

GitHub

Terminologie

Résumé du site : <https://www.christopheducamp.com/2013/12/15/github-pour-nuls-partie-1/>

Git est un logiciel de contrôle de version, ce qui signifie qu'il gère les modifications d'un projet sans écraser n'importe quelle partie du projet.

Pourquoi utiliser quelque chose comme Git ?

Supposons que vous mettiez à jour avec un collègue des pages sur le même site web. Vous faites des modifications, vous les sauvegardez et les versez sur le site. À ce stade, tout va bien.

Le problème survient quand votre collègue travaille sur la même page que vous en même temps. L'un de vous va voir son travail écrasé.

GitHub facilite l'utilisation de Git sur deux points :

1. Si vous [téléchargez le logiciel GitHub](#) sur votre ordinateur, GitHub fournit une interface visuelle pour vous aider à gérer localement vos projets avec les contrôles de version.
2. Créer un compte sur GitHub.com apporte les contrôles de versions à vos projets web, et leur confère des fonctionnalités de réseaux sociaux.

Dépôt : Un répertoire ou de l'espace de stockage où vos projets peuvent vivre. Parfois les utilisateurs GitHub raccourcissent ça en "repo". Il peut être local sur un répertoire de votre ordinateur, ou ce peut être un espace de stockage sur GitHub ou un autre hébergeur en ligne. À l'intérieur d'un dépôt, Vous pouvez conserver des fichiers de code, des fichiers texte, des images.

Contrôle de Version : Fondamentalement, l'objectif pour lequel Git a été conçu. Quand vous avez un fichier Microsoft Word, vous l'écrasez à chaque fois que vous faites une nouvelle sauvegarde, ou vous sauvegardez plusieurs versions. Avec Git, vous n'êtes plus obligé de faire ça. Git conserve des "instantanés" de chaque point dans l'historique d'un projet, de sorte que vous ne pouvez jamais le perdre ou l'écraser.

Commit : C'est la commande qui donne à Git toute sa puissance. Quand vous "committez", vous prenez un "instantané", une "photo" de votre dépôt à ce stade, vous donnant un point de contrôle que vous pouvez ensuite réévaluer ou restaurer votre projet à un état précédent.

Branche : Comment plusieurs personnes travaillant sur un projet en même temps sans que Git ne s’embrouille ? Habituellement, elles se “débranchent” du projet principal avec leurs propres versions complètes des modifications qu’elles ont chacune produites de leur côté. Après avoir terminé, il est temps de “fusionner” cette branche pour la ramener vers la branche “master”, le répertoire principal du projet.

Videos

<https://youtu.be/kFix7UDJ7LA>

<https://youtu.be/om42hY4A5Qg>

Site

<https://blog.wppusher.com/github-and-sourcetree-throwdown/>

<https://dbatools.io/join-us/github/>

Branch

<https://youtu.be/6cLIX0ZqO5Q>

<https://youtu.be/zgxaNRA7e1Q>

GITHUB-GIT-GITHUB DESKTOP CLIENT

Créer un compte sur GitHub

www.github.com 

Installer git pour utiliser avec github desktop

<https://git-scm.com/download/win> 

GitHub utilise l'adresse e-mail que vous avez définie dans votre configuration Git locale pour associer des «commits» à votre compte GitHub.

Définir la configuration locale de Git

<https://help.github.com/articles/setting-your-email-in-git/> 

Installer Github Desktop

<https://desktop.github.com/> 

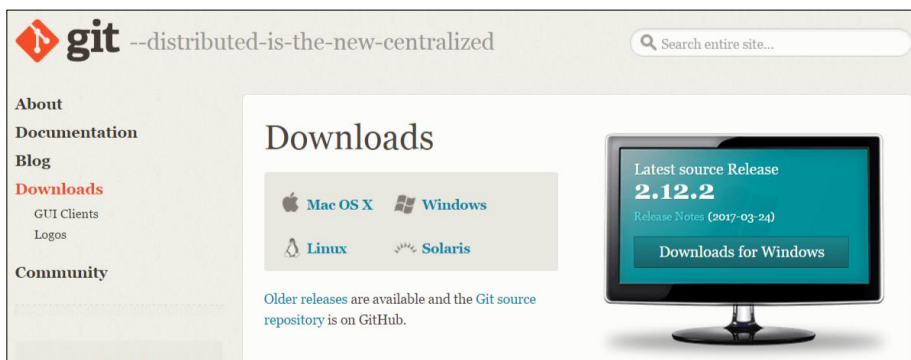
AU BESOIN

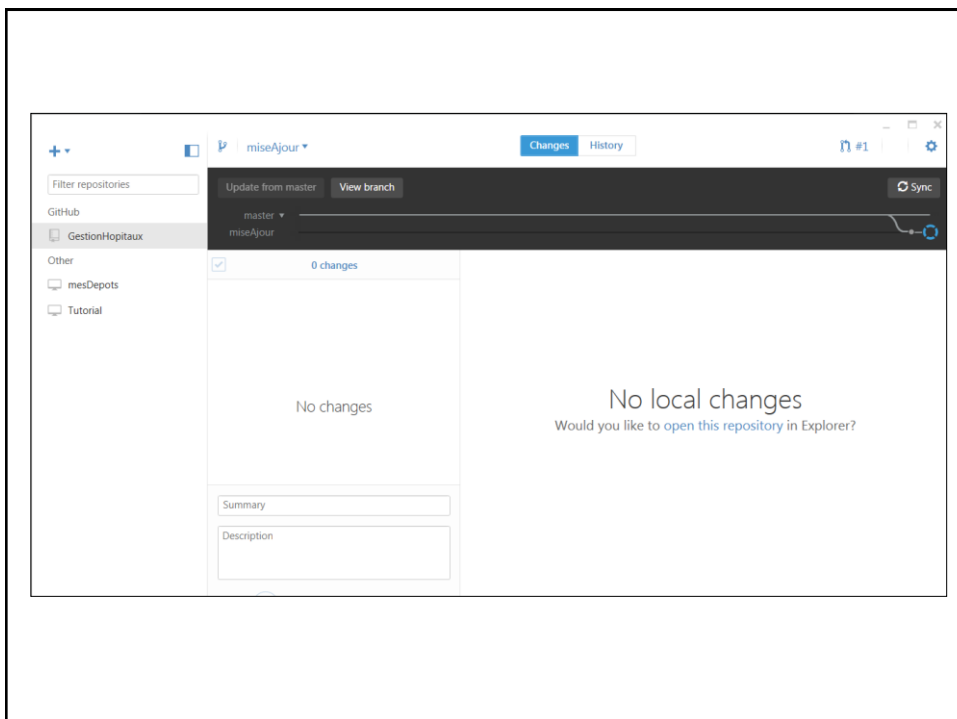
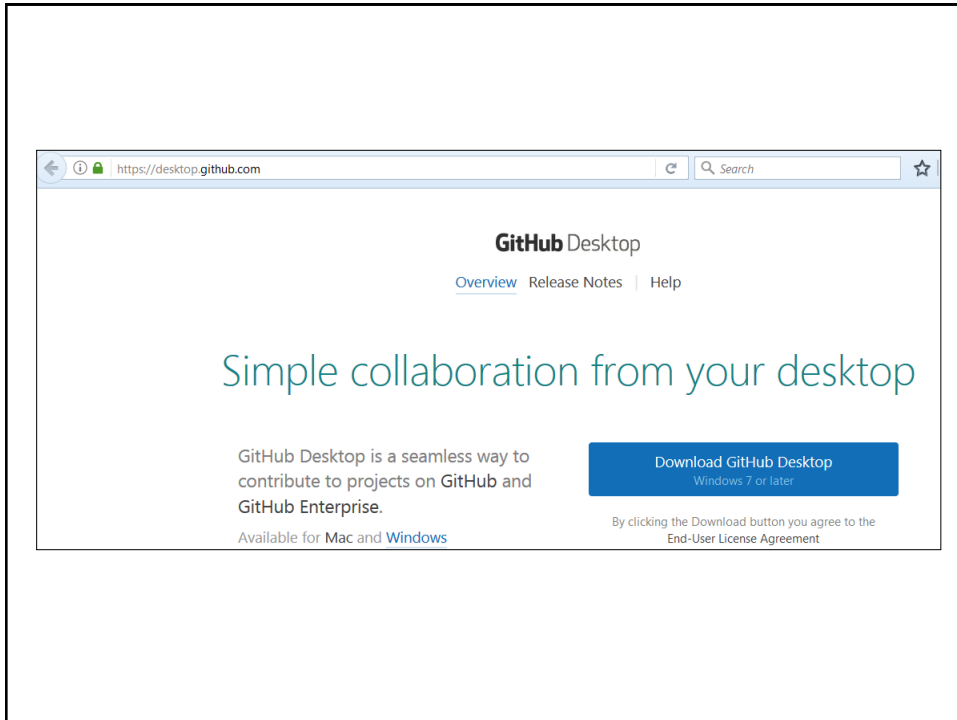
Documentation

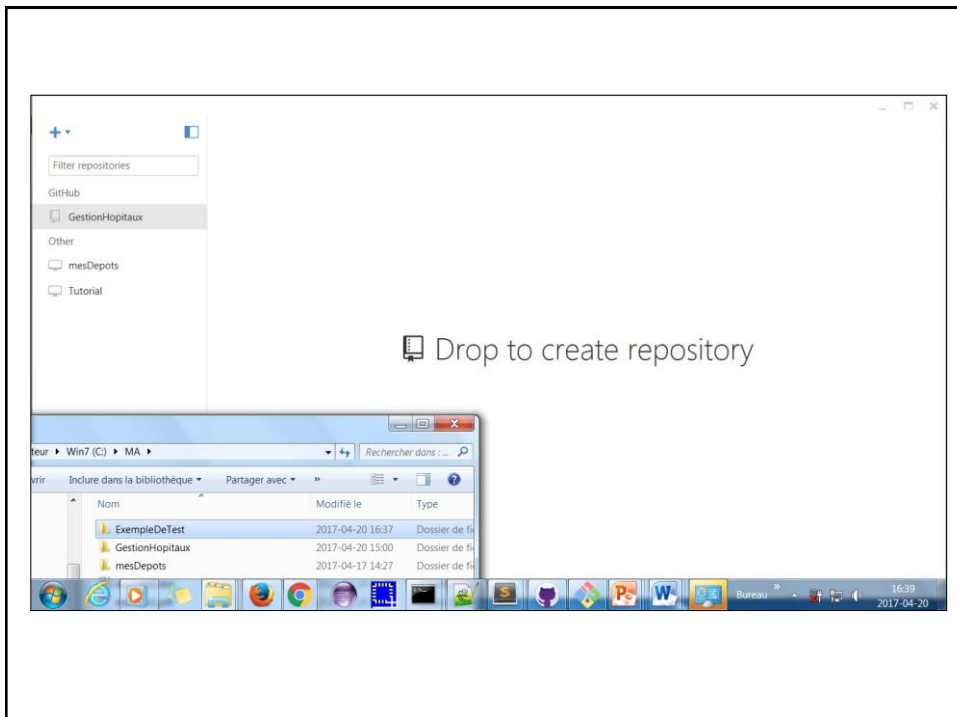
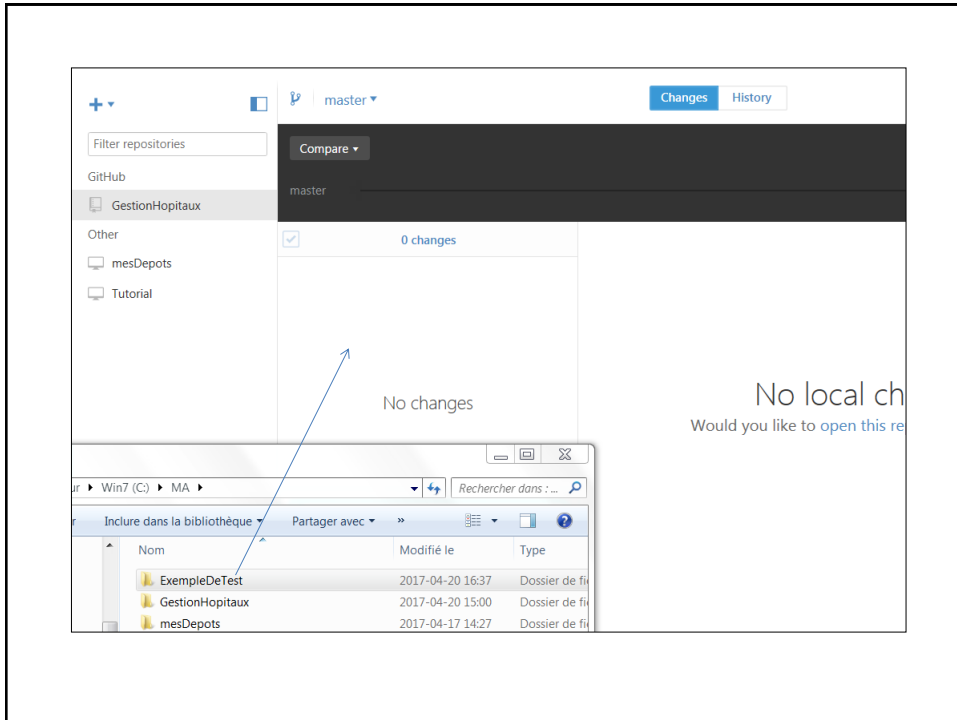
<https://help.github.com/desktop/guides/getting-started/>

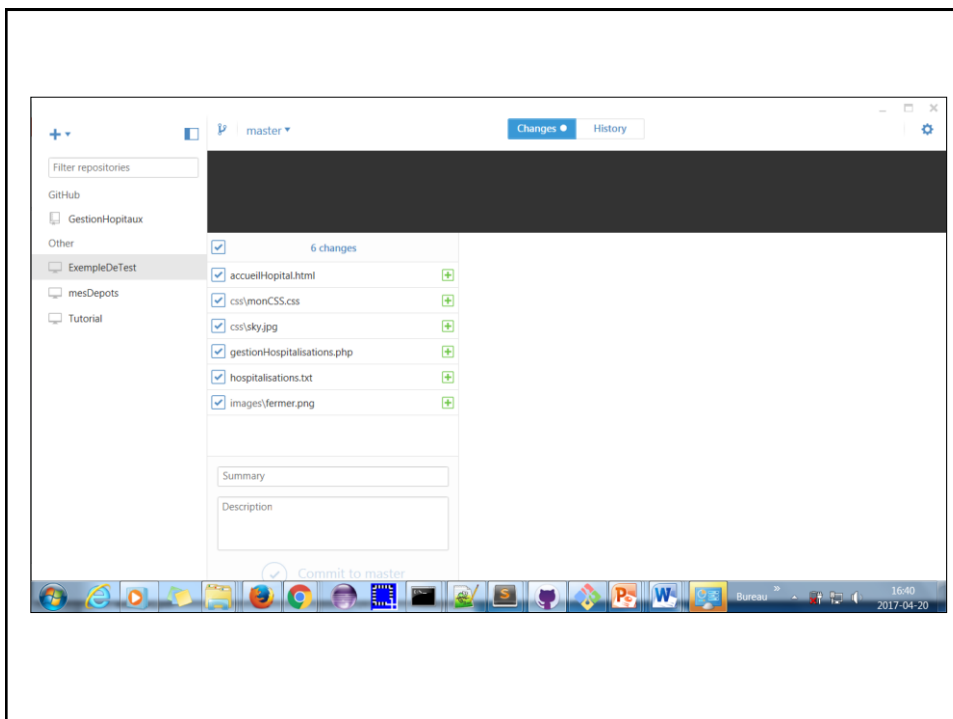
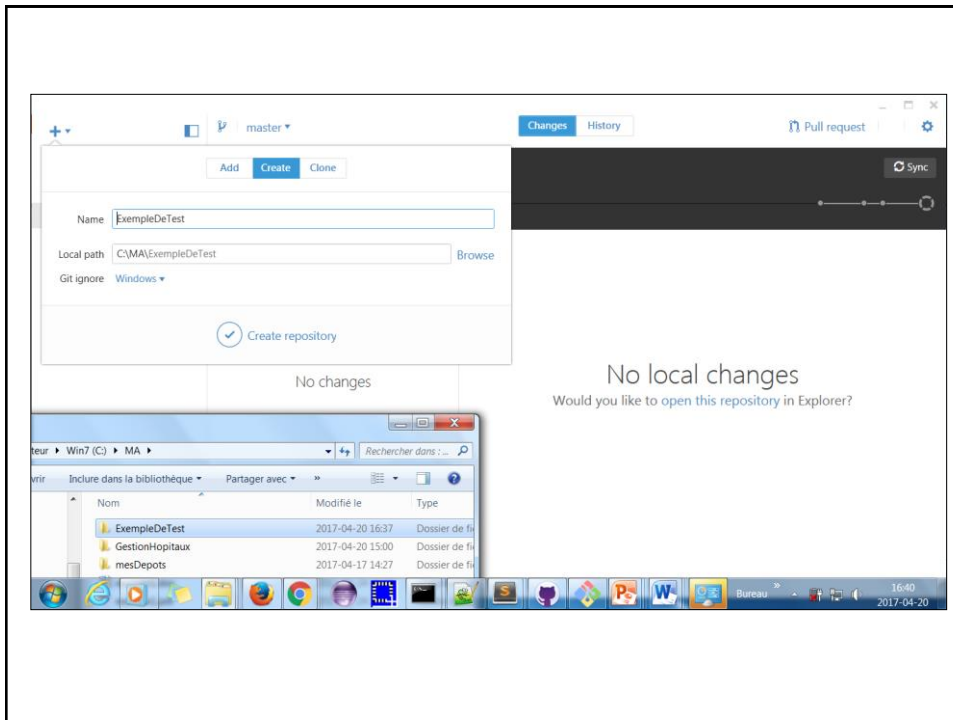
HowTo

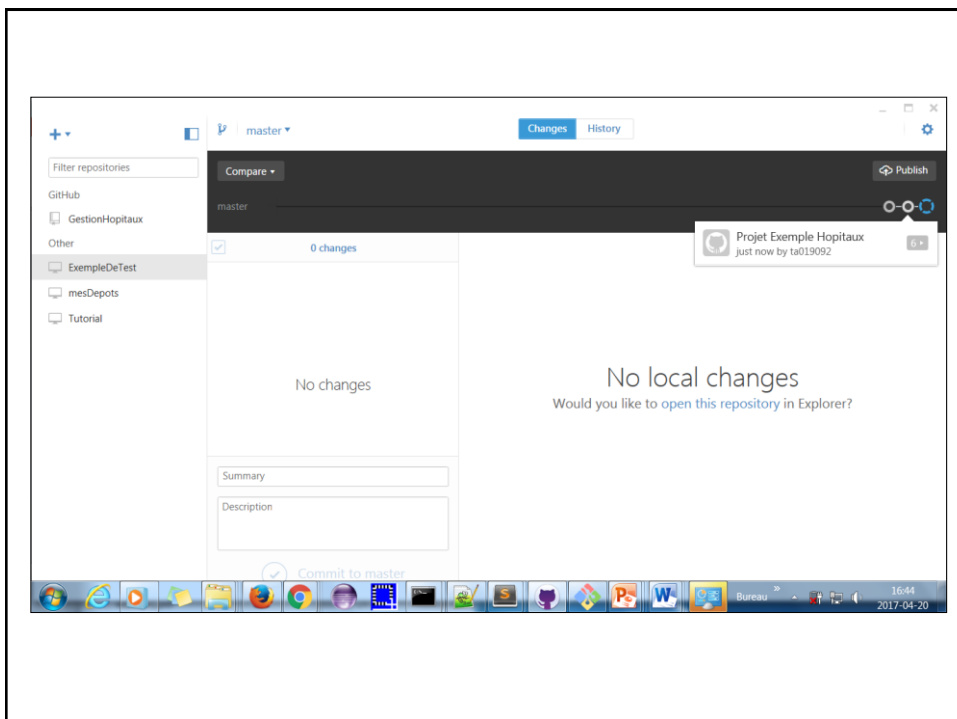
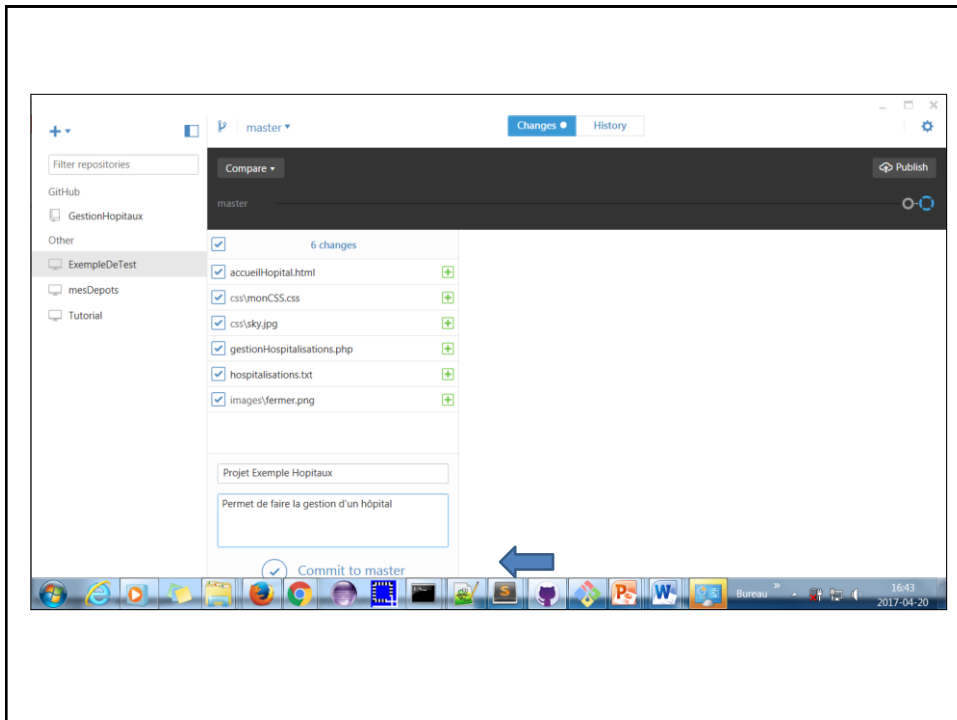
<https://guides.github.com/introduction/getting-your-project-on-github/>

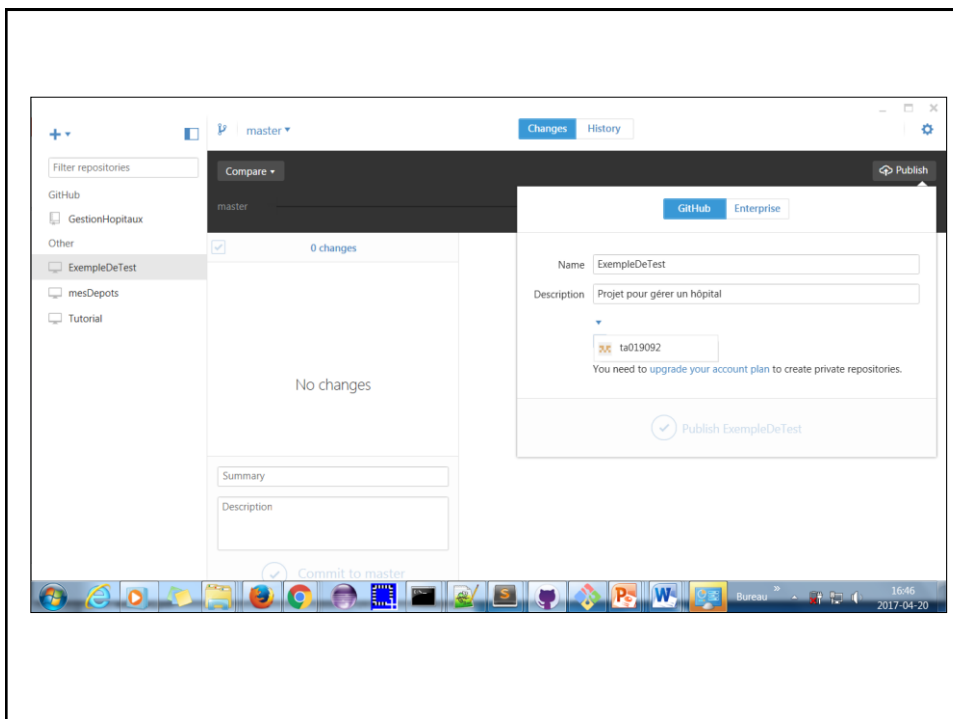
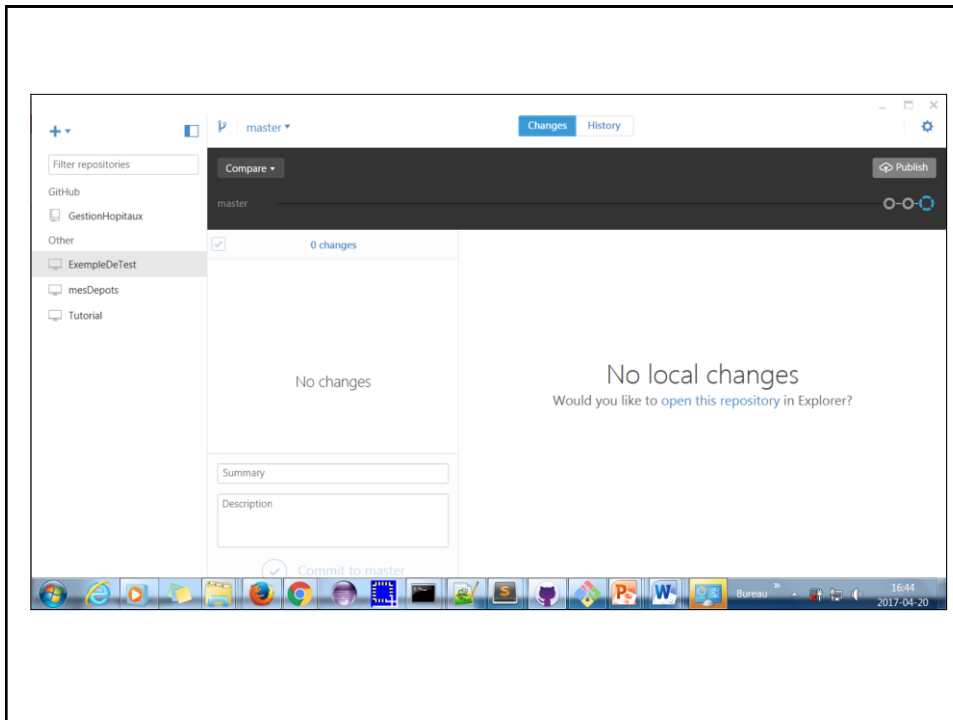


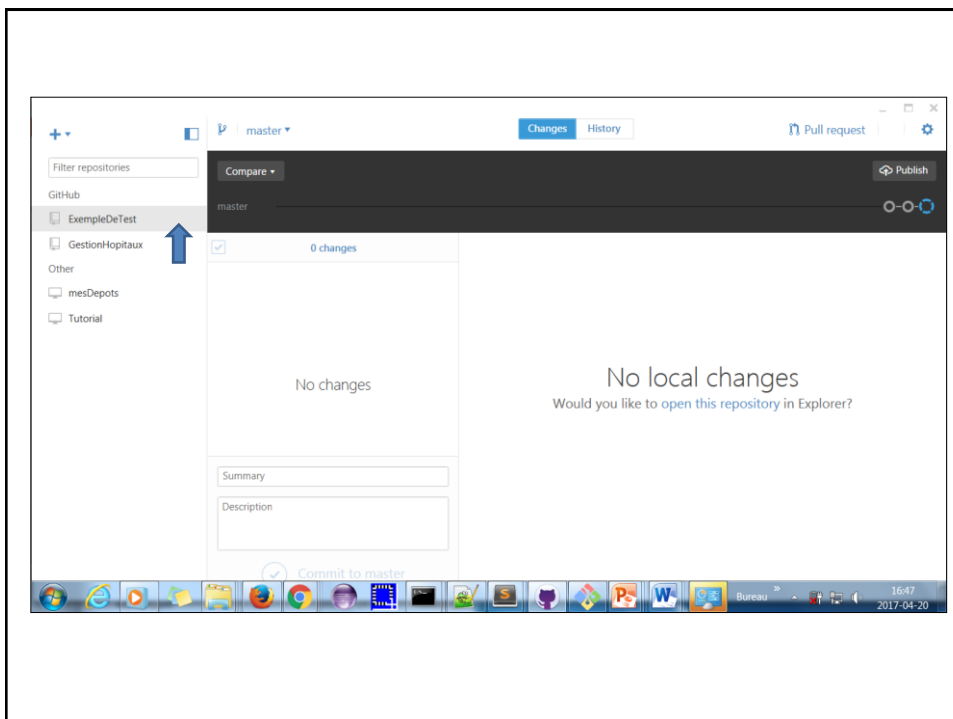
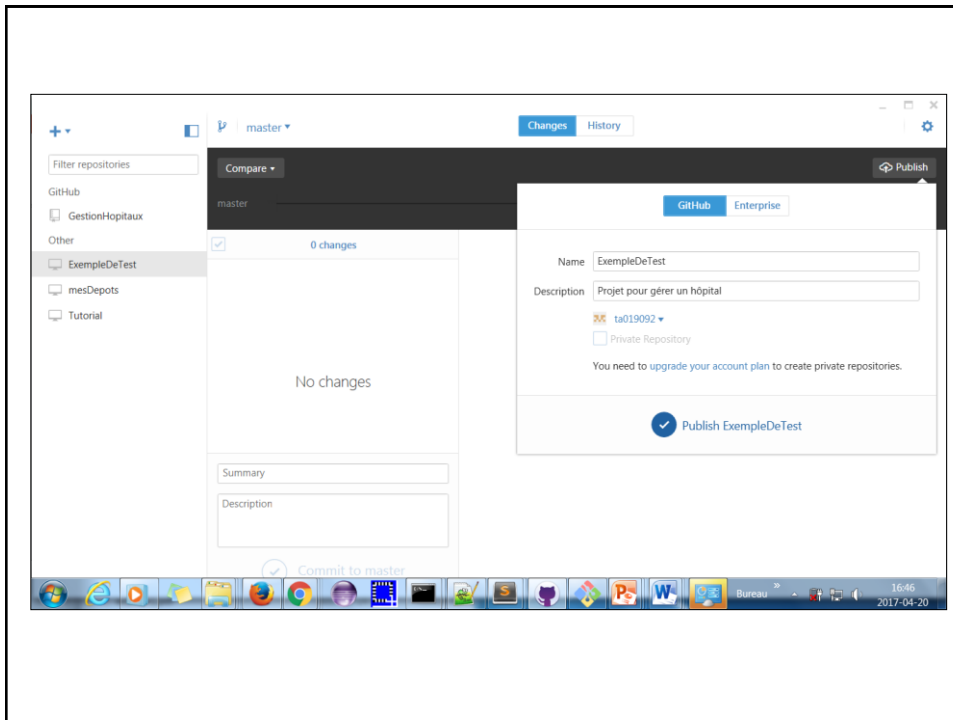












Sur GitHub

ta019092

Overview Repositories 8 Stars 0 Followers 1 Following 0

ta019092/GestionHopitaux 3 commits

ta019092/ExempleDeTest 2 commits

ta019092/Hopitaux 1 commit

Created 3 repositories

ta019092/ExempleDeTest • PHP Apr 20

ta019092/GestionHopitaux • PHP Apr 18

ta019092/Hopitaux Apr 17

Add topics

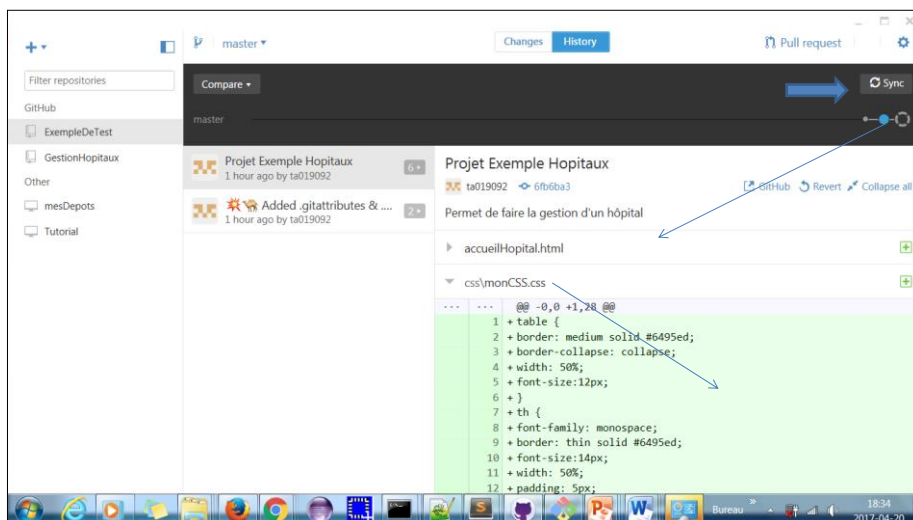
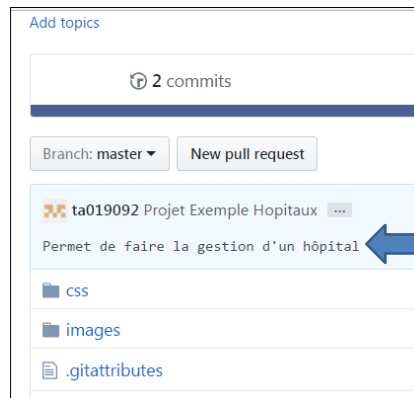
2 commits 1 branch 0 releases 1 contributor

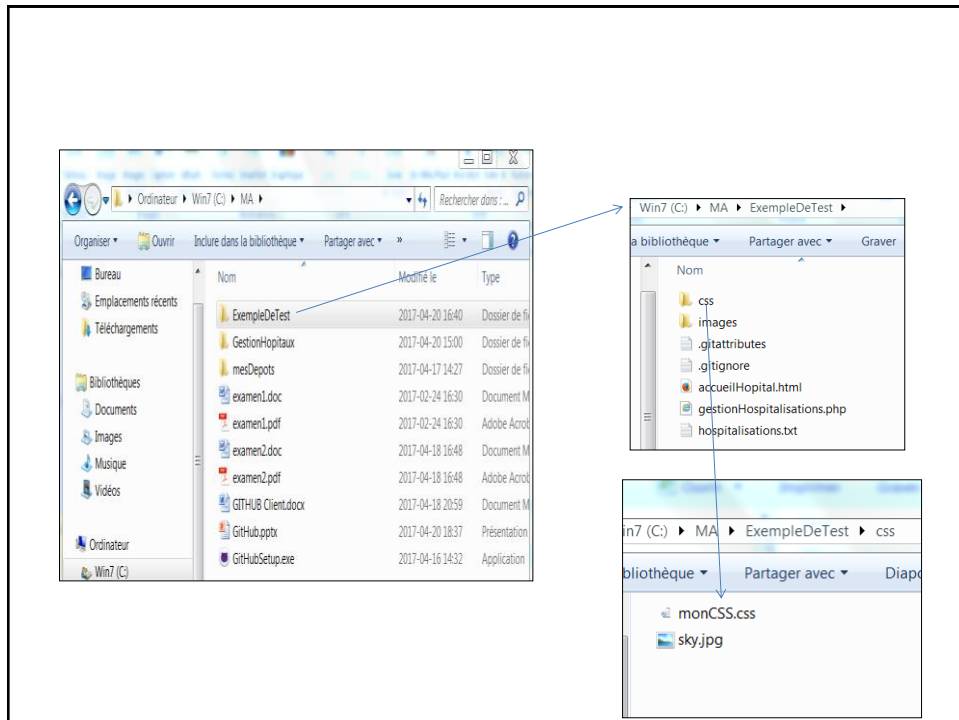
Branch: master New pull request Create new file Upload files Find file Clone or download

ta019092 Projet Exemple Hopitaux Latest commit 6f6ba3 9 minutes ago

File	Commit	Timestamp
css	Project Exemple Hopitaux	9 minutes ago
images	Project Exemple Hopitaux	9 minutes ago
.gitattributes	Added .gitattributes & .gitignore files	12 minutes ago
.gitignore	Added .gitattributes & .gitignore files	12 minutes ago
accueilHopital.html	Project Exemple Hopitaux	9 minutes ago
gestionHospitalisations.php	Project Exemple Hopitaux	9 minutes ago
hospitalisations.txt	Project Exemple Hopitaux	9 minutes ago

Help people interested in this repository understand your project by adding a README. Add a README





Apportons de modifications au fichier **monCSS.css**

Avant

```

1  etable {
2  border: medium solid #6495ed;
3  border-collapse: collapse;
4  width: 50%;
5  font-size: 12px;
6  }
7  eth {
8  font-family: monospace;
9  border: thin solid #6495ed;
10 font-size: 14px;
11 width: 50%;
12 padding: 5px;
13 background-color: #D0E3FA;
14 background-image: url(sky.jpg);
15 }

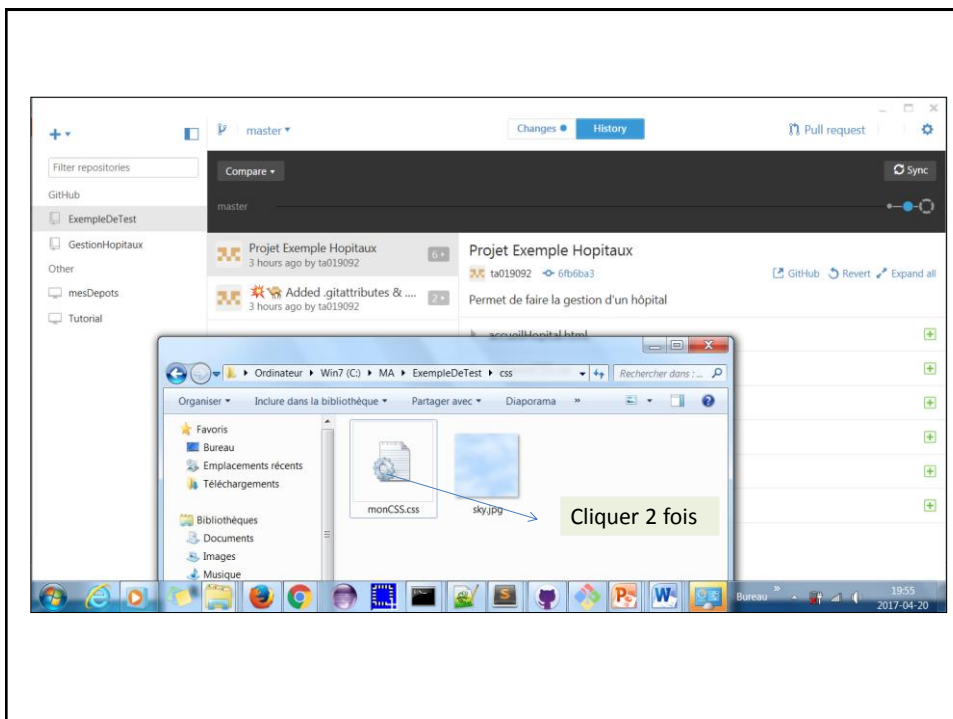
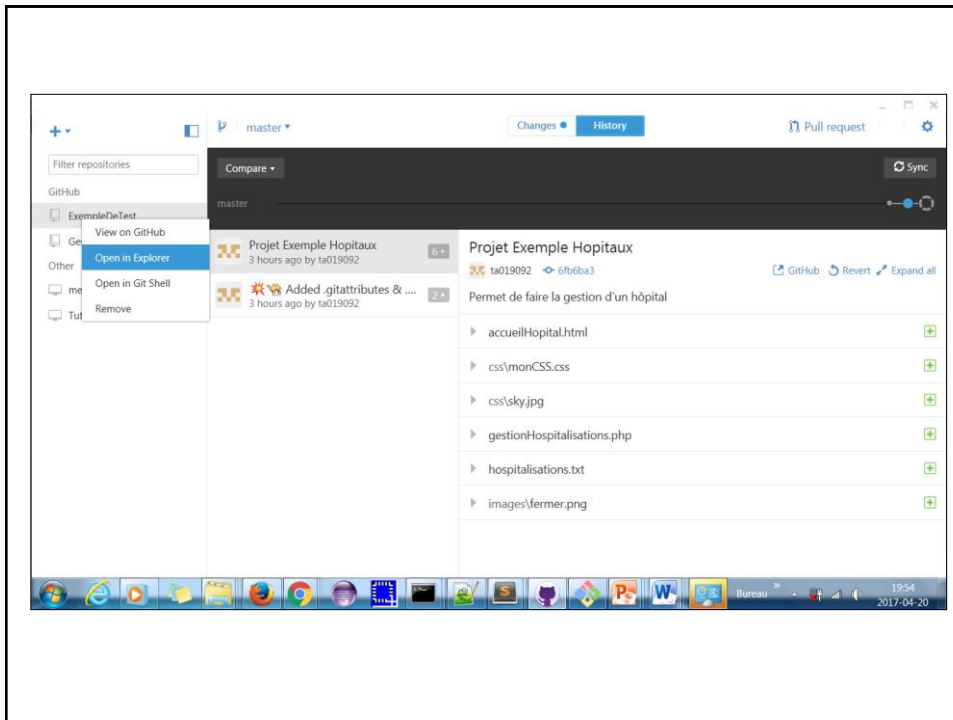
```

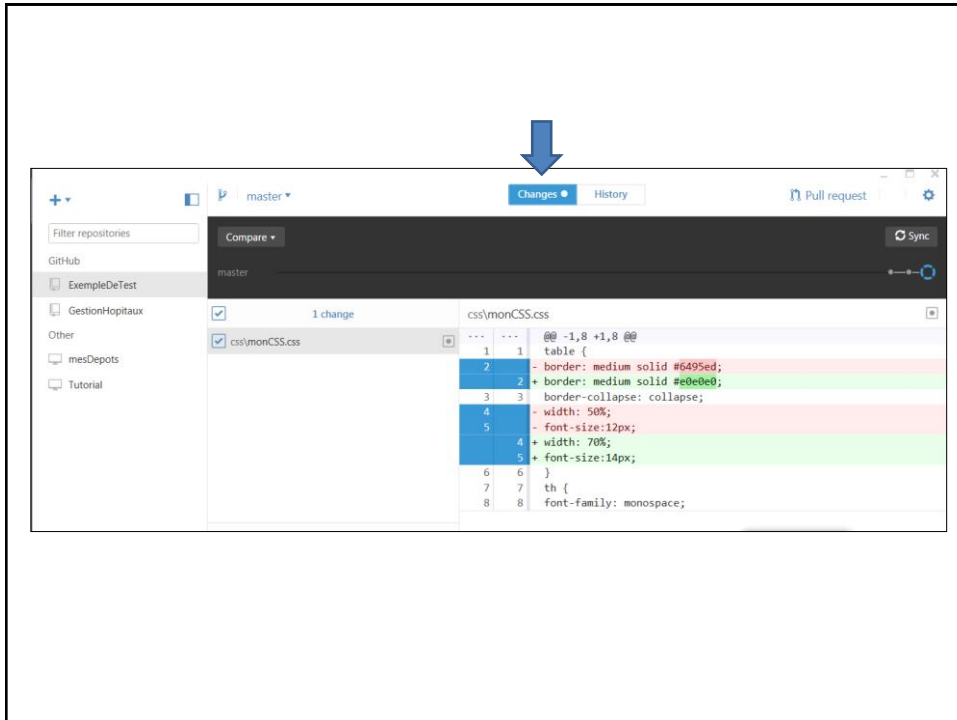
Après

```

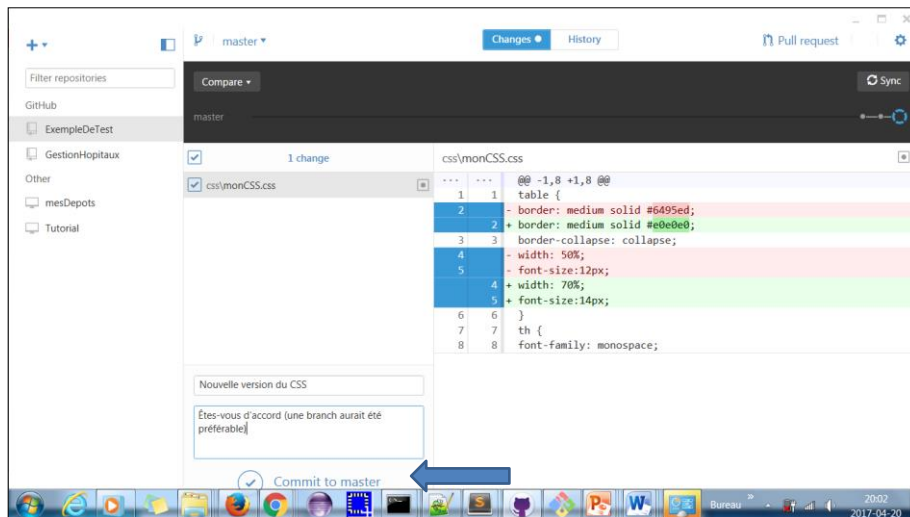
1  etable {
2  border: medium solid #e0e0e0;
3  border-collapse: collapse;
4  width: 70%;
5  font-size: 14px;
6  }
7  eth {
8  font-family: monospace;
9  border: thin solid #6495ed;
10 font-size: 14px;
11 width: 50%;
12 padding: 5px;
13 background-color: #D0E3FA;
14 background-image: url(sky.jpg);
15 }

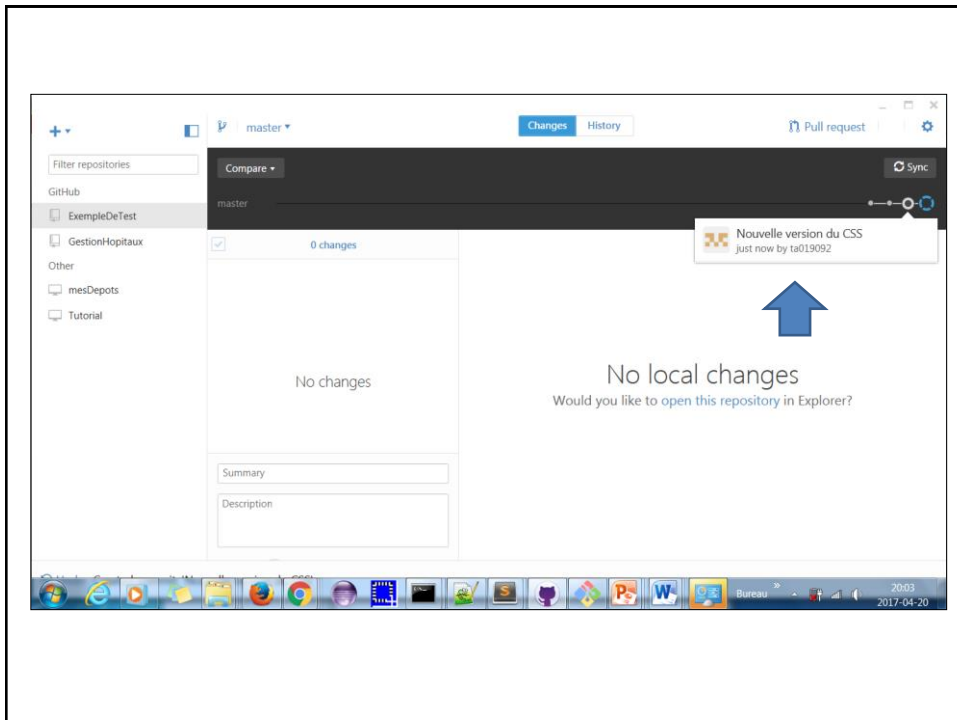
```





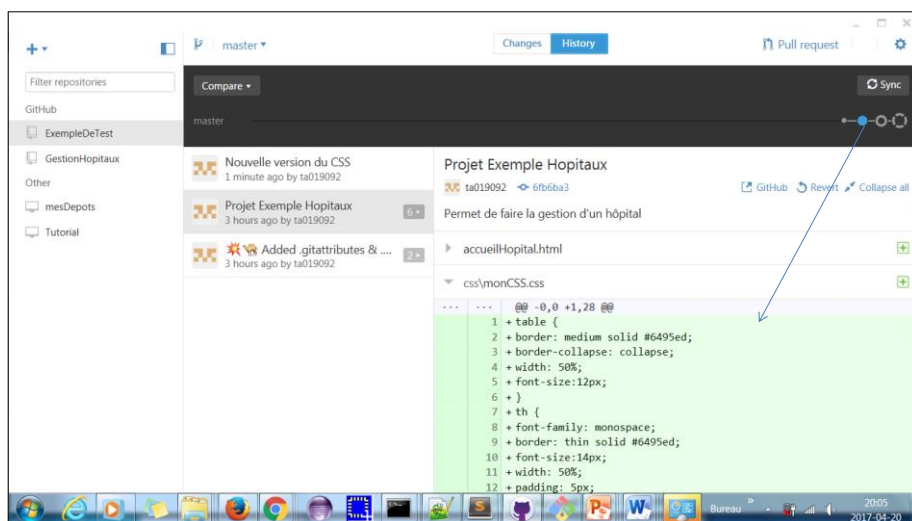
Dans le master il est encore ancien



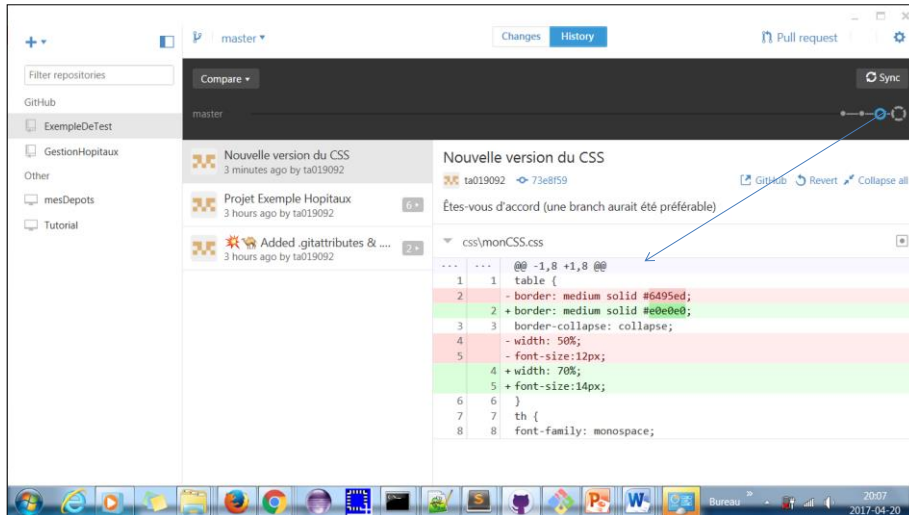


Dans le master

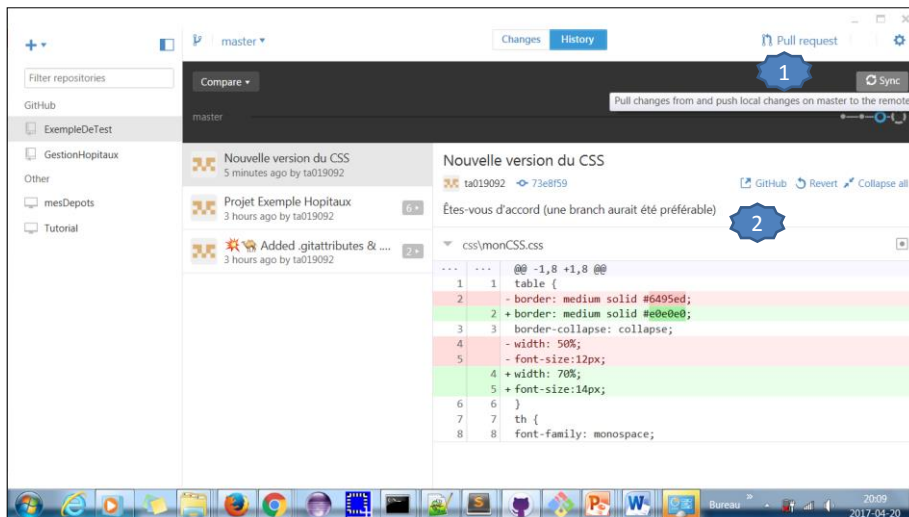
Ancien état du projet

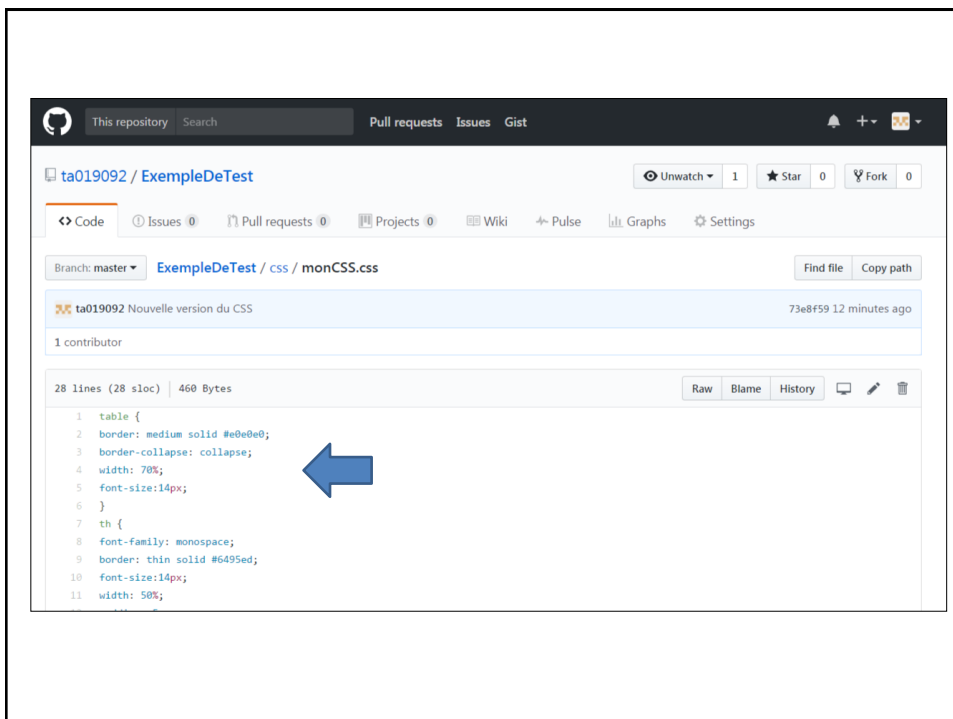
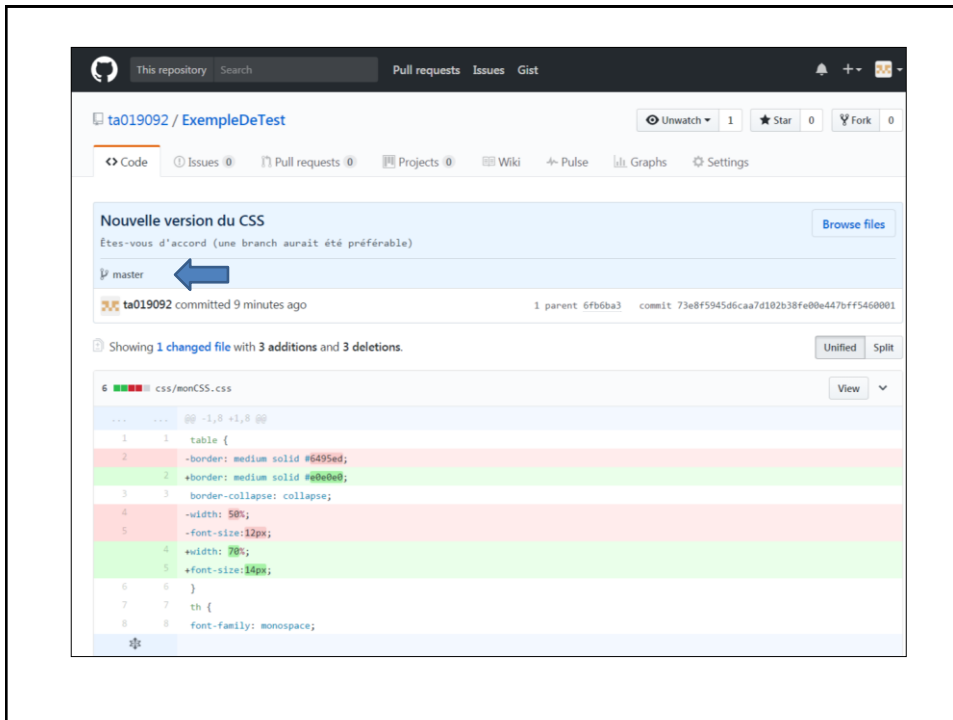


Nouveau état



Ne pas oublier de synchroniser avec GitHub





GITHUB Branches

<https://www.youtube.com/watch?v=6cLIX0ZqO5Q>