



INNOVATIVE SOLUTIONS
BY OPEN SOURCE EXPERTS

OCA DAYS 2022

**How to contribute to OCA doing reviews
and submitting your first module**

Akim Juillerat

Introduction

- OCA Guidelines
- Reviews
 - Why do we review code?
 - Mindset
 - Review process
- Submitting your first module
 - Getting started
 - Be OCA Compliant

OCA Guidelines

→ Contributing.rst

- <https://github.com/OCA/odoo-community.org/blob/master/website/Contribution/CONTRIBUTING.rst>

→ Useful resources

- <https://github.com/OCA/maintainer-tools/>
 - Module template
 - Wiki
 - Migration guides
 - FAQ

Code reviews

Code reviews

- Why do we review code?
 - Save time
 - Find bugs
 - Improve maintainability

→ Mindset

- Think positive, be nice and patient
- Learn from the feedbacks, don't get offended at them
- You are not your code
- Empathize with the author's situation, understand the context
- Everyone can give feedbacks, everyone need reviews
- Be constructive, better to ask rather than to assert

→ Egoless programming

<https://blog.codinghorror.com/the-ten-commandments-of-egoless-programming/>

→ Review process

- Learn about the scope and the context
- Check that the code does what is stated in the module description
- Check that the changes matches the commit messages
- Check for unwanted side effects
- Check the overall design

- Approving, commenting, requesting changes
 - If the code does what's stated, and previous checks didn't raise any issue, you **MUST** approve the PR.
 - If the code does what's stated, but the code is not efficient, you can comment and suggest how to improve it (or open another PR targeting the original PR branch)
 - If the code does not do what's stated, or there are dangerous side effects, then you can request changes, which must be the exception rather than the rule.
 - By requesting change, you commit yourself for another review

→ Finally

- A PR does not need to be perfect to be merged.
- Consider maturity levels
- Consider adding entries in the module roadmap.rst
- Let's try to merge as many PRs as possible instead of leaving them open and being closed by the OCA bot after a while

Submitting your first module

Submitting your first module

→ Getting started

- Fork the repository, and clone it on your machine
- Install pre-commit
- Start coding

Submitting your first module

- Be OCA Compliant
 - Follow the guidelines
 - Take care of the readme generation
 - Be kind to reviewers
 - Write and run the tests
 - Check the diff
 - Write proper commit messages
 - Review yourself
 - Open the Pull Request
 - Make sure travis is happy

camptocamp[^]

INNOVATIVE SOLUTIONS
BY OPEN SOURCE EXPERTS

Questions?