

ODIM Review Process

The following is a proposed review process that will be discussed in upcoming TSC meetings and should not be considered final until ratified by TSC

Larger Features and/or New Projects

There are times when new features and ideas will need a TSC review. Examples are

- A completely new Redfish service for the Aggregation Function
- A new feature that changes existing interfaces (e.g upgrade of northbound Redfish interfaces, a change to interfaces between plugins and the Aggregator)
- A new project that will be under ODIM
- Potential other larger changes to the existing architecture or behavior

In all these cases it is advised that a proposal is created on the wiki and that the TSC is asked to review it. The following should be considered for such a proposal:

- A thorough description of the feature/project/change and why we think it is a good idea
- A thorough description of any changes to the architecture or behavior of the stack
- A thorough description of any changes to existing interfaces
- Indicate if this change would go into an existing repository or if it would require a new repository
- Any other information that could be helpful to TSC when reviewing the proposal

After the proposal has been approved by TSC GIT issues can be created and work can start

Feature and bugs Management

For smaller features and bug fixes it is advised to instead create a GIT issue directly

- Features and bugs to be filed as Github Issues
- Issues are then appropriately labeled using the standard github labels.

ODIMRA Code Review Process

1. Developer to pick up an open issue
2. Developer has 2 options here to clone a repository

Option	Pros	Cons
Developer creating feature branches on ODIMRA repository directly	Better collaboration on the features or bugs that are getting implemented	<ol style="list-style-type: none">a. In order to allow user to create feature branches, we need to add the developer as collaborator to this repository. Hence we may end up adding all the developers who wish to contribute here as collaborators.b. Over the period of time, there would be lot of features branches in the repository. With strict guidelines we can discipline developers to delete the feature branches once the PR is merged.
Developer can fork the repository and implement the feature or fix the bug in the forked repository and raise a PR against the appropriate branch when ready.	<ol style="list-style-type: none">a. No need to add the developer as collaborator to the repositoryb. there will be lesser branches in the repository	Lesser collaboration on the features or bugs that are getting implemented

3. Implement the PATCH and raises a PR against the ODIMRA repository
4. When developer raises the PR, the GitHub actions checks are triggered automatically.
5. If the checks passes, then PR will go under review process and at least 2 maintainers needed to approve it before the PR gets merged.
6. If the checks fail, the PR will be rejected.

