


# Pull Requests Guide

- [Do's and Don'ts when submitting a PR](#)
- [Pull Request Naming](#)
  - [Examples](#)
  - [Specifics](#)
- [Don't](#)
- [Pull Request Review edict](#)
- [Pull Request Description](#)
- [Links](#)

 This is a best practice guide, not a policy. If you think there is a better way, try it out and bring your lessons back to this guide.

Pull Requests are the primary mechanism by which we collaborate as a team, ensure code quality and ultimately underpin our whole development process.

## Do's and Don'ts when submitting a PR

- **Do** submit PRs from your own fork of the repo (this one is a rule)
- **Do** submit PRs using a branch, ideally name your branch after a Jira story or a descriptive name using [kebab-case](#)
- **Do** fill in the PR description using the predefined headings
- **Do** utilise the “draft” feature on GitHub when submitting PRs that are not yet ready for review
- **Do** keep PRs small
- **Do** squash commits when merging
- **Do** tag PRs with the relevant Slack channel
- **Don't** submit single PRs that e.g. both add new functionality *and* refactor — [make it work, make it right](#)

## Pull Request Naming

Use this convention: `ABC-123 [Tag] Title`

where

- **ABC-123** - Card reference
- **[Tag]** - High level context
- **Title** - Change description
- **GitHub labels** - Common and long-lived tags

## Examples

 Remove GraphQL schema check from CI



ORI-222 Remove GraphQL schema check from CI

RKT-333 RKT-334 Remove GraphQL schema check from CI


[Deposits] Remove GraphQL schema check from CI


## Specifics

- **ABC-123**
  - uppercase, with a dash, no `[]` around
  - multiple references are separated by space
- **[Tag]**
  - optional
  - separated by space
  - can have multiple tags, each wrapped in `[]`, tags are separated by space
  - Use your own tags
- **Title**
  - required, sentence casing, no period at the end
- **GitHub labels**
  - optional, lowercase, with a dash

## Don't

- **Don't** use common title tags
  - Instead use PR labels
    - `[Ops]` -> `dev-ops` label
    - `[Backend]` -> `dev-backend` label
    - `[Hotfix]` -> `hotfix` label
    - etc
  - **Why?** We already have standardised PR labels, no need for duplication. Certain labels are linked to Slack channels.

 `[Ops]` Remove GraphQL schema check from CI

 `ENG-1337` Remove GraphQL schema check from CI

labels: `dev-ops`

- **Don't** use or create labels which are short lived or not common
  - Instead use title tags
    - `auth-api` label -> `[AuthApi]` title tag
    - `proj-new-proposal-editor` label -> `[Project New Proposal Editor]` title tag
  - **Why?**
    - These descriptors are short lived, while github labels have to exist forever.

- These descriptors bring additional context which will be included in merge commit messages, unlike github labels.

⚠ TRI-148 Remove GraphQL schema check from CI

labels: proj-auth-api

✓ TRI-148 [Auth Api] Remove GraphQL schema check from CI

- **Don't** use [Chore] or [NoTicket] title tag
  - Instead use make a Jira ticket for your team and include it, if it really isn't team oriented (or a quick PR) just leave it out.
  - **Why?** We try our best to associate work with teams rather than individual people. Jira is used to oversee what everyone spends their time on, if there's no related Jira ticket your team mates may not be across the change. If you really aren't up for making a Jira ticket just leave it out, just please don't use [NoTicket] .

⚠ [Chore] Add triton team to codeowners

✓ ENG-231 Add triton team to codeowners

- **Don't** use title tag if that can be part of the title
  - **Why?** Tags are not a replacement for a self-contained and readable title.

⚠ [Auth Api] TRI-148 userConfrim request spec

✓ TRI-148 Add Auth API userConfrim mutation request spec

## Pull Request Review edict

- **Do** come in curious, try to make it an open conversation without judgement.
- Comments vs. Request Changes on Pull Requests:
  - Request Changes means:
    - I'm actively taking part in the PR, I understand I will be needed to review future changes on it.
    - This contains "show stopping" changes, please don't send this live until it's been worked through a little more.
    - This is a blocking comment, only the reviewer can unblock this PR
  - Comments means:
    - I'm taking a little less responsibility with a drive by review, less responsibility and just throwing in my 2c.
    - A reviewer or author can unblock this PR by resolving the comment

## Pull Request Description

- **Do** fill in the PR description using the predefined headings so that others have enough context to review your changes
  - **What.** (required) What's the high level description of the change?
  - **Why.** (required) Why do we need this change? What's the context?  
Adding link to Jira ticket is not enough. Provide enough context that PR could be understood without Jira reference. Keep it short and informative.
  - **How.** (optional) Implementation details to highlight
  - **Before/After.** (optional) Include screenshots/results before and after
- **Do** include Jira ticket URL

### Why Jira link is not enough on its own?

- Better experience for devs, as all info is on a single page.
- Github is a permanent storage for code changes and we are more likely to stop using Jira and loose all the context.
- Jira ticket can have multiple PRs with different reason in the Why section.

## Links

- [📄 2022-04-04 Proposal: PR naming convention](#)