Procédures Gitlab

# I - Découpage des branches et explication de leur statut

## A – Dev

Branche par défaut du projet, elle va servir de base pour toutes les autres branches lors de la création d’une issue. Elle n’a pas pour vocation d’avoir un pipeline de CI/CD branché sur elle. Il est possible de push directement dessus même si c’est à déconseiller pour éviter de provoquer des merges plus compliqués aux personnes qui travaillent sur des branches d’issues.

## B – Hml

Branche d’homologation (=recette). Cette branche a le statut PROTECTED avec l’interdiction de push dedans. Seuls les merges sont autorisés et ces merges doivent se faire depuis la branche Dev uniquement. Il faut mettre un pipeline CI/CD dessus pour pouvoir effectuer la recette du code qui arrive sur cette branche.

## C – Prod

Branche de production. Cette branche a le statut PROTECTED avec l’interdiction de push dedans. Seuls les merges sont autorisés et ces merges doivent se faire depuis la branche Hml uniquement. Il faut mettre un pipeline CI/CD dessus pour pouvoir livrer le code qui arrive sur cette branche en production. Avant de merge ici, tout doit avoir été testé de manière intensive pour ne rien casser.

# II – Ajouter une nouvelle feature/Corriger un bug

1/ Créer une issue sur le projet

2/ Créer la merge request associée à l'issue

3/ Coder en local et push sur la branche de l’issue pour éviter des pertes de données

4/ Tester en local

5/ Une fois le test en local réussi, merge dans dev

6/Le rôle du Tech Lead est de valider ce qui a été poussé dans dev et qui répond à la correction du bug ou à la nouvelle feature

7/ Merge dev dans hml une fois la validation effective

8/ Faire la recette sur hml et valider

9/ Merge en prod

10/ Valider la prod

11/ Recommencer avec les next features