to a specific release enhances the reproducability.

Apache foundation describes their release policy.

Release cycles will depend on the project specifics, but in general we encourage quick agile development: *release early and often*

Semantic versioning

Releases are identified by a version number. Semantic Versioning (semver) is the most accepted and used way to add numbers to software versions. It is a way of communicating impact of changes in the software on users.

A version number consists of three numbers: major, minor, and patch, separated by a dot: *2.0.0*. After some changes to the code, you would do a new release, and increment the version number. Increment the: * MAJOR version when you make incompatible API changes, * MINOR version when you add functionality in a backwards-compatible manner, and * PATCH version when you make backwards-compatible bug fixes.

Very often package managers depend on `semver` and will not work as expected otherwise.

Releasing code on github

Github makes it easy to do a release straight from your repositories website. See github releases for more information.

CHANGELOG.md

A change log is a way to communicate notable changes in a release to the users and contributors. It is typically a text file at the root of your repository called *CHANGELOG.md*. Every release should have relevant entry in change log.

See Keep a CHANGELOG for some best practices.