

Git Workflow

@dieuph

Contents

- Versioning
- Branching
- Merging
- Pull Request
- Code Review

I. Versioning

X.X.X

Major Minor Patch

I. Versioning

- Comparison of development stage indicators

Stage	Semver	Num. Status	Num 90+
Alpha	1.2.0-a.1	1.2.0.1	1.1.90
Beta	1.2.0-b.2	1.2.1.2	1.1.93
Release candidate	1.2.0-rc.3	1.2.2.3	1.1.97
Release	1.2.0	1.2.3.0	1.2.0
Post-release fixes	1.2.5	1.2.3.5	1.2.5

II. Branching



III. Merging

- Merge
- Rebase
- Cherry-pick

IV. Pull Request

Two types of pull requests:

- Those that receive few comments and get merged quickly.
- Those that get a lot of comments and require several review rounds.

IV. Pull Request

1. Add “Why” Code Comments

2. Make Your PRs [Small](#)

3. Make a Clear Description

- A link to the ticket.
- A summary of what has been done (if it's not obvious from PR's title).
- Links to related pull requests (for example, related changes in another service).

IV. Pull Request

- 4. Comment Your Own Pull Request
- 5. Discuss the Overall Approach Before Implementing the Whole Feature
- 6. Rebase Onto Fresh Master Before Creating a PR
- 7. Respond to Reviews Quickly
- 8. Don't Amend Already Reviewed Commits — Send New Ones
- 9. Thank Reviewers for Their Suggestions

V. Code Review

1. Dependencies

- When was it last updated?
- Who maintains it?
- What are others experiences?

V. Code Review

2. Organization

- What design patterns do they use?
- Is the application design consistent?
- Is the system easily testable? i.e. Controllable & Observable components

V. Code Review

3. Recommendations

- How can they improve the reliability?
- How can they improve the maintainability and efficiency?
- What is my overall assessment of the project?

THANK YOU!