**Git normalization**

Git branches:

* main (production line)
* dev (develop line)
* <features> (temporary lines)

We have a main branch which is the production line, and a dev branch that is the pre-lunch branch of main.

Some temporary branches should be created for each feature development inside the project.

For each feature, we:

* create a branch locally
* develop & tests locally
* create a Pull Request for the dev branch
* validate the feature in a String review
* merge the feature and the dev branches
* delete the feature branch

Every month, we review the dev branch and merge it into the main branch.

Commit messages rules:

* <name>: <feature\_name>: <feature\_description>

Link Github:

<https://github.com/EpitechMscProPromo2023/T-ESP-800-18318-Yaboi-1>

We’d rather have little frequent updates than heavy sporadic ones.

**Code normalization**

Technologies:

* Unity C#
* React / Angular / VueJS
* ?

We can use code conventions in order to have organized and beautiful codes, with commentaries and documentations.

Ex: <https://docs.microsoft.com/fr-fr/dotnet/csharp/fundamentals/coding-style/coding-conventions>

**Project organization**

The project works on an agile strategy with meetings every other week to set goals regularly. Work is usually spread in groups of 2 or 3.

Tasks to do are visible on the Jira accessible through Teams and are updated in the meetings.