

# Présentation de la formation sur les Systèmes de Gestion de Versions

Florian Legendre

Université de Poitiers

Année 2020 - 2021

# Table of Contents

- 1 Qui suis-je ?
- 2 Pourquoi suivre cette formation ?
- 3 Qu'allez-vous apprendre ?
- 4 Comment allez-vous l'apprendre ?
- 5 Voulez-vous quelques exemples ?

# Qui suis-je ?

Bonjour ! Merci de l'intérêt que vous portez à cette formation et merci d'avoir répondu à l'enquête de besoins. Je m'appelle Florian Legendre et je suis étudiant en dernière année de licence informatique à l'Université de Poitiers.

J'effectue actuellement un stage au laboratoire PPrime sous la responsabilité de M. Domalain et c'est dans le cadre de ce stage que je vous propose cette formation aux outils de gestion des versions.

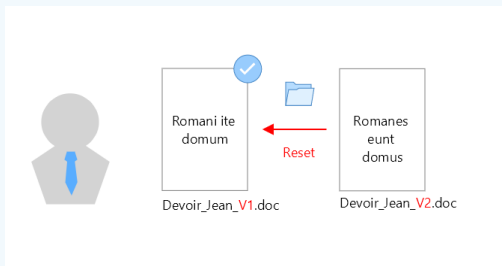


# Pourquoi suivre cette formation ?

Vous avez certainement déjà géré des versions de documents à la main : une thèse, du code, un rapport, un document de recherche...

Pour gérer ces versions vous avez probablement un document principal sur lequel vous écrivez et un dossier dans lequel vous stockez des versions de ce document (Exemple : doc\_V2.docx).

Pour revenir à une version antérieure vous procédez ainsi :



# Pourquoi suivre cette formation ?

Cette méthode a ses avantages mais présente un certain nombre d'inconvénients, surtout quand vous travaillez à plusieurs sur de mêmes documents :

Avantages	Inconvénients
<p>+ Simple</p>	<ul style="list-style-type: none"> <li>- Pas toujours évident de savoir quelle version a apporté quel changement</li> <li>- Prend beaucoup de place !</li> <li>- Devient vite illisible si vous voulez faire des "versions alternatives"</li> <li>- Impossible quand on travaille à plusieurs de savoir qui a fait quoi et si on édite une partie d'un coéquipier</li> <li>- Prend du temps !</li> </ul>

# Qu'allez-vous apprendre ?

L'outil que je souhaite vous faire découvrir est un outil très puissant qui vous permettra de résoudre tous ces problèmes sur n'importe quel type de document texte.

Cet outil s'appelle Git et il vous permettra de :

- ➊ Sauvegarder des versions
- ➋ Garder/Consulter un historique des versions
- ➌ Comparer des versions/Revenir en arrière
- ➍ Faire des branches alternatives de versions
- ➎ Partager des versions
- ➏ Identifier des conflits d'éditions lors de collaborations
- ➐ Savoir à tout moment et très exactement qui a fait quoi

# Qu'allez-vous apprendre ?

Un dernier avantage de Git est qu'il suffit de l'apprendre une fois et vous maîtriserez tous les autres logiciels de gestion de versions du marché !

# Comment allez-vous l'apprendre ?

Pour vous faire découvrir cet outil j'ai choisi une approche :

- 1 Visuelle => Je vous fournirai de nombreux schémas et illustrations qui permettront une acquisition rapide des notions
- 2 Concrète => Les séances, d'une durée de 1h, alterneront entre courts moments de théories et moments de pratiques sur machine
- 3 Centrée sur vos besoins => Très vite nous verrons comment intégrer ce que vous apprendrez sur vos projets en cours



# Voulez-vous quelques exemples ?

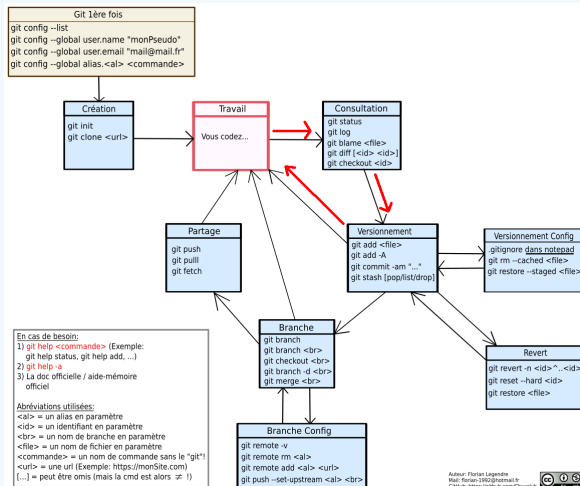
Git est un logiciel utilisant des commandes dont voici un exemple :

```
crex@crex:~/projects/test$ git init  
Initialized empty Git repository in /home/crex/projects/test/.git/
```

**Listing 1** – Contenu du dossier .git/

# Voulez-vous quelques exemples ?

Ces commandes s'utilisent en boucles de travail. Aussi elles sont rapides à acquérir. Voici un exemple de boucle de travail :



# Voulez-vous quelques exemples ?

Un exemple illustrant la détection automatique de conflits d'édition entre plusieurs collaborateurs :

```
<<<<<<< HEAD
VisuRepereMokka( markers.Potence ,Rpotence_R0 ,c3d , 'Rpotence_R0 ' )
=====
VisuRepereMokla( markers.Potence ,Rpotence_R0 ,c3d , 'Rpotence_R0 ' )
>>>>>>> travail_de_monOuma_collegue
```

**Listing 2** – Exemple de détection automatique de conflit d'édition

# Merci !

Merci d'avoir lu cette présentation, et j'espère vous dire à bientôt pour en découvrir davantage !

