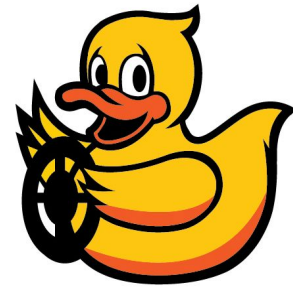


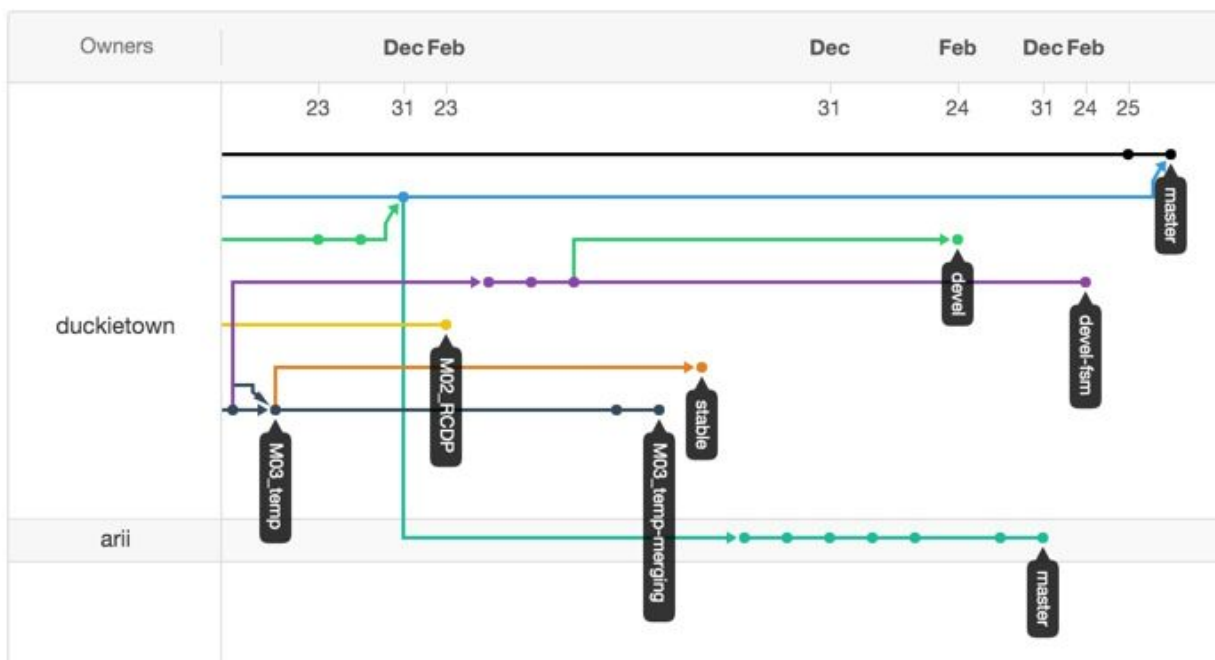
Duckietown Engineering

Source Code Management

Rules and Conventions



This is the set of conventions we use to work as a team on our codebase. The Git policy will evolve in the future, as we learn more and more about Git.



Related documentation

Before you can understand this, you have to know about Git. You can find pointers to that documentation in [Essential References](#).

Simple Rules

The module branch

- Each module has a `<module-name>` with the pattern `MXX_YYYY`. For example the name for the second module as `M02_RCDP`.
- Each module has a branch named `<module-name>`.

The personal branch off the module's branch

- When you start working on the module, you **MUST** check out the module branch, and create a personal branch named `<module-name>-<handle>`.
- You **MAY** commit and you **MAY** push to that branch.
- When you push to that branch, your work becomes public.

The master branch

- You **MAY NOT** push to master, except when explicitly indicated by the instructors.

Note: if you pushed to master by mistake: We are going to fix it in a way that is transparent to the others, but it will involve you having to remove `~/duckietown` and recreate it.

Advanced rules

Forks

- If you don't know about "forks": you cannot fork.



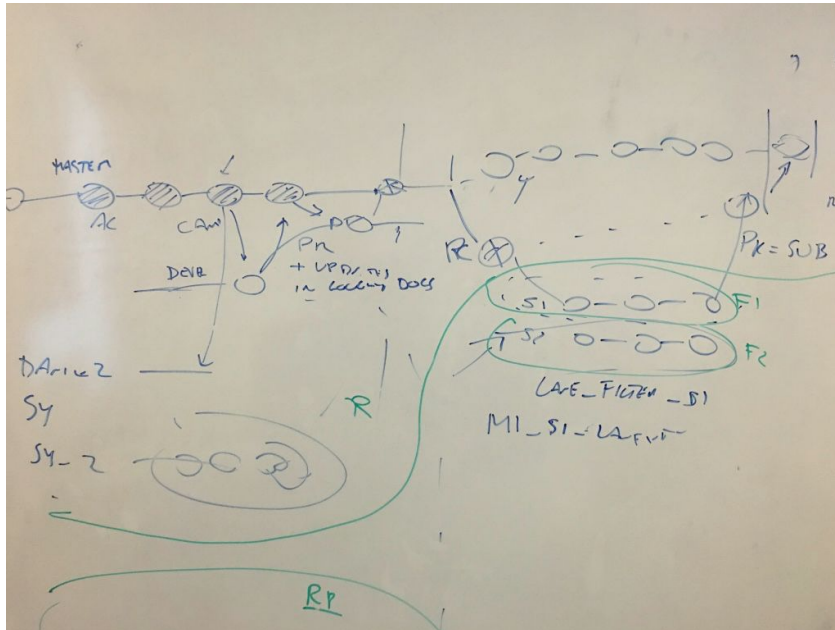
- Ceci n'est pas une fork:
- If you know what is a fork: You MAY fork our repository to your account. Then issue a pull request.

Pull requests

It's still too early, but prepare for something wonderful.

Rules for staff

This is the basic idea:



Starting from that, we decided the following refinements:

- master is a sandbox
- “stable is the new master”
- devel is development
- stable is owned by Andrea
- devel is owned by Liam
- devel -> stable is by pull requests approved by Andrea
- to develop: create devel-<feature> off devel
- the documentation is in sync with stable at all times
- the documentation is in sync with master, unless master is temporarily corrupted.
- student module branches start from stable, not from devel