

Initiation GIT

moshi@moshi.fr

6 février 2020

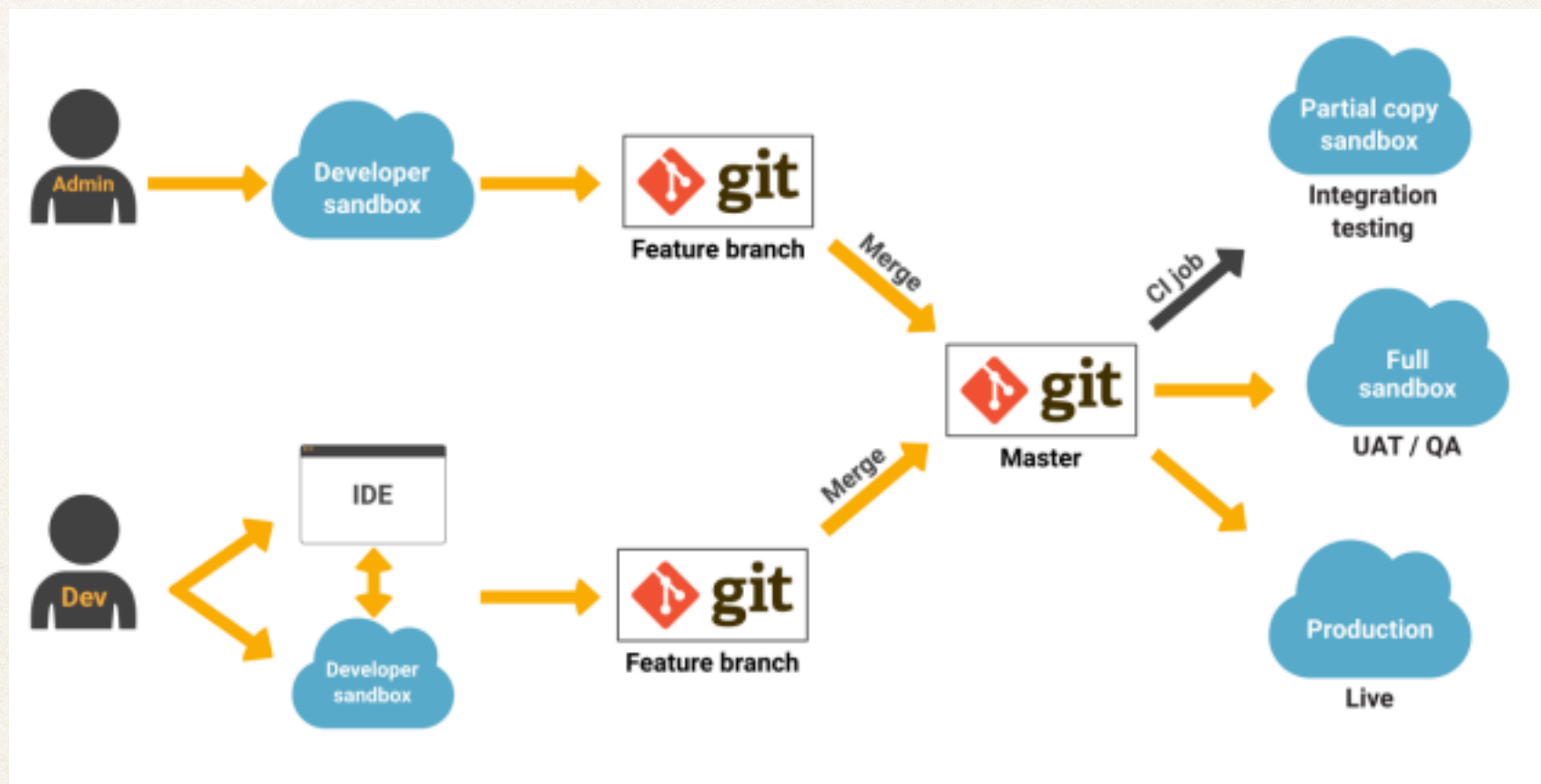
Présentation

- ❖ Créateur : Linus Torvalds
- ❖ Logiciel de versioning reconnu et le plus utilisé professionnellement
- ❖ Décentralisé
- ❖ Développements parallèles possible

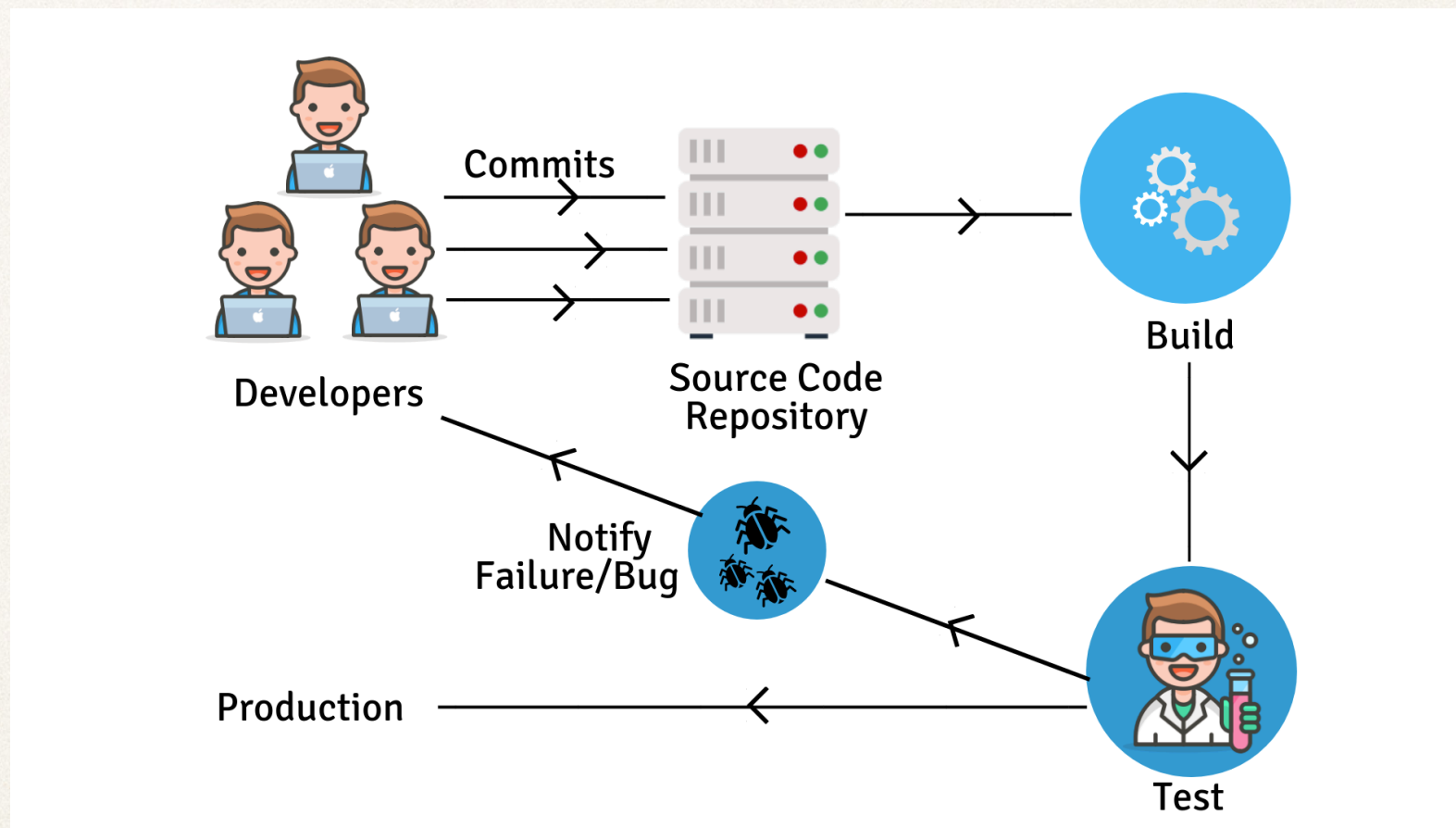
Versionning

- ❖ Historique (différentes versions)
- ❖ Travail d'équipe
- ❖ Blame
- ❖ Déploiement

Présentation



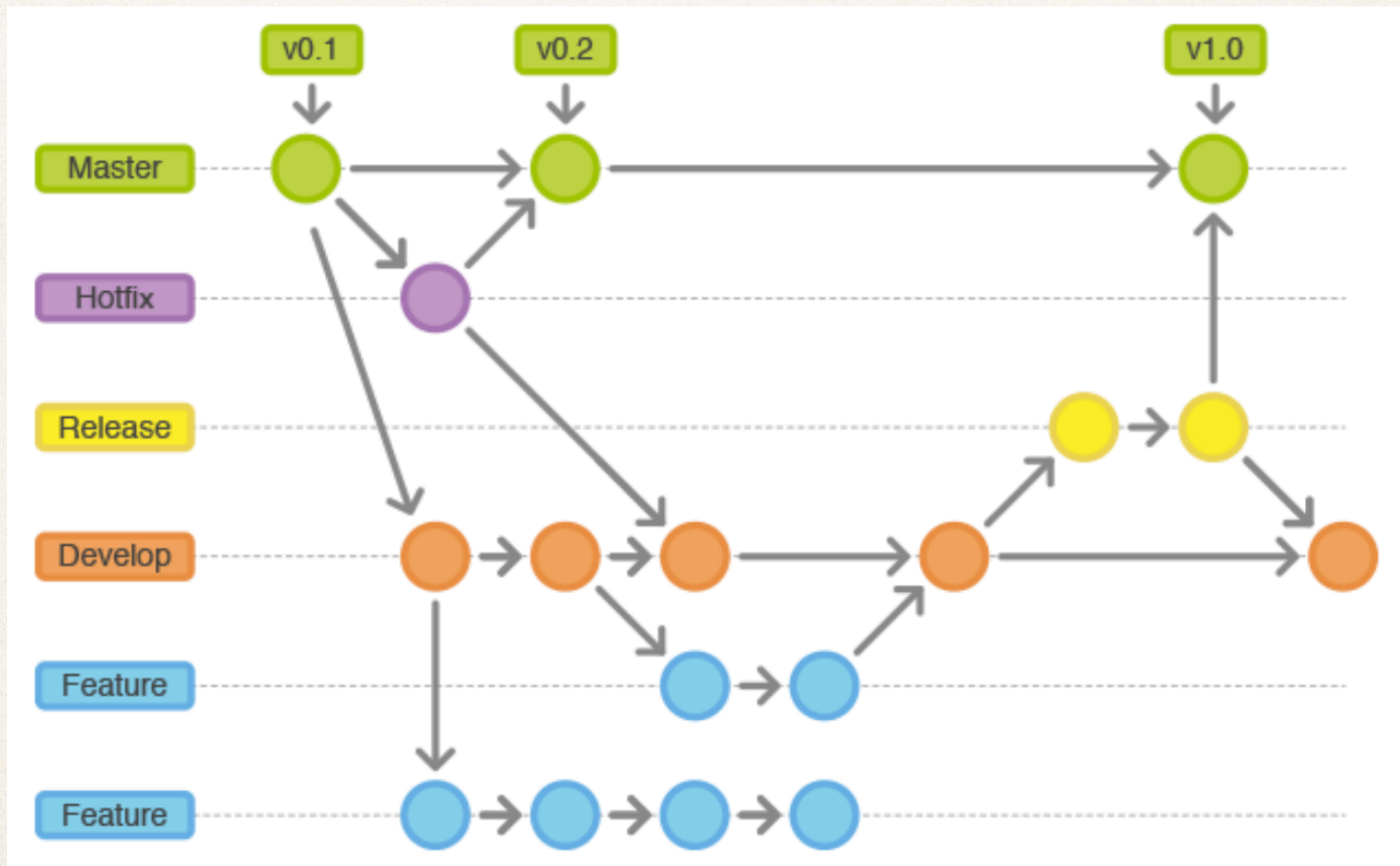
Présentation



Comment ça marche ?

- ❖ Repository (dépôts)
- ❖ Branches
- ❖ Commit
- ❖ Push
- ❖ Merge

Branches



Github / GitLab / Bitbucket

- ❖ Github, GitLab et BitBucket remotes qui utilisent git
- ❖ Repositories accessible en ligne
- ❖ Visuel
- ❖ Services tiers et connectivité

Commandes utiles / Configuration

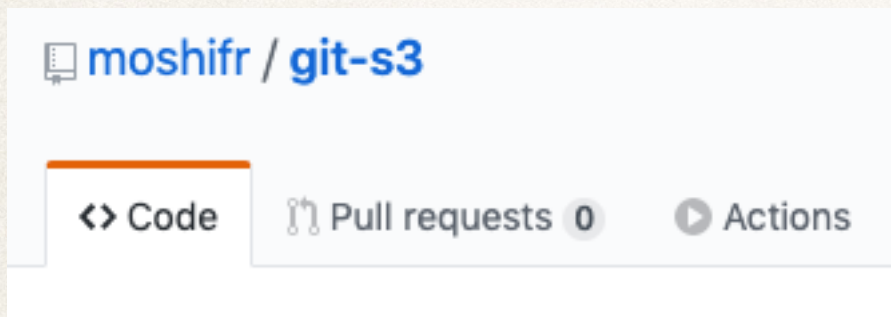
- ❖ `git config --global user.name « votre_pseudo »`
- ❖ `git config --global user.email "moi@email.com"`

Commandes utiles / Utilisation

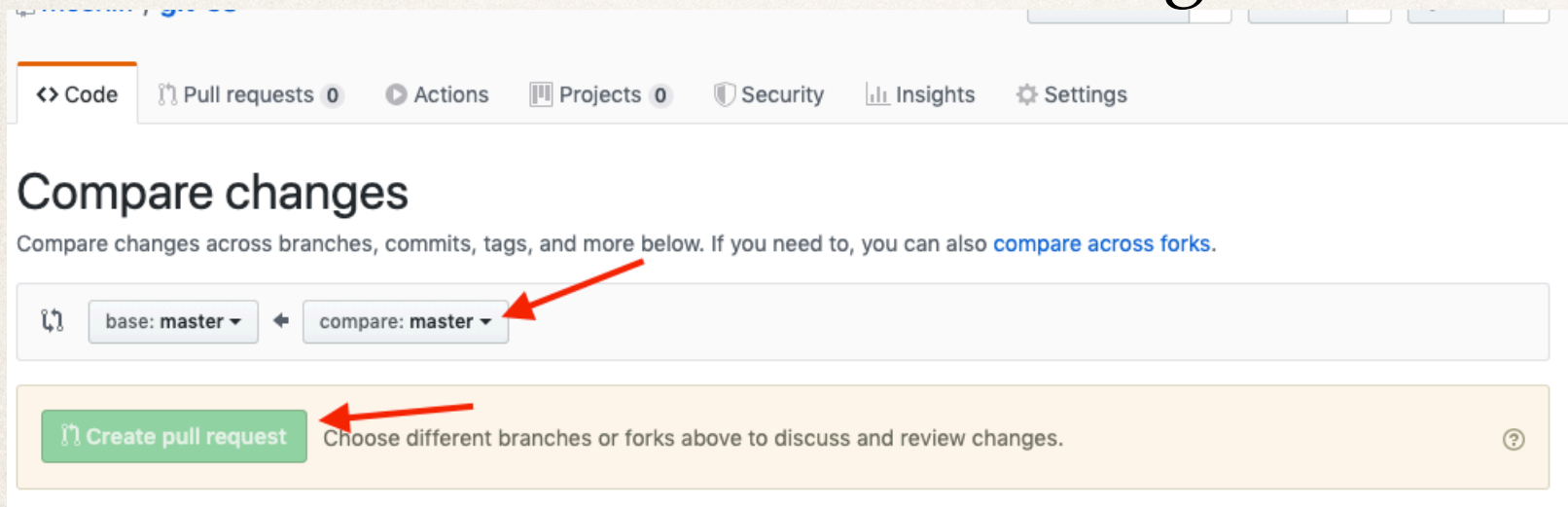
- ❖ git init
- ❖ git clone
- ❖ git add
- ❖ git commit
- ❖ git push
- ❖ git pull
- ❖ git checkout
- ❖ git branch
- ❖ Création d'un dépôt vide
- ❖ Cloner un dépôt existant
- ❖ Ajouter un fichier dans l'index
- ❖ Enregistrer des modifications dans le dépôt local
- ❖ Envoyer les modifications locales vers une référence
- ❖ Récupérer les dernières modifications de la branche
- ❖ Bascule sur une autre branche en restaurant les modifications si branche locale
- ❖ Permet de lister / créer modifier ou supprimer des branches

Pull Request

- ❖ Sert à fusionner des branches une fois le développement finalisé



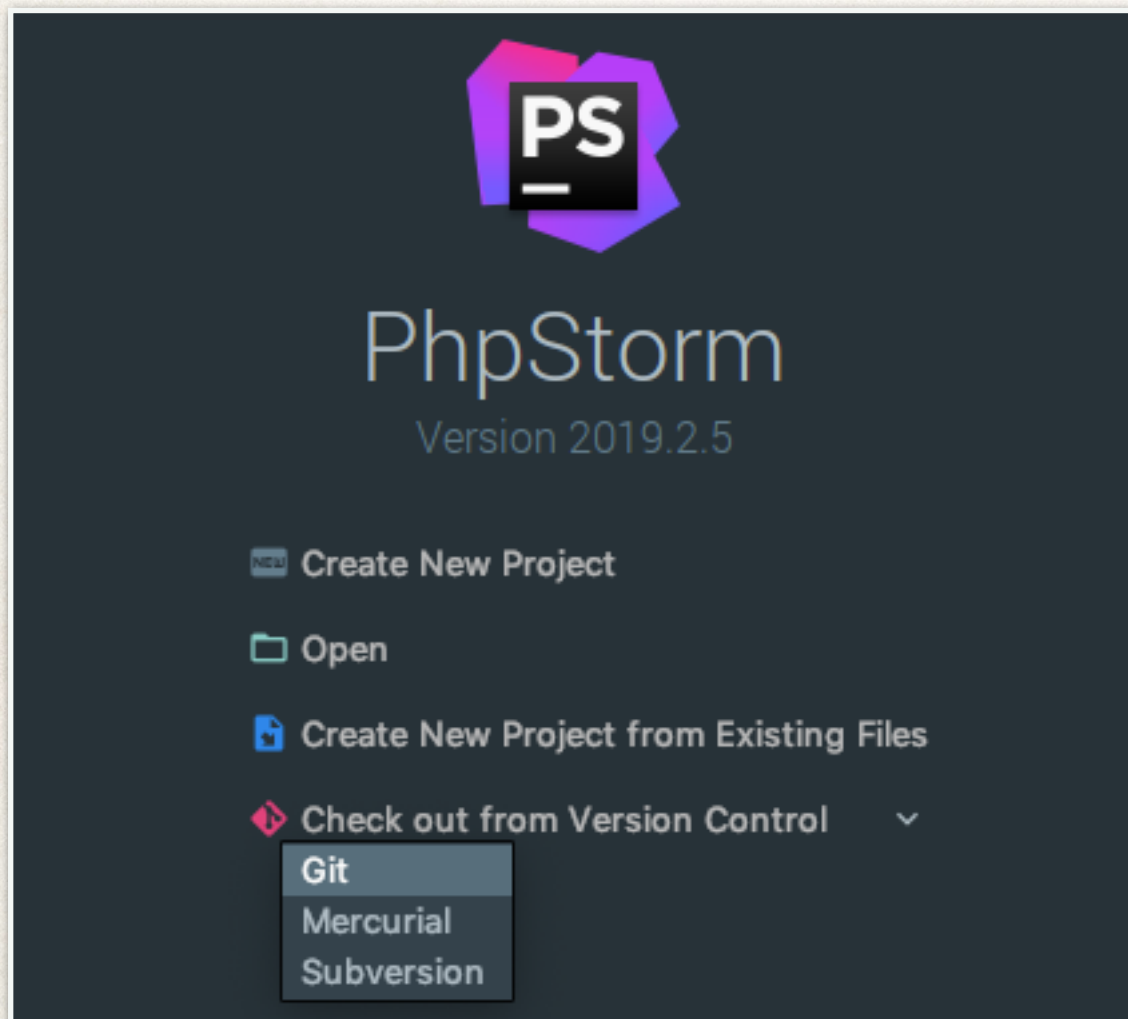
- ❖ Choisir la branche destination à gauche et source à droite



TD - Initialisation PhpStorm

PhpStorm

- ❖ Pour notre TD / TP nous utiliserons PhpStorm pour gérer les fonctionnalités git
- ❖ Cloner une dépôt existant :



PhpStorm

- ❖ Récupérer l'url du dépôt sur Github : <https://github.com/moshifr/git-s3>

The screenshot shows the GitHub repository page for 'moshifr/git-s3'. At the top, there are buttons for 'Unwatch' (1), 'Star' (0), and 'Fork' (0). Below these are tabs for 'Code', 'Issues' (0), 'Pull requests' (0), 'Actions', 'Projects' (0), 'Wiki', 'Security', 'Insights', and 'Settings'. The main content area states 'No description, website, or topics provided.' and includes a 'Manage topics' link. Below this, statistics show '1 commit', '1 branch', '0 packages', '0 releases', and '1 contributor'. A red arrow points from the '1 contributor' text to the 'Clone or download' button. The 'Clone or download' button is green and has a dropdown arrow. Below the button, there are links for 'Branch: master', 'New pull request', 'Create new file', 'Upload files', and 'Find file'. The repository content shows a commit 'Dnd-Moshi first commit' with a file 'README.md' and a 'first commit' label. A modal window titled 'Clone with HTTPS' is open, showing the URL 'https://github.com/moshifr/git-s3.git' and a 'Use SSH' link. A red arrow points from the 'Use SSH' link to the 'Clone with HTTPS' title. At the bottom of the modal are buttons for 'Open in Desktop' and 'Download ZIP'.

moshifr / git-s3

Unwatch 1 Star 0 Fork 0

Code Issues 0 Pull requests 0 Actions Projects 0 Wiki Security Insights Settings

No description, website, or topics provided. Edit

Manage topics

1 commit 1 branch 0 packages 0 releases 1 contributor

Branch: master New pull request Create new file Upload files Find file Clone or download

Dnd-Moshi first commit

README.md first commit

README.md

Clone with HTTPS Use SSH

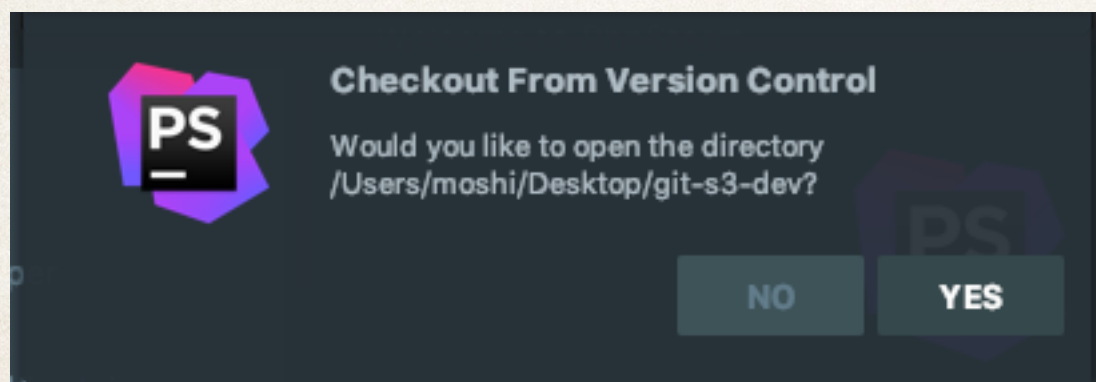
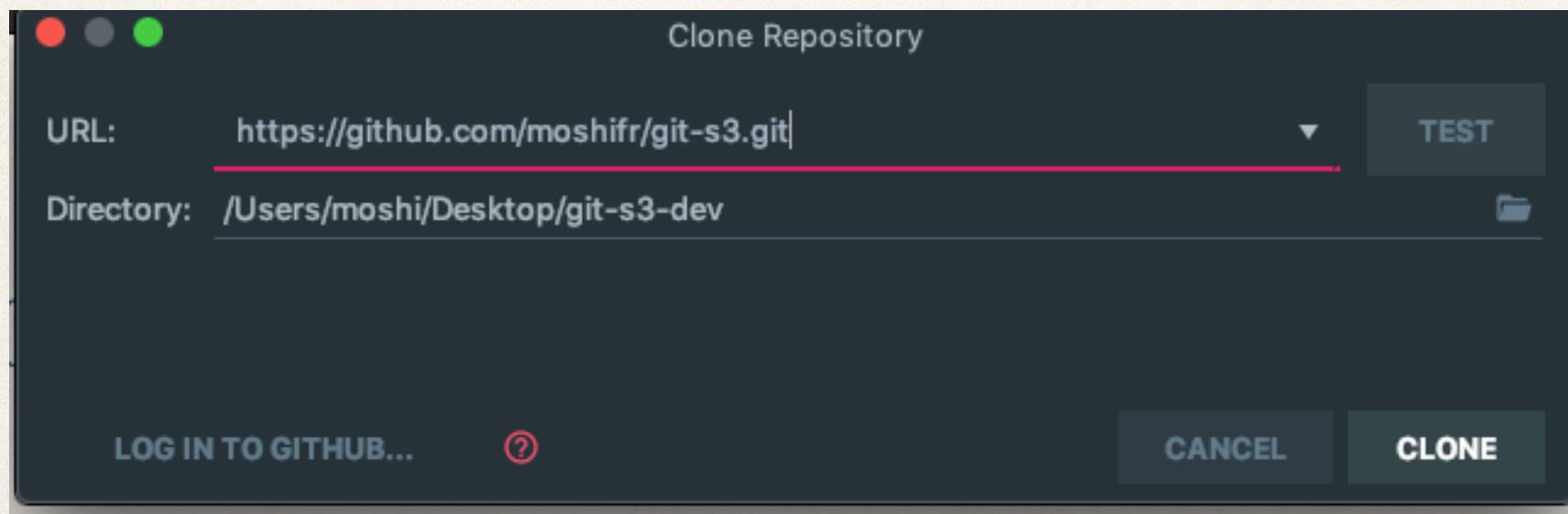
Use Git or checkout with SVN using the web URL.

https://github.com/moshifr/git-s3.git

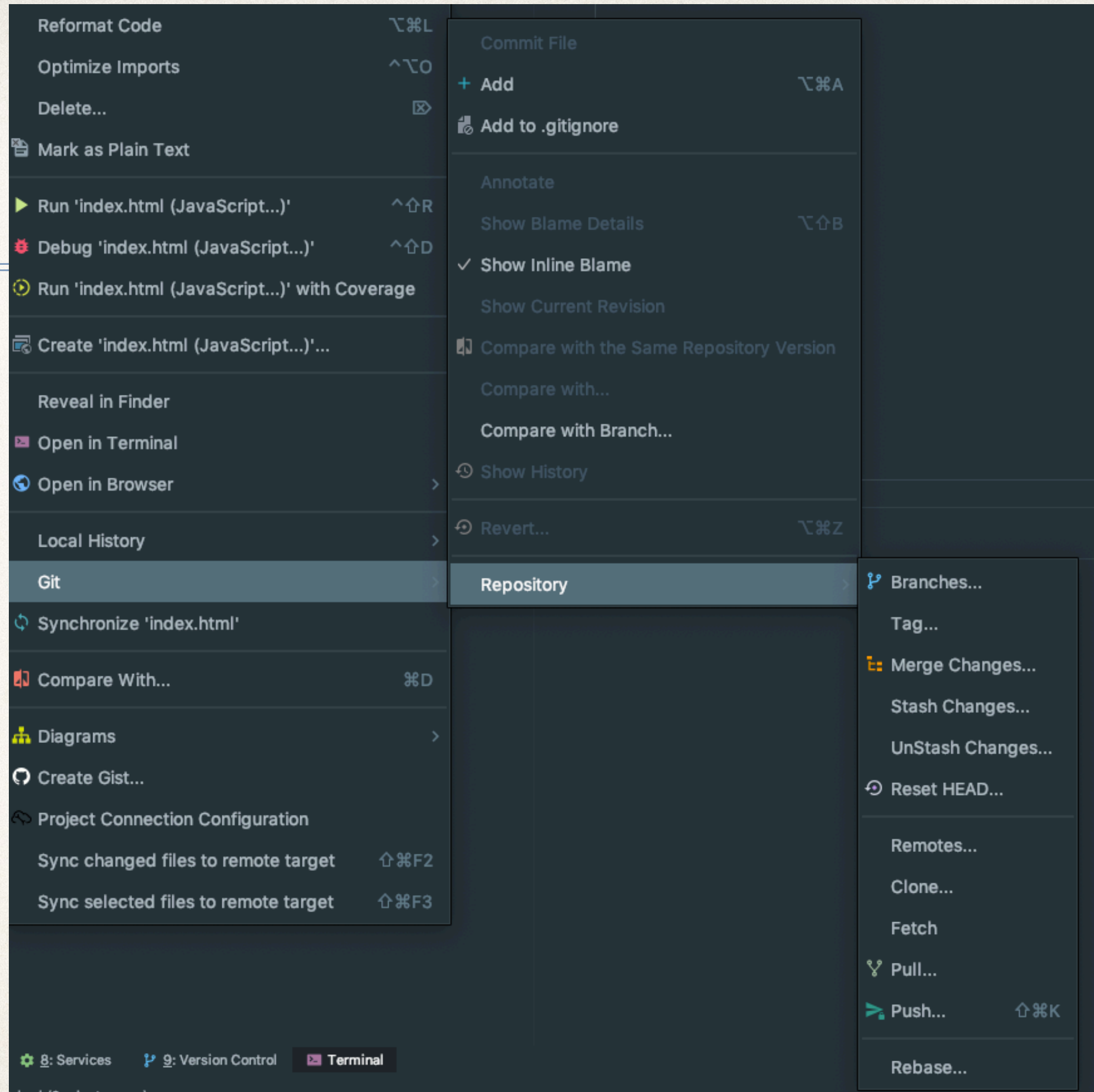
Open in Desktop Download ZIP

PhpStorm

- ❖ Copier et coller l'url du dépôt sur PhpStorm:



Add / Commit




Push

In case of fire



 **1. git commit**

 **2. git push**

 **3. leave building**

Git flow

- ❖ Voici les étapes obligatoires quand on travaille sur un dépôt GIT :
- ❖ - Vérifier la branche courante `Git: master` (ne jamais pousser sur la branche master)
- ❖ - Récupérer la dernière version de la branche (git pull)
- ❖ - Faites vos modifications
- ❖ - Ajouter les fichiers modifiés (git add)
- ❖ - Valider les changements (git commit -m « mon message »)
- ❖ - Envoi des changements (git push origin **<branch>**)
- ❖ - Si validation client de la branche on peut réaliser un pull request

Git flow

- ❖ Voici les étapes obligatoires quand on travaille sur un dépôt GIT :
- ❖ - Vérifier la branche courante `Git: master` (ne jamais pousser sur la branche master)
- ❖ - Récupérer la dernière version de la branche (git pull)
- ❖ - Faites vos modifications
- ❖ - Ajouter les fichiers modifiés (git add)
- ❖ - Valider les changements (git commit -m « mon message »)
- ❖ - Envoi des changements (git push origin **<branch>**)

TP -

- ❖ Par groupe de 3 choisir un élève qui sera le dépôt et 3 développeurs qui vont modifier ce fichier : index.html
- ❖ Chacun rajoute son nom, pseudo, une mini description avec une couleur différente (via une classe css)
- ❖ Faire en sorte que 3 pseudos soit visible sur le dépôt.

TP -

- ❖ Créer une branche avec votre login MMI (feature / mmi99xx)
- ❖ Rajouter votre nom et login MMI dans le fichier index.html
- ❖ Ajouter une classe css dans le fichier style.css pour que la couleur de votre ligne dans le tableau change.
- ❖ Envoyer ces modifications
- ❖ Faire un pull request depuis github