

# Workflow de développement DSI Banque

## A – Développement d'une nouvelle fonctionnalité

**Fréquence** : Dès que besoin

**Responsable** : Développeur

**Actions** :

Action	Commande	Note
Se placer sur la branche commune	git co dev-commun	/
Mettre à jour les branches en local	git fetch origin	/
Mettre la branche commune à jour en rebasant	git rebase origin/dev-commun	/
Créer une branche temporaire et s'y placer	git co -b {nom-branche}	<i>Pour le nom de la branche, voir annexe</i>
Mettre à jour les fichiers dans son IDE	/	<i>Download sur NetBeans ou PHPStorm</i>
Développement	/	/
Ajouter / Supprimer les fichiers au commit	git add . git rm {nom-fichier}	/
Commiter chaque micro-fonctionnalité	git ci '{message}'	<i>Plus il y a de commits mieux s'est. Ne pas en abuser non plus.</i>
Envoyer les commits sur le serveur Gitlab	git push origin {nom-branche}	<i>Pusher au moins tous les soirs</i>

## B – Fin d'une fonctionnalité

**Fréquence** : Dès qu'une fonctionnalité est terminée

**Responsable** : Développeur

**Actions** :

Action	Commande	Note
Effectuer les vérifications qualitatives	/	<i>Code Sniffer (Copy Paste Detector) (Mess Detector)</i>
Envoyer un mail à David	/	<i>Pour demande le merge sur commun. Sujet : « MR.BFI – {nom-branche} »</i>

## C – Merge

**Fréquence** : A réception d'un mail

**Responsable** : Lead Développeur

**Actions** :

Action	Commande	Note
Se placer dans le répertoire commun	cd /app/webphp/bfi/	/
Mettre à jour les branches en local	git fetch origin	/
Effectuer le code review	/	/
Se placer sur la branche commune	git co dev-commun	/
Merger la branche temporaire	git merge origin/{nom-branche}	/
Envoyer les commits sur le serveur Gitlab	git push origin dev-commun	/
Mettre à jour le cache sur l'environnement	php app/console ca:cl php app/console ca:cl --env=prod	/
Mettre à jour les ressources publiques	php app/console as:du php app/console as:in	/
Supprimer la branche temporaire à distance	git push origin --delete {nom-bran}	<i>Sauf si stipulé de la conserver</i>
Envoyer un mail au développeur	/	<i>Sujet : « MR.BFI – Fin processus »</i>

## D – Test sur l'environnement commun

**Fréquence** : Dès que le merge est terminé

**Responsable** : Développeur

**Actions** :

Action	Commande	Note
Se connecter à l'environnement commun	/	<i>commun.dev.bfiweb.fiducial.dom</i>
Tester que ses développements fonctionnent	/	/
Supprimer la branche en local	git branch -d {nom-branch}	<i>Seulement si développement validé</i>

## E – Préparer la mise en production (MEP)

**Fréquence** : Le mercredi soir, au besoin

**Responsable** : Lead Développeur

**Actions** :

Action	Commande	Note
Se placer dans le répertoire commun	cd /app/webphp/bfi/	/
Mettre à jour les branches en local	git fetch origin	/
Créer un tag de version	git tag -a {nom-tag} -m '{message}'	<i>Pour le nom du tag, voir annexe</i>
Envoyer le tag sur le serveur Gitlab	git push origin {nom-tag}	/
Créer une nouvelle évolution	/	<i><a href="http://evol.azur.fiducial.dom">http://evol.azur.fiducial.dom</a></i>
Créer le plan de déploiement	/	/

## Annexe – Nommage des branches

Les branches temporaires (une par fonctionnalité) doivent respecter le format suivant :

**{type}/{nom}**

<b>Type</b>	Selon type de la fonctionnalité : « fix » / « hotfix » / « fonc » / « test » / « div »
<b>Nom</b>	En français. Représentatif et concis. En camelCase.

Exemple :

- Correctif sur les codes événements du module OD : **fix/codeEvenementOD**
- Correctif urgent sur le relevés de comptes : **hotfix/releveCompte**
- Ajout d'un éditique : **fonc/lettreCaution**
- Création d'une branche de test : **test/mergeBranche**
- Rattrapage d'une erreur : **div/rattrapageErreur**

## Annexe – Nommage des tags de version

Les tags de version (un par MEP) doivent respecter le format suivant :

- Le tag doit toujours commencer par « v ».
- Chaque tag est ensuite composé de 1 à 2 digits indique la version (exemple v1.2 ou v1.2.5)
  - Le premier digit étant la version majeure actuelle (changement en cas de refonte complète) ;
  - Le deuxième digit étant la version mineure (changement en cas de nouvelle fonctionnalité) ;
  - Le troisième digit (optionnel) étant la version corrective (ajout et changement en cas de correctif de la version mineure).