

Pull Request Guidelines for TBMQ

This document describes procedures for creating issues and the contribution of code changes via pull requests (PR). Following the guidelines is mandatory for ThingsBoard employees/contractors and highly recommended for outside contributors. Please read the doc before submitting your contribution. The PRs that do not follow the guidelines will be closed without merging.

These guidelines are not final and may change over time. We welcome you to submit your feedback as a comment to this document or via email to info@thingsboard.io.

Issues

TBMQ GitHub repository supports 3 templates for [issues](#): Bug report, Feature request, and Question. Each PR must reference either a Bug report or a Feature request. The issue must be created before submitting the PR.

Motivation

TBMQ is an open-source project. Publicity is good for the community and for project adoption. GitHub Issues is the perfect place to discuss new feature requests and receive feedback/improvement suggestions.

Valid Issue/PR description allows users to use a pre-release version of the product built from sources and optimize work for QA, Technical Writers, and Marketing team.

Version policy

ThingsBoard attempts to follow [Semantic Versioning 2.0](#) without sacrificing the speed of development. As a short summary:

Given a version number MAJOR.MINOR.PATCH, increment the:

1. MAJOR version when you make incompatible API changes,
2. MINOR version when you add functionality in a backward-compatible manner, and
3. PATCH version when you make backward-compatible bug fixes.

Additional labels for pre-release and build metadata are available as extensions to the MAJOR.MINOR.PATCH format.

Branch policy

The “**main**” branch is used to prepare the latest critical bug-fix release. The “main” branch should always be ready for immediate release. We can’t merge new features or low-priority bug fixes directly to main since they require additional QA efforts and may block the delivery of critical bugfix releases.

The “**develop/MAJOR.MINOR**” branch (e.g. develop/1.1) is used to prepare the next minor release. This branch should accumulate all non-critical bug fixes and new features with the corresponding fix version.

The “**feature/ISSUE_NUMBER**” branch (e.g. feature/5976) is created by [Dima Landiak](#) , [Andrew Shvayka](#) , or [Volodymyr Babak](#) upon request. The branch must be created if more than 1 person contributes to the PR.

Pull requests

PR Submission

Each PR must reference an [issue](#). Use [Branch policy](#) to determine the target branch. PR name may contain **[WIP]** prefix (Work In Progress) but must contain the fix version as well. For example: “[WIP][3.4] Draft of my super feature for internal review by other team members”.

For Bug Fix:

The Issue must contain valid steps to reproduce, while the PR must contain a description of what and how the bug was fixed.

UI: Add a screen of the bug fix result (e.g. screenshot of the component that was fixed. Add “before” and “after” for best user experience.) whenever it is reasonable.

Back-end: As a first commit, please add the corresponding unit and/or integration test(s) that fails on the bug. After the bugfix, the test must turn from failed to successful. Provide a written explanation if you have failed to add tests.

For Feature Request:

The PR must include a valid description of the scope of changes. See PR checklist. Feel free to contribute to the “pull_request_template.md” file that is located at the root of the project.