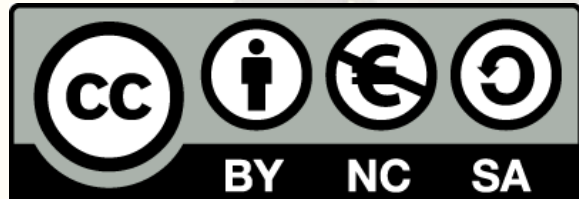


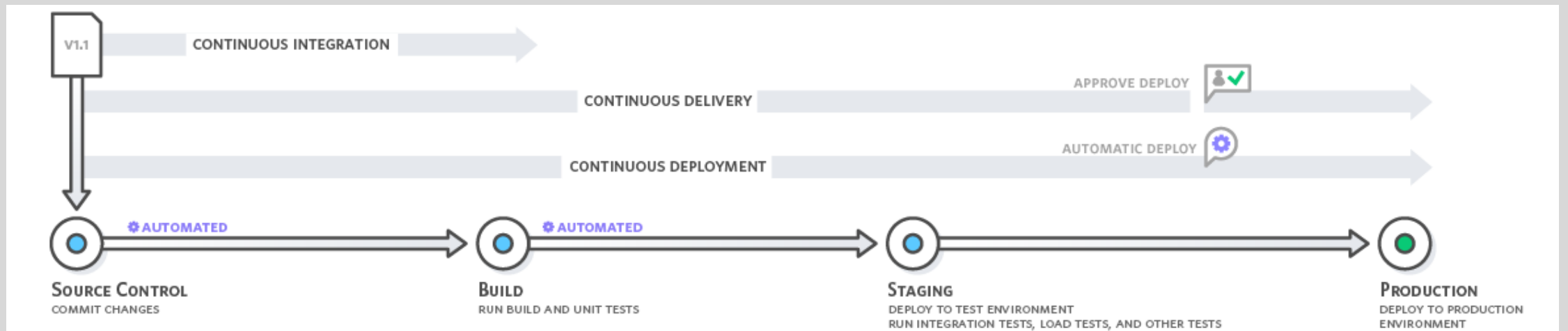
# R3.14 DÉPLOIEMENT DE SERVICES

TP.1 L'outil GIT



# Intégration en continue

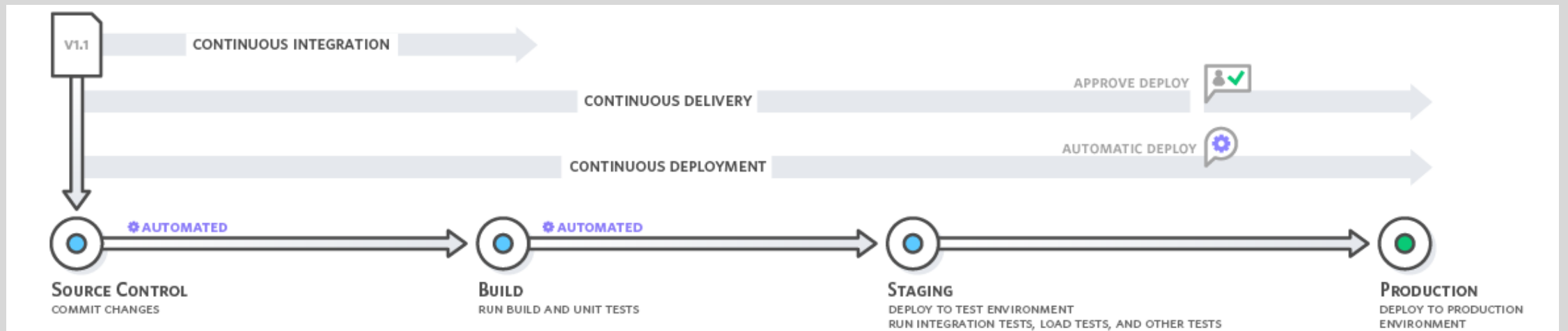
- L'intégration continue est une méthode de développement de logiciel **DevOps** avec laquelle les développeurs intègrent régulièrement leurs modifications de code à **un référentiel centralisé**.



<https://aws.amazon.com/fr/devops/continuous-integration>

# Intégration en continue – 1ère étape

Développement et Intégration WEB

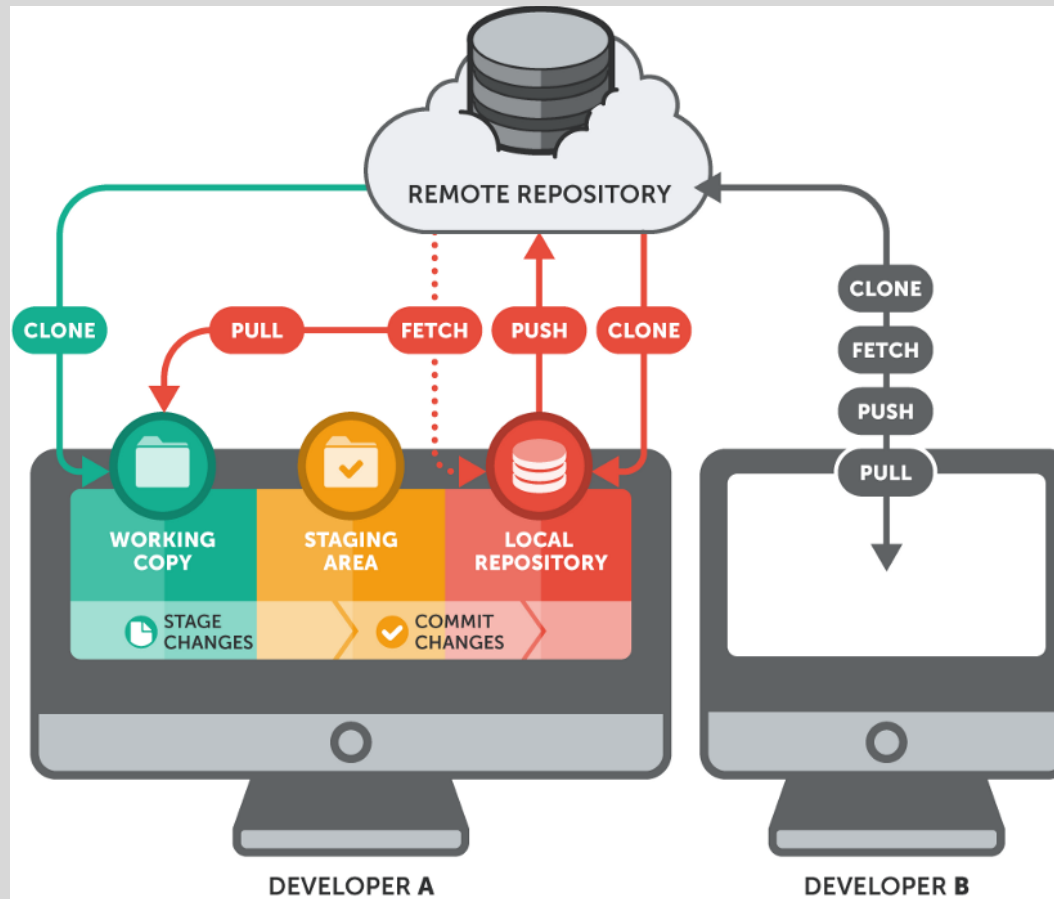


<https://aws.amazon.com/fr/devops/continuous-integration>

# Git

- **Git** est un logiciel de gestion de versions décentralisé. C'est un logiciel libre et gratuit, créé en 2005 par Linus Torvalds, auteur du noyau Linux (système d'exploitation).
- **L'utilisation de GIT permet de :**
  - Supporter le développement en équipe de grands projets.
  - Favoriser la fusion de différentes versions du même projet.
  - Favoriser le déploiement d'une application.
- **GitHub** se trouve parmi les plus grands fournisseurs d'hébergement d'entrepôts Git. Github prévoit des plans gratuits pour des petites équipes (c'est notre cas).

# Git – en une seule image



# Git

- L'idée principale de **Git** c'est d'avoir un entrepôt partagé (dans le cloud – **Remote repository** )
- Une copie de l'entrepôt est régulièrement copiée et stockée en local (**Local repository**)
- Quand nous apportons des modifications à l'entrepôt localement, nous travaillons sur une copie (Working Copy) de l'entrepôt local.
- Modifier l'entrepôt partagé est une opération que nous effectuons en 3 étapes:
  1. **(Add)** Ajoute des modifications dans la « staging area »
  2. **(Commit)** Bascule les éléments de la staging area dans l'entrepôt local.
  3. **(Push)** Met à jour de l'entrepôt partagé avec les éléments de l'entrepôt local.

# TP avec Github

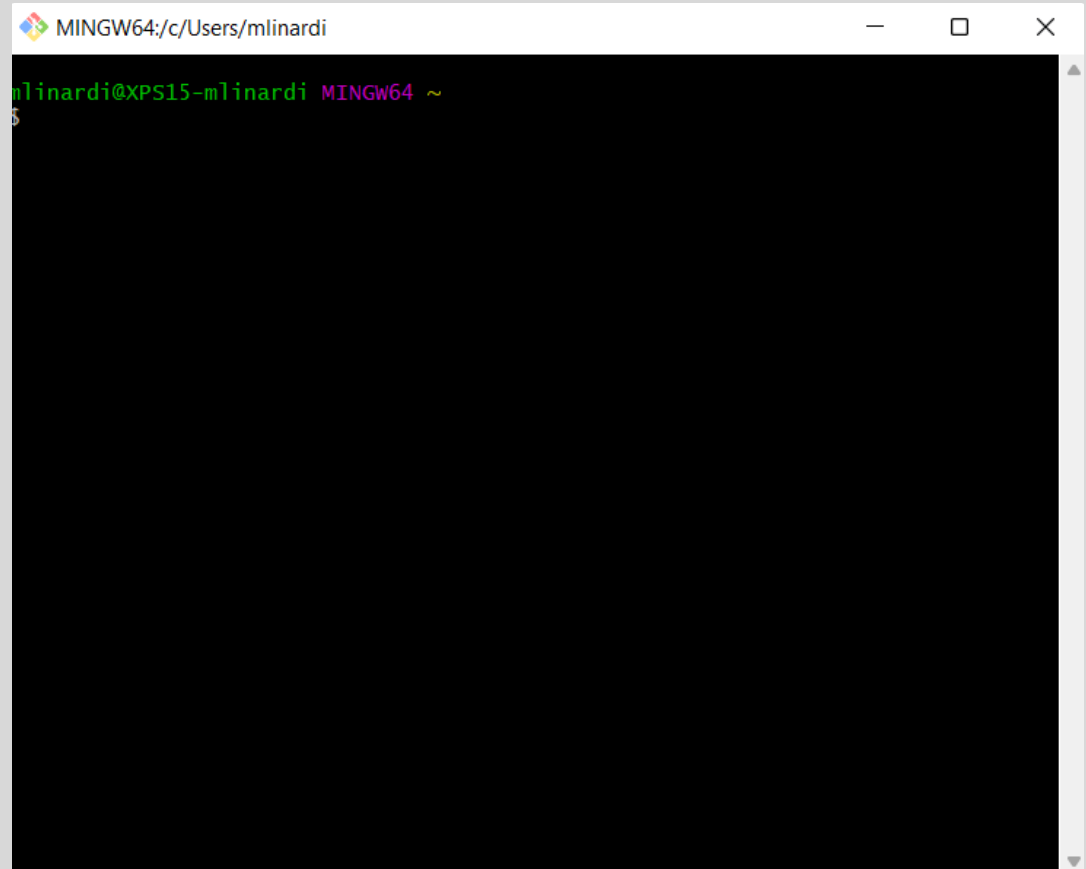
- Vous avez crée....

# Objective du TP: travailler avec un entrepôt git

- 1 Installer la Git Bash : invite de commande GIT

Ceci est le client que nous utiliserons pour gérer nos entrepôts (mis à jour, gestion des versions, visualisation d'état d'un entrepôt et gestion des conflits)

<https://gitforwindows.org/>





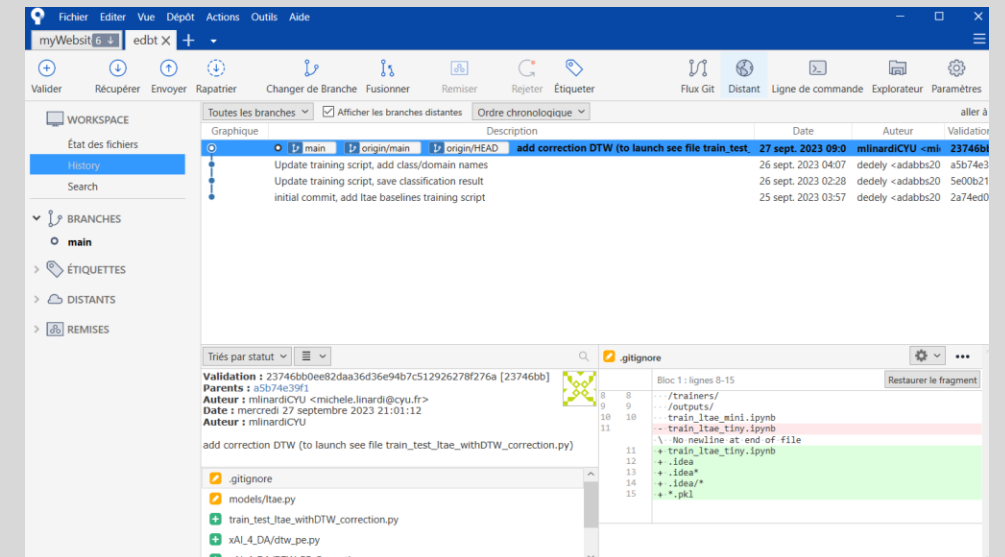
# Objective du TP: travailler avec un entrepôt git

- 1 Installer le logiciel SourceTree



<https://www.sourcetreeapp.com/> (Windows et Mac OS X)

Des autres solutions sont disponible (cherchez sur le web)



# Avant de commencer

- C' est fortement conseillé de jeter un coup d'œil à la guide (doc) de GIT (disponible aussi tout au long du TP):
- <https://git-scm.com/docs>
- <https://education.github.com/git-cheat-sheet-education.pdf>
- ...

# Dans ce TP on vous demande de...

(en utilisant l'invite de commande Git (Git bash) ou un logiciel avec interface graphique (par exemple SourceTree))

- Créer un entrepôt sur GitHub (dans votre navigateur)
- Regarder le « status » (état) d'un entrepôt git. (?)
- Ajouter des éléments (?)
- Exécuter un commit (?), envoyer le commit (?), mettre à jour l'entrepôt (?)
- Créer un « branch » (branche : version parallèle de l'entrepôt) et travailler sur la branche (répétez le point précédent). Regardons d'abord le concept de branche...
- Faire un "merge"(fusion de la branche dans l'entrepôt)
- Gérer les conflits. (?)
- ...(?) signifie que nous avons besoins de trouver la commande correcte (faisons cet exercice ensemble)

# Objective du TP

- Travailler en groupe (Gardez les mêmes groupes du TP 0 )
- Produire un document qui résume toutes les étapes et les commandes utilisées.
- Ajouter les documents dans l'entrepôt (celui où se trouve le rendu du TP 0). Attention , vous devez chacun soumettre 1 rendu. (1 rendu par étudiant)
- Faites bien attention à lancer toutes les commandes, elles devront apparaître dans l'historique de votre entrepôt GIT.