# Contribution Guidelies

This document serves as a standard git workflow to utilize

across the dataunit project in order to facilitate

consistency and avoid mess and collision within the

repository's history. This document is subject to change

based on team conversation and is simply an effort to

keep everyone on the same page.

## Branches

- master - The master branch is intended to be functional (i.e. No broken or in-progress features) at all times. This means passing tests at all times. Releases should be tagged on this branch.

- feature/<issue\_number> - Feature branches should be named in this format, where issue\_number corresponds to the issue which is being worked on. This is for standard feature development. Bugs/defects have their own branch type.

- bugfix/<issue\_number> - Bugfix branches are identical to feature branches. The only difference is the type of work being done (new development or bugfix).

## Workflow

- Choose an issue to work on and assign it to yourself.

- Create a feature or bugfix branch for the issue. First, update your local copy of the master branch using `git checkout master && git pull`. Then, create and checkout your new branch using `git checkout -b feature/123`.

- Perform all your development work on this new branch. This keeps in-progress work out of the master branch so it is always in a passing state. This also allows you to commit/push often to avoid losing work.

- As often as makes sense, commit your changes to your feature branch. Many small commits are good, as they allow reverting back changes easily. `git add . && git commit -m "Short description of what you did" && git push`. Keep in mind that once this becomes public open source, all commit messages will be publicly visible.

- Once the feature is ready to merge to the master branch (i.e. all code passes all tests), create a merge request. This will allow the team to code review everything going into the master branch in order to raise collective awareness and increase code quality.

- Once at least one team member has approved the changes, the merge request should be approved and the changes merged to the master branch.