

Reviewing code

Tom Rochette <tom.rochette@coreteks.org>

July 26, 2021 — [f5d1b1b8](#)

- Verify that the build/tests pass
- Read the issue title and description
- New code
 - Understand the feature and associated requirements that are supposed to be implemented
 - Verify code implements the desired feature and that the requirements are completed
- New/Changed code
 - Check code contains tests
 - * Is all the new code covered by those tests?
 - Verify the location of new/moved files
 - * Are the files in the right directory?
 - * Are they appropriately named?
 - Verify classes, methods, functions, parameters naming
 - * Are they significant of their purpose?
 - * Are they clear enough?
 - * Are they respecting the naming convention?
 - Does the code respect **SOLID**?
 - Consider that when functions/methods signature change, code may now be backward incompatible.
 - * Discuss whether this is necessary
 - * Backward incompatible changes should be documented
 - In a weak typed or type hinted language, are parameters and return of functions/methods typed?
 - Are there TODOs that should be completed within this review?
 - Check code for code style issues
- Bug fix
 - Verify that the fix is applied at the right location and will not “fix the symptoms, not the cause”

When reviewing

- Provide specific and actionable feedback
- Clearly mark nitpicks and optional comments
 - Alternatively, use an approach such as [RFC2219](#) where you indicate whether a change is a **MUST**, **SHOULD**, or **MAY**
- Assume competence
- Provide rationale or context
- Consider how comments may be interpreted
- Don't criticize the person, criticize the code
- Don't use harsh language

1 Reference

- <https://en.wikipedia.org/wiki/SOLID>
- <https://medium.com/palantir/code-review-best-practices-19e02780015f>
- <https://phauer.com/2018/code-review-guidelines/>
- <https://smartbear.com/learn/code-review/guide-to-code-review-process/>

- <https://smartbear.com/learn/code-review/best-practices-for-peer-code-review/>
- <https://nyu-cds.github.io/effective-code-reviews/03-checklist/>
- <https://google.github.io/eng-practices/>
- <https://testing.googleblog.com/2019/11/code-health-respectful-reviews-useful.html>
- <https://engineering.18f.gov/code-review/>
- <https://conventionalcomments.org/>
- https://docs.gitlab.com/ee/development/code_review.html