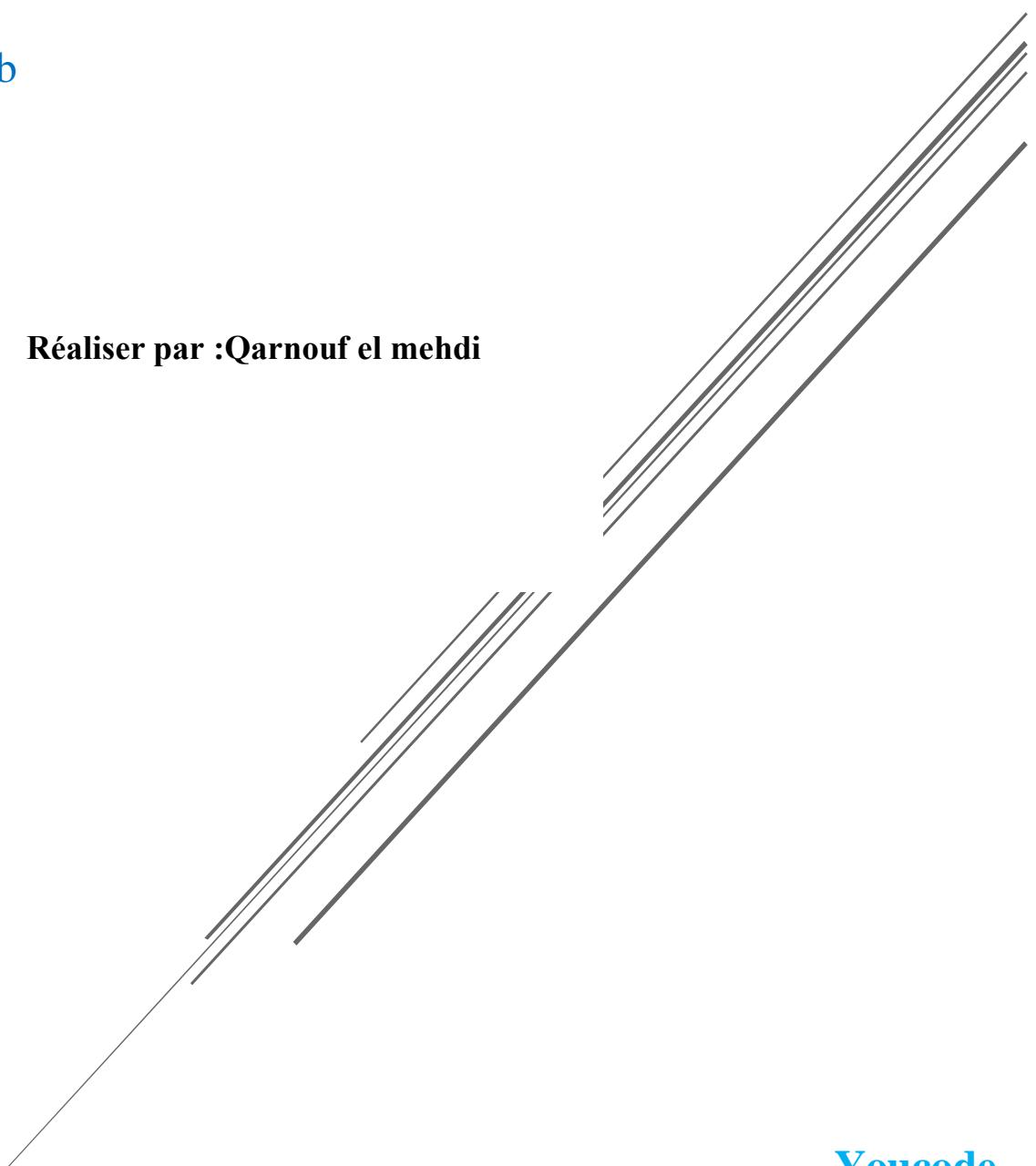


RAPPORT D'ACTIVITÉS DES SCÉNARIOS

Git / GitHub

Réaliser par :Qarnouf el mehdi



Youcode

Introduction général

Dans le cadre de la validation des compétences de la période SAS ; le briefe projet est un moyen utile pour que vous validiez les compétences dans leur niveau respectif (N1, N2, N3)

La gestion des workflows sous GIT/GITHUB sera la base de notre apprentissage en terme de la gestion de projet agile

❖ Objectifs :

- ❖ La formation d'un groupe de 5 apprenants
- ❖ Planification des tâches selon une méthode de gestion de projet (SCRUM).
- ❖ La finalisation des Scénario dans les délais.
- ❖ La rédaction d'un rapport d'activités du groupe.
- ❖ La manipulation des commandes GIT.
- ❖ La découverte du monde GITHUB
- ❖ La gestion du temps, du stress et la finalisation des livrables.
- ❖ Les réunions de fin de Scénario
- ❖ La formalisation

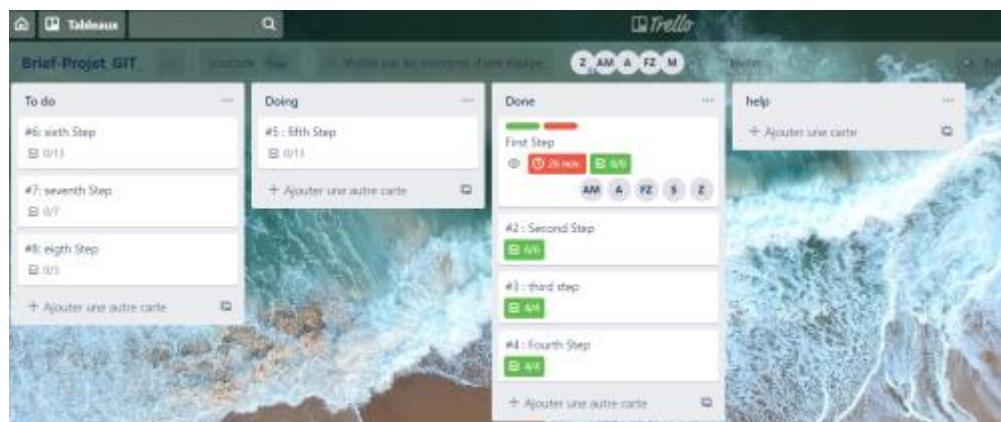
Scénario #1

❖ documentation sur trello

Trello est un outil très simple en apparence mais quand vous y regardez de plus près, vous découvrez à quel point il est dynamique et efficace, quel que soit le projet. Des millions de personnes et d'entreprises de toutes sortes et de toutes tailles utilisent Trello comme outil pour tout gérer : communication des équipes au travail, processus d'intégration en entreprise, support technique, organisateur de tâches hebdomadaires et même planification des repas.

Trello regorge de fonctionnalités, de raccourcis et de Power-Ups qui vont booster votre efficacité et votre productivité au travail comme à la maison. Vous souhaitez savoir comment fonctionne Trello, ou vous êtes déjà un fervent utilisateur mais vous souhaitez passer la 5e et apprendre à bien utiliser Trello à son plein potentiel? Alors, en voiture !

❖ Organisez vos tâches sur trello



❖ création d'un fichier drive pour partager les informations de chacun

https://docs.google.com/document/d/1jsyFnNvmueArItcUg1bQ2i_7w-lt6z3fJrMat3xd-w0/edit?ts=5ddcff96

❖ capture d'écran git bash pour expliquez les commandes

First Step : Immersion

➤ Définissez git en quelques mots :

Git est un logiciel de gestion de versions décentralisé. C'est un logiciel libre créé par Linus Torvalds, le créateur du noyau Linux, et distribué selon les termes de la licence publique générale GNU version 2.

➤ Expliquez le fonctionnement du fichier caché « .git »

Les éléments cachés sont des éléments qui ne sont pas visibles par défaut lorsque l'utilisateur affiche le contenu d'un dossier. Cette possibilité de dissimuler des éléments permet de protéger des fichiers de manipulations involontaires et d'alléger l'affichage des dossiers dans lesquels ils sont stockés. Les éléments cachés se reconnaissent au point précédent leur nom. Un fichier nommé ".test" sera donc reconnu par le système comme un fichier caché alors qu'un fichier nommé "test" ne le sera pas.

```
Youcode@DESKTOP-R41R7JE MINGW64 ~/Desktop/projects
$ mkdir demo

Youcode@DESKTOP-R41R7JE MINGW64 ~/Desktop/projects
$ cd demo

Youcode@DESKTOP-R41R7JE MINGW64 ~/Desktop/projects/demo
$ git status
fatal: not a git repository (or any of the parent directories): .git

Youcode@DESKTOP-R41R7JE MINGW64 ~/Desktop/projects/demo
$ echo "#Demo project un simple fichier" >> README.md

Youcode@DESKTOP-R41R7JE MINGW64 ~/Desktop/projects/demo
$ git init
Initialized empty Git repository in C:/Users/Youcode/Desktop/projects/demo/.git/
Youcode@DESKTOP-R41R7JE MINGW64 ~/Desktop/projects/demo (master)
$ git add README.md
warning: LF will be replaced by CRLF in README.md.
The file will have its original line endings in your working directory

Youcode@DESKTOP-R41R7JE MINGW64 ~/Desktop/projects/demo (master)
$ git commit -m "premiere commit"
[master (root-commit) 0ef4822] premiere commit
 1 file changed, 1 insertion(+)
 create mode 100644 README.md
```

➤ Expliquez le commentaire.

git commit -m "Commit message" : Valider les modifications en tête (mais pas encore dans le référentiel distant)

git commit -a : Validez tous les fichiers que vous avez ajoutés git add, ainsi que tous les fichiers que vous avez modifiés depuis

Second Step : La découverte

- Déplacez-vous dans les fichiers de configuration « .git »

```
create mode 100644 README.md  
yc@DESKTOP-ISOQRAB MINGW64 ~/Desktop/project/demo (master)  
$ cd .git
```

- Tapez Ls -al

```
yc@DESKTOP-ISOQRAB MINGW64 ~/Desktop/project/demo/.git (GIT_DIR!)  
$ ls -al  
total 13  
drwxr-xr-x 1 yc 197121 0 déc. 3 18:54 ./  
drwxr-xr-x 1 yc 197121 0 déc. 3 18:51 ../  
-rw-r--r-- 1 yc 197121 15 déc. 3 18:54 COMMIT_EDITMSG  
-rw-r--r-- 1 yc 197121 130 déc. 3 18:49 config  
-rw-r--r-- 1 yc 197121 73 déc. 3 18:49 description  
-rw-r--r-- 1 yc 197121 23 déc. 3 18:49 HEAD  
drwxr-xr-x 1 yc 197121 0 déc. 3 18:49 hooks/  
-rw-r--r-- 1 yc 197121 137 déc. 3 18:54 index  
drwxr-xr-x 1 yc 197121 0 déc. 3 18:49 info/  
drwxr-xr-x 1 yc 197121 0 déc. 3 18:54 logs/  
drwxr-xr-x 1 yc 197121 0 déc. 3 18:54 objects/  
drwxr-xr-x 1 yc 197121 0 déc. 3 18:49 refs/
```

- Expliquez les clauses suivantes : HEAD, LOGS, BRANCHES

HEAD : est positionné sur une branche qui constitue une référence, et qui est positionnée sur un commit. Lorsqu'un nouveau commit est créé, la branche se positionne sur ce nouveau commit, et HEAD suit la branche

une branche : signifie diverger de la ligne principale de développement et continuer à travailler sans impacter cette ligne

LOGS : Le log de Git est un outil formidable pour analyser l'historique des commits et résituer un contexte. Il nous permet aussi bien de suivre un projet dans sa globalité que dans ses détails

- Retournez vers votre Repo et créez le fichier « Licence.md ».

```
yc@DESKTOP-ISOQRAB MINGW64 ~/Desktop/project/demo/.git (GIT_DIR!)  
$ cd ..  
  
yc@DESKTOP-ISOQRAB MINGW64 ~/Desktop/project/demo (master)  
$ touch Licence.md  
  
yc@DESKTOP-ISOQRAB MINGW64 ~/Desktop/project/demo (master)  
$ ls  
Licence.md  README.md
```

- Faites-le Commit

```
yc@DESKTOP-ISOQRAB MINGW64 ~/Desktop/project/demo (master)
$ git add .

yc@DESKTOP-ISOQRAB MINGW64 ~/Desktop/project/demo (master)
$ git commit -m "deuxième commit"
[master b6c48f8] deuxième commit
 1 file changed, 0 insertions(+), 0 deletions(-)
 create mode 100644 Licence.md
```

- Affichez les fichiers traqués.

```
yc@DESKTOP-ISOQRAB MINGW64 ~/Desktop/project/demo (master)
$ git status
On branch master
nothing to commit, working tree clean
```

third step : Historique

- Affichez le dernier commit sur une ligne en ajoutant l'option d'affichage de la hiérarchie de la branche, avec les commit et leur branche aussi

```
Youcode@DESKTOP-R41R7JE MINGW64 ~/Desktop/projects/demo (master)
$ git log --oneline --graph --decorate --all
* 570ad08 (HEAD -> master) créez le fichier Licence.md
* 0ef4822 première commit

Youcode@DESKTOP-R41R7JE MINGW64 ~/Desktop/projects/demo (master)
$ |
```

- Créez un alias de la commande précédente le nom de l'alias est : historique

```
Youcode@DESKTOP-R41R7JE MINGW64 ~/Desktop/projects/demo (master)
$ $ git config --global alias.historique "log --oneline --graph --decorate --all"
```

- Affichez la liste des alias

```
Youcode@DESKTOP-R41R7JE MINGW64 ~/Desktop/projects/demo (master)
$ git config --get-regexp alias
alias.historique log --oneline --graph --decorate --all

Youcode@DESKTOP-R41R7JE MINGW64 ~/Desktop/projects/demo (master)
$ git config --global -e

Youcode@DESKTOP-R41R7JE MINGW64 ~/Desktop/projects/demo (master)
$ |
```

- Affichez l'historique des commit du fichier README.md avec l'alias.

```
Youcode@DESKTOP-R41R7JE MINGW64 ~/Desktop/projects/demo (master)
$ git historique README.md
* 0ef4822 première commit

Youcode@DESKTOP-R41R7JE MINGW64 ~/Desktop/projects/demo (master)
$ |
```

Fourth Step : Excluding files

- Renommer le fichier Licence.md à Licence.txt

```
Youcode@DESKTOP-R41R7JE MINGW64 ~/Desktop/projects/demo (master)
$ git mv Licence.md Licence.txt

Youcode@DESKTOP-R41R7JE MINGW64 ~/Desktop/projects/demo (master)
$ |
```

- Faites le staging avec mise à jour (ne pas faire « git add . »)

```
Youcode@DESKTOP-R41R7JE MINGW64 ~/Desktop/projects/demo (master)
$ git add -u
```

- Faites-le commit.

```
Youcode@DESKTOP-R41R7JE MINGW64 ~/Desktop/projects/demo (master)
$ git commit -m "Renommer le fichier Licence.md à Licence.txt"
[master 2e3e816] Renommer le fichier Licence.md à Licence.txt
 1 file changed, 0 insertions(+), 0 deletions(-)
 rename Licence.md => Licence.txt (100%)
```

- Créez un fichier nommé application.log

```
Youcode@DESKTOP-R41R7JE MINGW64 ~/Desktop/projects/demo (master)
$ touch application.log
```

- Ne faites pas le staging mais créez un fichier nommé « .gitignore »

```
Youcode@DESKTOP-R41R7JE MINGW64 ~/Desktop/projects/demo (master)
$ touch .gitignore

Youcode@DESKTOP-R41R7JE MINGW64 ~/Desktop/projects/demo (master)
$ |
```

- Sur le fichier .gitignore Ajoutez la ligne suivante « *.log »

```
Youcode@DESKTOP-R41R7JE MINGW64 ~/Desktop/projects/demo (master)
$ vim .gitignore

Youcode@DESKTOP-R41R7JE MINGW64 ~/Desktop/projects/demo (master)
$ |
```

```
MINGW64:/c/Users/Youcode/Desktop/projects/demo
% .log
```

- Faites le staging et le commit

```
Youcode@DESKTOP-R41R7JE MINGW64 ~/Desktop/projects/demo (master)
$ git add .*
The following paths are ignored by one of your .gitignore files:
application.log
Use -f if you really want to add them.
warning: LF will be replaced by CRLF in .gitignore.
The file will have its original line endings in your working directory

Youcode@DESKTOP-R41R7JE MINGW64 ~/Desktop/projects/demo (master)
$ git status
On branch master
Changes to be committed:
  (use "git restore --staged <file>..." to unstage)
    new file:   .gitignore

Youcode@DESKTOP-R41R7JE MINGW64 ~/Desktop/projects/demo (master)
$ git commit -m "Créez un fichier application.log et créez un fichier nommé « .gitignore »"
[master d87949a] Créez un fichier application.log et créez un fichier nommé « .gitignore »
 1 file changed, 1 insertion(+)
 create mode 100644 .gitignore
```

➤ Qu'est-ce que vous constatez ?

ce que nous pouvons comprendre de cette étape est que quel que soit le fichier ou l'extension de fichier que nous avons mis dans .gitignore ne sera ni (staged) ni commitez

fifth Step : Branching and Merging

➤ Modifiez le fichier README.md

```
MINGW64:/c/Users/Youcode/Desktop/projects/demo
#Demo project |
~
```

➤ Créez une branche pour la modification du fichier README.md du nom ‘updates’

```
Youcode@DESKTOP-R41R7JE MINGW64 ~/Desktop/projects/demo (master)
$ git branch updates
```

➤ Faites le staging et le commit en une seule ligne

```
Youcode@DESKTOP-R41R7JE MINGW64 ~/Desktop/projects/demo (updates)
$ git commit -am "modifier README"
[updates bebb273] modifier README
 1 file changed, 1 insertion(+), 1 deletion(-)

Youcode@DESKTOP-R41R7JE MINGW64 ~/Desktop/projects/demo (updates)
$ |
```

➤ Retournez vers la branche Master.

```
Youcode@DESKTOP-R41R7JE MINGW64 ~/Desktop/projects/demo (updates)
$ git checkout master
Switched to branch 'master'
```

➤ Affichez l'historique avec l'alias

```
Youcode@DESKTOP-R41R7JE MINGW64 ~/Desktop/projects/demo (master)
$ git historique
* bebb273 (updates) modifier README
* d87949a (HEAD -> master) Créez un fichier application.log et créez un fichier nommé « .gitignore »
* 2e3e816 Renommer le fichier Licence.md à Licence.txt
* 570ad08 créez le fichier Licence.md
* 0ef4822 première commit

Youcode@DESKTOP-R41R7JE MINGW64 ~/Desktop/projects/demo (master)
$ |
```

➤ Qu'est-ce que vous constatez

une fois retourné à un branche Master et tapé git historique nous trouvons HAED est devenue un branche Master

➤ Maintenant faites le merge.

```
Youcode@DESKTOP-R41R7JE MINGW64 ~/Desktop/projects/demo (master)
$ git merge updates
Updating d87949a..bebb273
Fast-forward
 README.md | 2 +-
 1 file changed, 1 insertion(+), 1 deletion(-)

Youcode@DESKTOP-R41R7JE MINGW64 ~/Desktop/projects/demo (master)
$ |
```

sixth Step : Conflict Resolution

➤ Créez une branche avec le nom ‘BAD’

```
Youcode@DESKTOP-R41R7JE MINGW64 ~/Desktop/projects/demo (master)
$ git branch BAD
```

➤ Modifiez le fichier README.md et ajouter la ligne ‘Trouble’

```
MINGW64:/c/Users/Youcode/Desktop/projects/demo
#Demo project
Trouble|
```

➤ Switchez faire la branche principale ‘master’

```
Youcode@DESKTOP-R41R7JE MINGW64 ~/Desktop/projects/demo (BAD)
$ git checkout master
Switched to branch 'master'

Youcode@DESKTOP-R41R7JE MINGW64 ~/Desktop/projects/demo (master)
$ |
```

- Maintenant modifiez le fichier README.md et ajoutez la ligne ‘Troubleshooting’
 - ⇒ Vim README.md
 - ⇒ Ajouter ligne Troubleshooting
- Staging/commiting avec commentaire ‘branche bien faite’

```
Youcode@DESKTOP-R41R7JE MINGW64 ~/Desktop/projects/demo (master)
$ git commit -am "branche bine faite"
[master c5167d2] branche bine faite
 1 file changed, 2 insertions(+), 1 deletion(-)

Youcode@DESKTOP-R41R7JE MINGW64 ~/Desktop/projects/demo (master)
$ |
```

- corriger le commentaire du dernier commit
 - ⇒ tapez la commande : git commit --amend

```
~/Desktop/projects/demo
commit bien| faite

# Please enter the commit message for your changes. Lines starting
# with '#' will be ignored, and an empty message aborts the commit.
#
# Date:      Tue Nov 26 16:42:10 2019 +0100
#
# On branch master
# Changes to be committed:
#       modified:   README.md
#
~
~
~
~
~
~
```

- Affichez l'historique

```
Youcode@DESKTOP-R41R7JE MINGW64 ~/Desktop/projects/demo (master)
$ git historique
* 254f34e (HEAD -> master) commit bien faite
| * f3595df (BAD) ajouter la ligne 'Trouble'
|/
* bebb273 (updates) modifier README
* d87949a Créez un fichier application.log et créez un fichier nommé « .gitignore »
* 2e3e816 Renommer le fichier Licence.md à Licence.txt
* 570ad08 créez le fichier Licence.md
* 0ef4822 première commit

Youcode@DESKTOP-R41R7JE MINGW64 ~/Desktop/projects/demo (master)
$ |
```

- Qu'est-ce que vous constatez ? expliquez l'objectif de ce petit scenario ?

comprendre comment modifier le message de validation en cas d'erreur de frappe ou si vous souhaitez simplement modifier le message de validation.

- Faites le merge de la branche ‘BAD’

```
Youcode@DESKTOP-R41R7JE MINGW64 ~/Desktop/projects/demo (master)
$ git merge BAD
Auto-merging README.md
CONFLICT (content): Merge conflict in README.md
Automatic merge failed; fix conflicts and then commit the result.

Youcode@DESKTOP-R41R7JE MINGW64 ~/Desktop/projects/demo (master|MERGING)
$ |
```

➤ Expliquez le message ?

conflit n'est pas résoudre car il existe deux modification au même fichier, il doit être discuter avec nos équipe d'accord à seul modification.

➤ Exécutez la commande suivante : cat README.md ? Expliquez ?

```
Youcode@DESKTOP-R41R7JE MINGW64 ~/Desktop/projects/demo (master|MERGING)
$ cat README.md
#Demo project
<<<<< HEAD
Troubleshooting
=====
Trouble
>>>>> BAD
```

⇒ La commande cat affichée le contenu de fichier.

➤ Tapez la commande « git mergetool » ? qu'est-ce que vous constatez ?

```
Youcode@DESKTOP-R41R7JE MINGW64 ~/Desktop/projects/demo (master|MERGING)
$ git mergetool

This message is displayed because 'merge.tool' is not configured.
See 'git mergetool --tool-help' or 'git help config' for more details.
'git mergetool' will now attempt to use one of the following tools:
opendiff kdiff3 tkdiff xxdiff meld tortoisemerge gvimdiff diffuse diffmerge ecmerge p4merge araxis bc codecompare smerge emerge vimdiff
Merging:
README.md

Normal merge conflict for 'README.md':
 {local}: modified file
 {remote}: modified file
Hit return to start merge resolution tool (vimdiff): |
```

⇒ Il faut installer un logiciel p4merge pour le GIT accepter la commande git mergetool.

seventh Step: merge tools

➤ tapez la commande : git config –global –list

➤ Git config –global merge.tool p4merge

```
Youcode@DESKTOP-R41R7JE MINGW64 ~/Desktop/projects/demo (master|MERGING)
$ git config --global --list
user.name=Salma Belassiri
user.email=salmabelassiri@gmail.com
alias.historique=log --oneline --graph --decorate --all

Youcode@DESKTOP-R41R7JE MINGW64 ~/Desktop/projects/demo (master|MERGING)
$ git config --global merge.tool p4merge
```

➤ Git config –global mergetool.p4merge.path “lien d’installation”

➤ N'oubliez pas de bien configurer le prompt ?

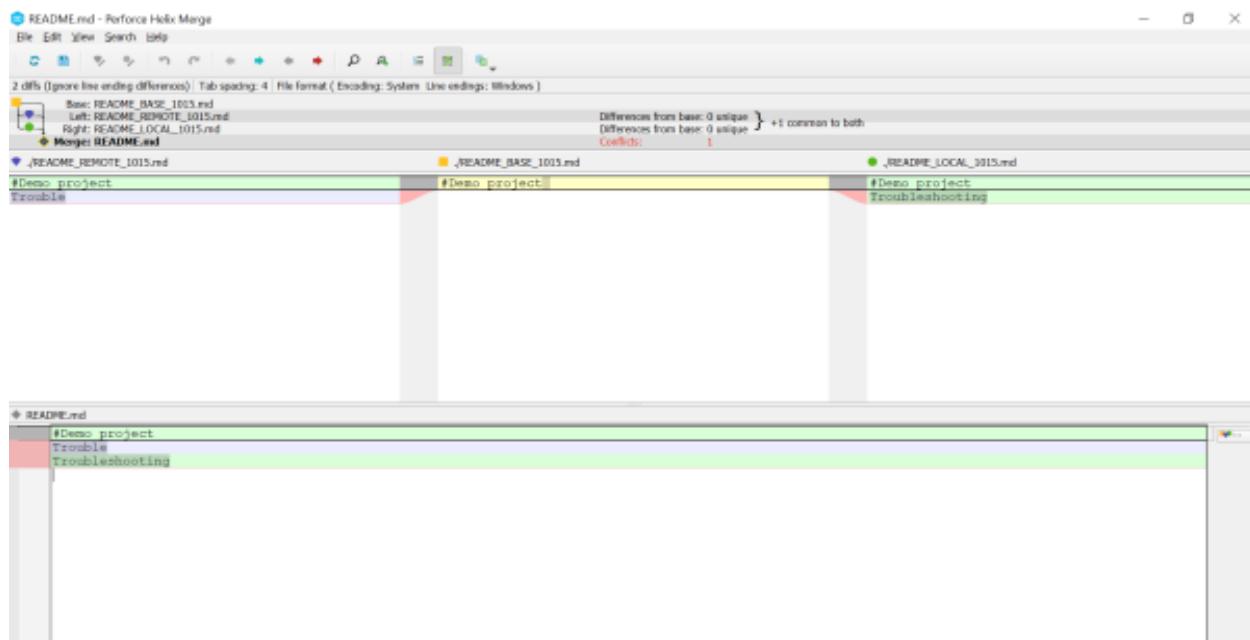
➤ Tapez la commande git mergetool

```
Youcode@DESKTOP-R41R7JE MINGW64 ~/Desktop/projects/demo (master|MERGING)
$ git config --global mergetool.prompt false
```

```
Youcode@DESKTOP-R41R7JE MINGW64 ~/Desktop/projects/demo (master|MERGING)
$ git mergetool
Merging:
 README.md
```

```
Normal merge conflict for 'README.md':
 {local}: modified file
 {remote}: modified file
```

```
README.md seems unchanged.
Was the merge successful [y/n]? README.md seems unchanged.
Was the merge successful [y/n]? README.md seems unchanged.
Was the merge successful [y/n]?
README.md seems unchanged.
Was the merge successful [y/n]?
README.md seems unchanged.
Was the merge successful [y/n]?
Youcode@DESKTOP-R41R7JE MINGW64 ~/Desktop/projects/demo (master|MERGING)
$ git historique
* 254f34e (HEAD -> master) commit bien faite
 | * f3595df (BAD) ajouter la ligne 'Trouble'
 |
* bebb273 (updates) modifier README
* d87949a Créez un fichier application.log et créez un fichier nommé « .gitignore »
* 2e3e816 Renommer le fichier Licence.md à Licence.txt
* 570ad08 créez le fichier Licence.md
* 0ef4822 première commit
```

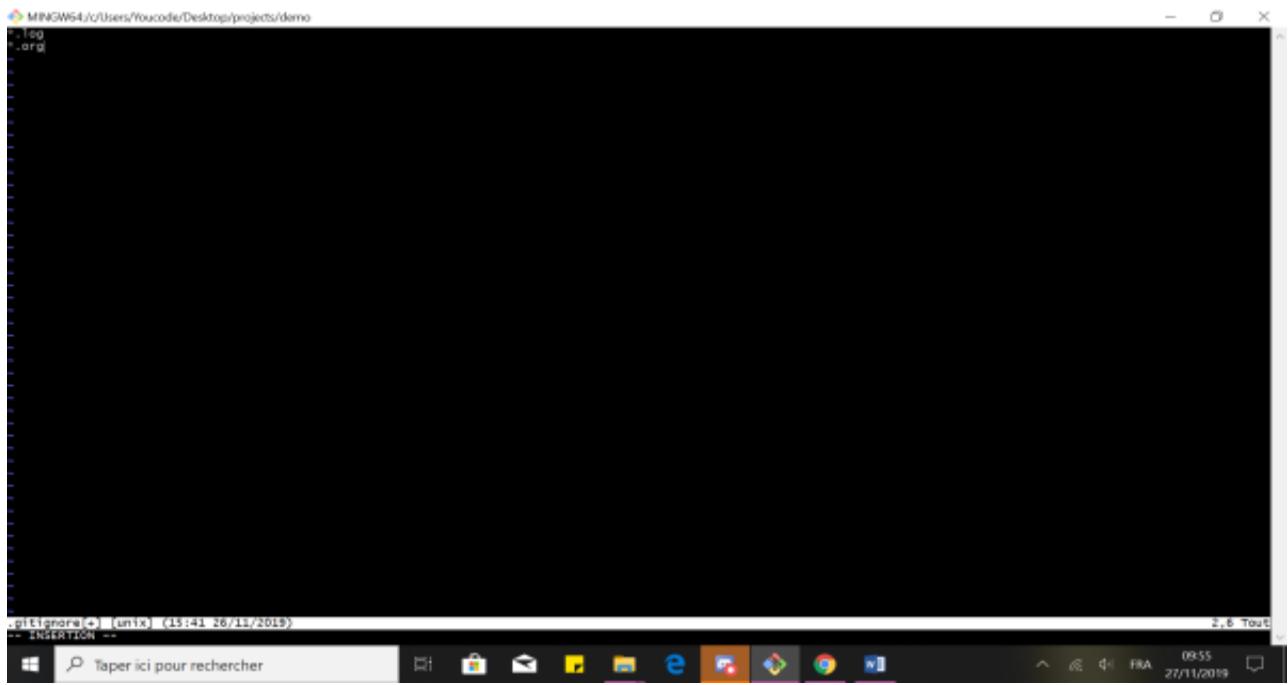


➤ Analysez ce que vous voyez ? et expliquez la plateforme ouverte en temps réel ?

Cet outil montre les deux différences qui ont été apportées au fichier et nous pouvons décider si nous souhaitons conserver une seule modification ou bien les deux.

eighth Step: Challenge

➤ Sur le fichier .gitignore ; écrivez une clause pour rejeter les fichiers indésirables et redondants



```
Youcode@DESKTOP-R41R7JE MINGW64 ~/Desktop/projects/demo (master)
$ git historique
* 030ad48 (HEAD -> master) conflict resolved
|\ 
* f3595df (BAD) ajouter la ligne 'Trouble'
* | 254f34e commit bien faite
|\ 
* bebb273 (updates) modifier README
* d87949a Créez un fichier application.log et créez un fichier nommé « .gitignore »
* 2e3e816 Renommer le fichier Licence.md à Licence.txt
* 570ad08 créez le fichier Licence.md
* 0ef4822 première commit
```

Scénario #2

First Step : Tagging

- Créez un TAG avec un nom V1.0 et un commentaire ‘ RELEASE 1.0 ‘
- Affichez les informations sur le TAG

```
Youcode@DESKTOP-R41R7JE MINGW64 ~/Desktop/projects/demo (master)
$ git tag -a v1.0 -m "RELEASE 1.0"

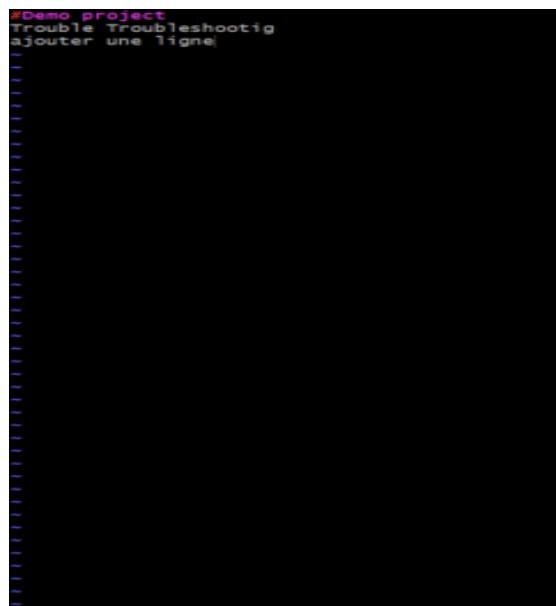
Youcode@DESKTOP-R41R7JE MINGW64 ~/Desktop/projects/demo (master)
$ git tag
v1.0
```

- D’après vous un TAG il sert à faire quoi au juste ?

Les tags sont un aspect simple de Git, ils vous permettent d’identifier des versions spécifiques de votre code. Vous pouvez considérer une étiquette comme une branche qui ne change pas. Une fois créé, il perd la possibilité de changer l’historique des commits.

Second Step : Stashing and Saving work in Progress

- Modifiez le fichier README.md , ajoutez une ligne
- ⇒ Tapez commande :
 - ✓ vim README.md



```
#Demo project
Trouble Troubleshooting
ajouter une ligne
```

- Tapez la commande git Stash ?
- Tapez la commande git stash list
- Exécutez la commande git status

```

$ stash
Youcode@DESKTOP-R41R7JE MINGW64 ~/Desktop/projects/demo (master)
$ git stash
warning: LF will be replaced by CRLF in .gitignore.
The file will have its original line endings in your working directory
Saved working directory and index state WIP on master: 030ad48 conflict resolved

Youcode@DESKTOP-R41R7JE MINGW64 ~/Desktop/projects/demo (master)
$ git stash
No local changes to save

Youcode@DESKTOP-R41R7JE MINGW64 ~/Desktop/projects/demo (master)
$ git stash list
stash@{0}: WIP on master: 030ad48 conflict resolved

Youcode@DESKTOP-R41R7JE MINGW64 ~/Desktop/projects/demo (master)
$ git status
On branch master
nothing to commit, working tree clean

```

➤ Expliquez le fonctionnement de la commande git Stash

Git stash est une commande de Git qui permet de "mettre de côté" des modifications. Cette fonctionnalité est très pratique, notamment lorsque le développeur a besoin de changer de branche rapidement. Cette fonctionnalité lui permet de ne pas se presser à effectuer un commit bâclé pour pouvoir changer de branche.

Git stash va récupérer les **modifications en cours** et les enregistrer dans un **conteneur** qu'on appellera un **stash**

➤ Modifiez le fichier « Licence.txt », ajoutez la ligne « APACHE 2.0 »



➤ Faites staging et le commit en une seule ligne

```

Youcode@DESKTOP-UAFPIDM MINGW64 ~/Desktop/projet/demo (master)
$ git commit -am "licence;txt mod"
[master 44251c1] licence;txt mod
 1 file changed, 1 insertion(+)

```

➤ Exécutez la commande « git stash pop » ?

```

$ git stash pop
On branch master
Changes not staged for commit:
  (use "git add <file>..." to update what will be committed)
  (use "git restore <file>..." to discard changes in working directory)
    modified: README.md

no changes added to commit (use "git add" and/or "git commit -a")
Dropped refs/stash@{0} (8c7e87c794814e6fe829e763bddafab9c00a8ca)

```

➤ Qu'est-ce que vous constatez ?

Git stash pop afficher les modification qui on a sauvegarder dans le stash .

third step: Voyage sur Github, Local Repo to github Repo

- Créez un repo github public sans ajouter le fichier README.md

Create a new repository

A repository contains all project files, including the revision history. Already have a project repository elsewhere? [Import a repository.](#)

Owner Repository name *

Great repository names are short and memorable. Need inspiration? How about [jubilant-guide](#)?

Description (optional)

 Public Anyone can see this repository. You choose who can commit.

 Private You choose who can see and commit to this repository.

Skip this step if you're importing an existing repository.

Initialize this repository with a README
This will let you immediately clone the repository to your computer.

- Créez le remote en https

```
yc@DESKTOP-ISOQRAB MINGW64 ~/Desktop/project/demo (master)
$ git remote add origin https://github.com/Salma-hu/pref-projet
```

- Pushez le tous à travers la commande : git push -u origin master - -tags

```
Youcode@DESKTOP-R41R7JE MINGW64 ~/Desktop/projects/demo (master)
$ git push -u origin master --tags
Enumerating objects: 32, done.
Counting objects: 100% (32/32), done.
Delta compression using up to 8 threads
Compressing objects: 100% (21/21), done.
Writing objects: 100% (32/32), 2.90 KiB | 165.00 KiB/s, done.
Total 32 (delta 7), reused 0 (delta 0)
remote: Resolving deltas: 100% (7/7), done.
To https://github.com/Salma-hu/pref-projet.git
 * [new branch]      master -> master
 * [new tag]          v1.0 -> v1.0
Branch 'master' set up to track remote branch 'master' from 'origin'.
Youcode@DESKTOP-R41R7JE MINGW64 ~/Desktop/projects/demo (master)
$ |
```

- Expliquez la commande ?

⇒ La commande est pour posher notre repo à github avec les tags .

- Sur Github Vérifiez la liste des commits

10 commits 1 branch 0 packages 1 release 1 contributor View license

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

Salma-hu Nouveau modification Latest commit 6dd4c14 2 hours ago

.gitignore	Nouveau modification	2 hours ago
Licence.txt	ajoutez APACHE 2.0 dans le fichier Licence.txt	2 hours ago
README.md	Nouveau modification	2 hours ago
README.orig	conflict resolved	4 hours ago
README_BACKUP_1015.md	conflict resolved	4 hours ago
README_BACKUP_2098.md	conflict resolved	4 hours ago
README_BASE_1015.md	conflict resolved	4 hours ago
README_BASE_2098.md	conflict resolved	4 hours ago
README_LOCAL_1015.md	conflict resolved	4 hours ago
README_LOCAL_2098.md	conflict resolved	4 hours ago
README_REMOTE_1015.md	conflict resolved	4 hours ago
README_REMOTE_2098.md	conflict resolved	4 hours ago

<https://github.com/Salma-hu/pref-projet/releases>

Fourth Step : Mini challenge (optionnel)

➤ Créez un dossier sous git nommé .SSH

⇒ Tapez la commande :mkdir .SSH

➤ Déplacez-vous dans le dossier .SSH

⇒ Déplacer par la commande : cd .SSH

➤ Créez une authentification sécurisé SSH entre votre repo local et votre repo distant (aide : Ssh –keygen-t rsa –C 'email').

```
Youcode@DESKTOP-R41R7JE MINGW64 ~/Desktop/projects/demo/.SSH (master)
$ ssh-keygen -t rsa -C "salmabellassiri@gmail.com"
Generating public/private rsa key pair.
Enter file in which to save the key (/c/Users/Youcode/.ssh/    ):
Enter passphrase (empty for no passphrase): |
Enter same passphrase again:
Your identification has been saved in.
Your public key has been saved in.pub.
The key fingerprint is:
SHA256:FOXmwCfTo22uryN1kBEbbxbx1m9b9nz+Dtcyj02Dokg salmabellassiri@gmail.com
The key's randomart image is:
----[RSA 3072]----+
| . . . . . |
| . . . . O |
| + . . . . * |
| .S . o .oB0 |
| o . o oOxO |
| E o . o .++* |
| o . +o. |
| .+++++ |
----[SHA256]----+
Youcode@DESKTOP-R41R7JE MINGW64 ~/Desktop/projects/demo/.SSH (master)
$ cat ~/.ssh/id_rsa.pub
ssh-rsa AAAQABAAQABAAA...QCM/1UpPjk8SjuaSB7Ur...By050GZd80JaLSgK+qJkE73p4aFEj1wU1l...r631Xed+39jdZtzb2C3vjSdfT4zE0TzBC4eLzV2NxKkwLVfVvPmoxVUhf...Y2VgOG...2jyYrj4+...F...P...D...G...X...i8QDf1s...BaZ...G...T...2...1...u...V...G...G...t...B...a...D...C...b...j...z...4...G...9...r...A...X...0...8...7...F...3...S...c...t...n...C...v...8...s...k...U...d...C...i...f...U...4...i...l...B...V...S...S...Q...i...h...I...v...P...w...g...B...8...8...4...a...M...b...0...4...3...z...S...Q...6...G...i...d...v...j...r...6...T...W...A...A...U...P...+...X...k...t.../...E...2...h...i...u.../...p...3...5...c...i...h...F...P...G.../...T...k...X...M...i...n...b...y...j...O...Z...w...k...z...4...q...d...k...z...3...B...f...q...5...7...k...v...y...k...n...j...x...u...v...f...f...h...i...v...F...d.../...Y...A...F...3...3...X...V...Z...k...R...S...m...k...g...o...k.../...w...f...g...h...F...w...7...X...3...0...8...3...s...D...L...k...x...h...k...P...L...S...3...4...k...f...H...h...V...z...v...b...l...u...s.../...A...x...z...h...f...w...c...i...f...t...a...8...4...M...q...k...C...4...P...o...+...L...v...5...d.../...i...g...v...1...c...C...S...e...w...Q...u...A...I...C...6...R...9...Q...R...a...l...t...6...u...3...o...i...E.../...X...C.../...Y...o...p...j...b...p...a...m...R...o...6...1...k...i...h...L...p...d...a...s...t...F...x...3...D...M...K...P...V...x...c...h...5...2...N...b...s...= salmabellassiri@gmail.com
Youcode@DESKTOP-R41R7JE MINGW64 ~/Desktop/projects/demo/.SSH (master)
$ |
```

The screenshot shows the GitHub 'SSH keys / Add new' page. On the left, there's a sidebar with options like Personal settings, Profile, Account, Security, Emails, Notifications, Billing, SSH and GPG keys, Blocked users, Repositories, Organizations, Saved replies, and Applications. The 'SSH and GPG keys' section is highlighted. The main area has a title 'SSH keys / Add new' and a 'Key' input field containing a long string of characters. Below the input field is a green 'Add SSH key' button.

Fifth Step: Crédation d'une local copy :

- Sur github créez un autre repo nommé (Monsiteweb)
- Ajoutez à l'arborescence toujours sur github le fichier .gitignore et un fichier licence.txt ‘APACHE 2.0 ‘

The screenshot shows the GitHub 'Create repository' form. It asks for the owner (Salma-hu) and repository name (Monsiteweb). It provides a suggestion for a great repository name: 'fuzzy-meme?'. There's a description field (optional) which is empty. The visibility is set to 'Public' (Anyone can see this repository). There's also a 'Private' option (You choose who can see and commit to this repository). Below these, there's a note about skipping if importing an existing repository. Under 'Initialize this repository with a README', there's a checked checkbox. It also offers to add a .gitignore file (Android) and a license file (Apache License 2.0). At the bottom is a green 'Create repository' button.

- Déplacez-vous dans le répertoire projets sous GIT
- Créez un clone github vers le local sous le nom (Monsiteweb-local)

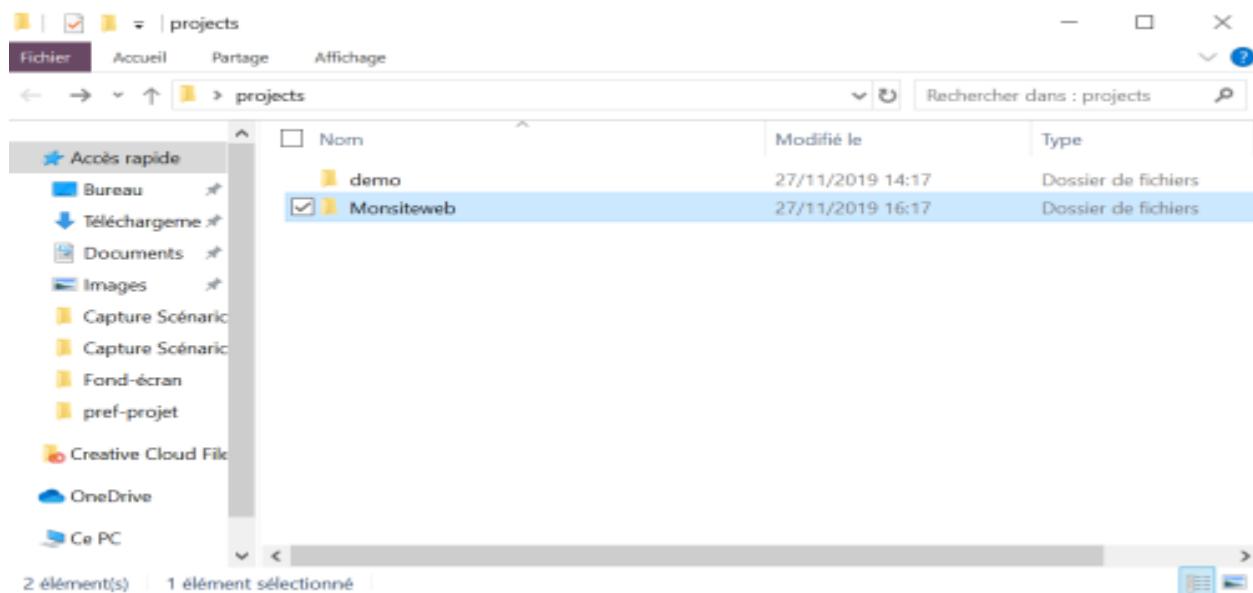
```
Youcode@DESKTOP-R41R7JE MINGW64 ~/Desktop/projects/demo/.SSH (master)
$ cd ..

Youcode@DESKTOP-R41R7JE MINGW64 ~/Desktop/projects/demo (master)
$ cd ..

Youcode@DESKTOP-R41R7JE MINGW64 ~/Desktop/projects
$ git clone https://github.com/Salma-hu/Monsiteweb.git
Cloning into 'Monsiteweb'...
remote: Enumerating objects: 8, done.
remote: Counting objects: 100% (8/8), done.
remote: Compressing objects: 100% (6/6), done.
remote: Total 8 (delta 1), reused 0 (delta 0), pack-reused 0
Unpacking objects: 100% (8/8), done.

Youcode@DESKTOP-R41R7JE MINGW64 ~/Desktop/projects
$ |
```

- Vérifiez si le clone est créé.



sixth step : *Sending the website*

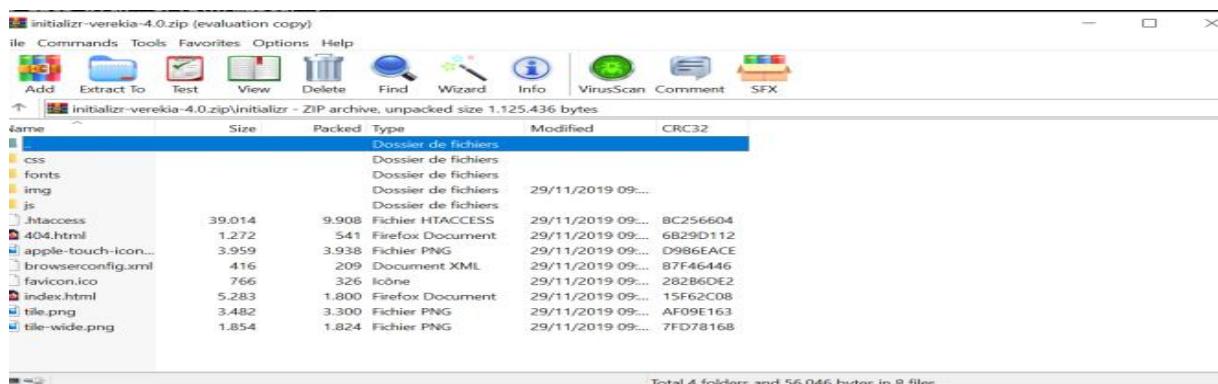
- Télécharger le site web depuis le lien suivant : <http://www.initializr.com/>
- Sur le site telecharger un site bootstrap avec le fichier .htaccess et le fichier 404.html

2 - Fine tuning

The screenshot shows the configuration interface for Initializr. It has four main sections:

- HTML/CSS Template:** Radio buttons for 'No template', 'Mobile-first Responsive', and 'Twitter Bootstrap'. 'Twitter Bootstrap' is selected.
- HTML5 Polyfills:** Radio buttons for 'Modernizr', 'Just HTML5shiv', and 'Respond - Alternatives'. 'Modernizr' is selected.
- jQuery:** Radio buttons for 'Minified' and 'Development'. 'Development' is selected.
- H5BP Optional:** Checkboxes for 'IE Classes', 'Old browser warning', 'Google Analytics', '.htaccess', 'Favicon', 'Apple and Windows Icons', 'plugins.js', 'Humans.txt', '404 Page', 'Adobe Cross Domain', and 'Robots.txt'. 'IE Classes', 'Favicon', '404 Page', and '.htaccess' are checked.

- Analysez l'arborescence.



Le site web contient dossier css,fonts,js,img.....

- Copiez le site télécharger dans votre repo local à travers une seule commande

```
Youcode@DESKTOP-UAFP1DM MINGW64 ~/Desktop/projet/monsiteweb-local
$ cp -R ../../initializr/* .
```

- Git status

```
Youcode@DESKTOP-UAFP1DM MINGW64 ~/Desktop/projet/monsiteweb-Tocal/monsiteweb (master)
$ git status
On branch master
Your branch is up to date with 'origin/master'.

Untracked files:
  (use "git add <file>..." to include in what will be committed)
    404.html
    apple-touch-icon.png
    browserconfig.xml
    css/
    favicon.ico
    fonts/
    index.html
    js/
    tile-wide.png
    tile.png

nothing added to commit but untracked files present (use "git add" to track)
```

- Faites le staging et le commit en une seule ligne

```
Youcode@DESKTOP-R41R7JE MINGW64 ~/Desktop/projects/Monsiteweb (master)
$ git add -A && git commit -am "ajouter initializr"
warning: LF will be replaced by CRLF in 404.html.
The file will have its original line endings in your working directory
warning: LF will be replaced by CRLF in css/bootstrap-theme.css.
The file will have its original line endings in your working directory
warning: LF will be replaced by CRLF in css/bootstrap-theme.min.css.
The file will have its original line endings in your working directory
```

- Faite le push à github

```
Youcode@DESKTOP-R41R7JE MINGW64 ~/Desktop/projects/Monsiteweb (master)
$ git push -u origin master
Enumerating objects: 26, done.
Counting objects: 100% (26/26), done.
Delta compression using up to 8 threads
Compressing objects: 100% (24/24), done.
Writing objects: 100% (25/25), 262.68 KiB | 5.71 MiB/s, done.
Total 25 (delta 3), reused 0 (delta 0)
remote: Resolving deltas: 100% (3/3), done.
To https://github.com/Salma-hu/Monsiteweb.git
  20a916a..83c5701 master -> master
Branch 'master' set up to track remote branch 'master' from 'origin'.
```



```
Youcode@DESKTOP-R41R7JE MINGW64 ~/Desktop/projects/Monsiteweb (master)
$ |
```

- Vérifiez l'existence du site local sur votre repo github

No description, website, or topics provided.

Manage topics

Edit

3 commits	1 branch	0 packages	0 releases	1 contributor	Apache-2.0
Branch: master ▾	New pull request	Create new file	Upload files	Find file	Clone or download ▾
Salma-hu ajouter initializr			Latest commit 83c5701 6 minutes ago		
 css	ajouter initializr				6 minutes ago
 fonts	ajouter initializr				6 minutes ago
 js	ajouter initializr				6 minutes ago
 .gitignore	Initial commit				1 hour ago
 404.html	ajouter initializr				6 minutes ago
 LICENSE	Initial commit				1 hour ago
 README.md	Update README.md				42 minutes ago
 index.html	ajouter initializr				6 minutes ago
 README.md					

seventh step : Fetch and pull

- Sur github éditez le fichier Index.html, sur la balise <title> </title> ajoutez le titre , mon premier site web .

- ## ➤ Faites le commit sur github

Salma-hu Update index.html

- Sur git et sur le repo local , Editez le fichier README.md

```
# Monsiteweb-local
premier site web
~
```

➤ Faites le staging et le commit en une seule ligne

```
Youcode@DESKTOP-R41R7JE MINGW64 ~/Desktop/projects/Monsiteweb (master)
$ git commit -am "modifier repo local"
[master 5045031] modifier repo local
 1 file changed, 1 insertion(+)

Youcode@DESKTOP-R41R7JE MINGW64 ~/Desktop/projects/Monsiteweb (master)
$ git status
On branch master
Your branch is ahead of 'origin/master' by 1 commit.
  (use "git push" to publish your local commits)

nothing to commit, working tree clean

Youcode@DESKTOP-R41R7JE MINGW64 ~/Desktop/projects/Monsiteweb (master)
$ |
```

- Executez la commande « git fetch »
- Git status
- Git pull
- Git push

```
Youcode@DESKTOP-R41R7JE MINGW64 ~/Desktop/projects/Monsiteweb (master)
$ git fetch
remote: Enumerating objects: 5, done.
remote: Counting objects: 100% (5/5), done.
remote: Compressing objects: 100% (3/3), done.
remote: Total 3 (delta 2), reused 0 (delta 0), pack-reused 0
Unpacking objects: 100% (3/3), done.
From https://github.com/Salma-hu/Monsiteweb
  83c5701..01546d2 master      -> origin/master

Youcode@DESKTOP-R41R7JE MINGW64 ~/Desktop/projects/Monsiteweb (master)
$ git status
On branch master
Your branch and 'origin/master' have diverged,
and have 1 and 1 different commits each, respectively.
  (use "git pull" to merge the remote branch into yours)

nothing to commit, working tree clean

Youcode@DESKTOP-R41R7JE MINGW64 ~/Desktop/projects/Monsiteweb (master)
$ git pull
Merge made by the 'recursive' strategy.
 index.html | 2 +-
 1 file changed, 1 insertion(+), 1 deletion(-)

Youcode@DESKTOP-R41R7JE MINGW64 ~/Desktop/projects/Monsiteweb (master)
$ git push
Enumerating objects: 9, done.
Counting objects: 100% (8/8), done.
Delta compression using up to 8 threads
Compressing objects: 100% (4/4), done.
Writing objects: 100% (5/5), 582 bytes | 145.00 KiB/s, done.
Total 5 (delta 2), reused 0 (delta 0)
remote: Resolving deltas: 100% (2/2), completed with 2 local objects.
To https://github.com/Salma-hu/Monsiteweb.git
  01546d2..c8ba7bf master -> master

Youcode@DESKTOP-R41R7JE MINGW64 ~/Desktop/projects/Monsiteweb (master)
$ |
```

➤ Verifiez les commit sur github

The screenshot shows the GitHub interface for the repository 'Monsiteweb'. The 'Branch: master' dropdown is set to 'master'. The commit history is as follows:

- Commits on Nov 28, 2019:**
 - Merge branch 'master' of https://github.com/Salma-hu/Monsiteweb by Salma-hu committed 3 minutes ago. (Commit hash: c8ba7bf)
 - modifier repo local by Salma-hu committed 12 minutes ago. (Commit hash: 5045031)
 - Update index.html by Salma-hu committed 39 minutes ago. (Commit hash: 01546d2) - Verified
- Commits on Nov 27, 2019:**
 - ajouter initializr by Salma-hu committed 17 hours ago. (Commit hash: 83c5701)
 - Update README.md by Salma-hu committed 18 hours ago. (Commit hash: 20a916a) - Verified
 - Initial commit by Salma-hu committed 18 hours ago. (Commit hash: 13d27af) - Verified

Scénario #3

First Step: Changes on GitHub

- Sur le dossier CSS ajoutez un autre fichier nommé style.css

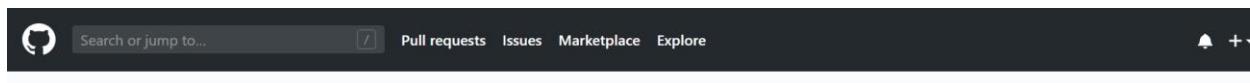
The screenshot shows a GitHub repository page for 'Salma-hu / Monsiteweb'. The repository has 9 commits, 1 branch, 0 packages, 0 releases, 1 contributor, and Apache-2.0 license. The commit history lists several commits, including one for 'style.css' and another for 'LICENSE'. A dropdown menu is open, showing options like 'Switch branches/tags' and 'Create branch: Example from 'master''. The commit for 'style.css' is highlighted.

- Sur index.html ajoutez juste après la balise <body> une balise <H1> : <h1> modification récente </h1>

The screenshot shows a code editor with the file 'index.html' open. The code includes meta tags, links to CSS files, and a script tag. A yellow box highlights the addition of a new

tag with the text 'modification récente' just after the opening tag.

- Renommer le fichier 404.html en error404.html



```
<!DOCTYPE html>
<html lang="en">
<head>
    <meta charset="utf-8">
    <title>Page Not Found</title>
    <meta name="viewport" content="width=device-width, initial-scale=1">
    <style>
        * {
            line-height: 1.2;
            margin: 0;
        }
        html {
            height: 100%;
```

➤ Sur gitHub vérifiez la liste des commits .

Branch: master ▾

- Commits on Nov 28, 2019
 - Rename 404.html to error404.html
Salma-hu committed 1 minute ago
 - Update index.html
Salma-hu committed 4 minutes ago
 - Create style.css
Salma-hu committed 39 minutes ago
 - Merge branch 'master' of https://github.com/Salma-hu/Monsiteweb
Salma-hu committed 1 hour ago
 - modifier repo local
Salma-hu committed 1 hour ago
 - Update index.html
Salma-hu committed 2 hours ago

➤ Sur GITBASH, vérifiez avec « git status »

```
$ git status
On branch master
Your branch is up to date with 'origin/master'.

nothing to commit, working tree clean
```

➤ Visualisez et télécharger les fichiers distants sur GITBASH

```
Youcode@DESKTOP-R41R7JE MINGW64 ~/Desktop/projects/Monsiteweb (master)
$ git fetch
remote: Enumerating objects: 11, done.
remote: Counting objects: 100% (11/11), done.
remote: Compressing objects: 100% (8/8), done.
remote: Total 8 (delta 5), reused 0 (delta 0), pack-reused 0
Unpacking objects: 100% (8/8), done.
From https://github.com/Salma-hu/Monsiteweb
  c8ba7bf..5e1c081      master      -> origin/master
```

- Maintenant faites le pull et fusionnez les changements distants avec le repo local
- ⇒ Taper la commande :git merge
- Vérifiez votre repo Local
- ⇒ Taper la commande :git historique
- Sur votre repo Distant GITHUB, laissez un commentaire sur les fichiers que vous avez modifiez en regardant la liste des commits

Rename 404.html to error404.html

master

Salma-hu committed 1 hour ago Verified

1 parent 181caca commit 5e1c08140be6392c0ad27cd0a3b5d7cdec7c9722

Showing 1 changed file with 0 additions and 0 deletions.

Unified Split

File renamed without changes.

0 comments on commit 5e1c081

bien Renommer!

Comment on this commit

Second Step: Branching and merging sur GITHUB

- Repo distant Sur GitHUB créez une branche du nom ‘Example’

Salma-hu / Monsiteweb

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.

Manage topics

9 commits 1 branch 0 packages 0 releases 1 contributor Apache-2.0

Branch: master New pull request

Switch branches/tags Example

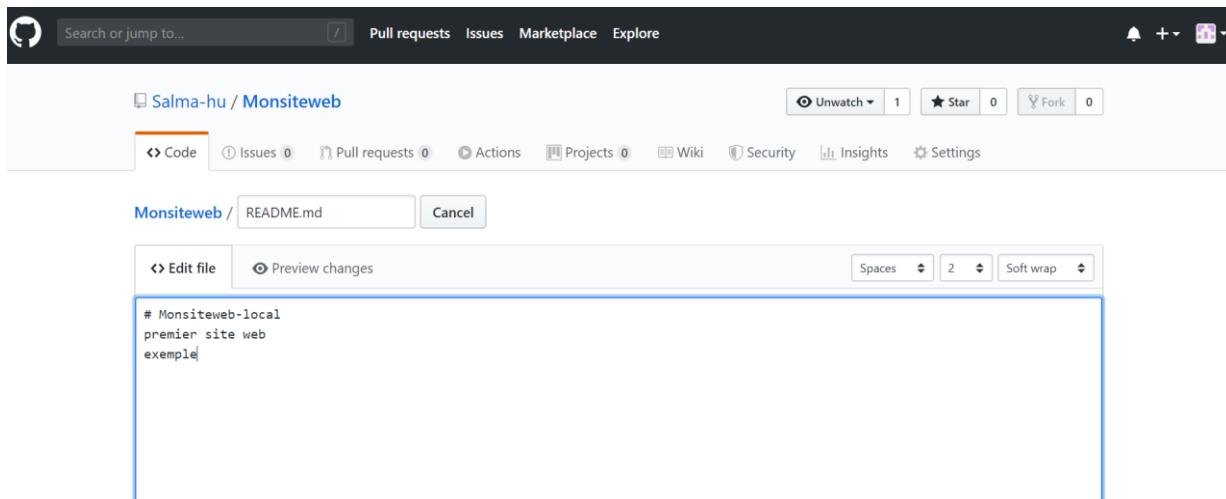
Create branch: Example from 'master'

LICENSE README.md

Latest commit 5e1c081 23 hours ago

style.css	23 hours ago	
initializr	2 days ago	
initializr	2 days ago	
al commit	2 days ago	
Initial commit	2 days ago	
modifier repo local	yesterday	

- Sur la branche ‘Example’, modifiez le fichier README.md et faites-le commit (toujours sur GITHUB)



- Repo local : Sur votre repo local, créez une branche nommé ‘annulation’ .

```
Youcode@DESKTOP-R41R7JE MINGW64 ~/Desktop/projects/Monsiteweb (master)
$ git checkout -b annulation
Switched to a new branch 'annulation'

Youcode@DESKTOP-R41R7JE MINGW64 ~/Desktop/projects/Monsiteweb (annulation)
$ vim index.html

Youcode@DESKTOP-R41R7JE MINGW64 ~/Desktop/projects/Monsiteweb (annulation)
$ |
```

- Sur votre repo local éditez le fichier « index.html » et supprimez la balise H1

```
<!DOCTYPE html>
<!--[if lt IE 7]>      <html class="no-js lt-ie9 lt-ie8 lt-ie7" lang=""> <![endif]-->
<!--[if IE 7]>      <html class="no-js lt-ie9 lt-ie8" lang=""> <![endif]-->
<!--[if IE 8]>      <html class="no-js lt-ie9" lang=""> <![endif]-->
<!--[if gt IE 8]><!--> <html class="no-js" lang=""> <!--<![endif]-->
<head>
    <meta charset="utf-8">
    <meta http-equiv="X-UA-Compatible" content="IE=edge,chrome=1">
    <title> mon premier site web </title>
    <meta name="description" content="">
    <meta name="viewport" content="width=device-width, initial-scale=1">
    <link rel="stylesheet" href="css/bootstrap.min.css">
    <style>
        body {
            padding-top: 50px;
            padding-bottom: 20px;
        }
    </style>
    <link rel="stylesheet" href="css/bootstrap-theme.min.css">
    <link rel="stylesheet" href="css/main.css">
    <script src="js/vendor/modernizr-2.8.3.min.js"></script>
</head>
<body>
    <nav class="navbar navbar-inverse navbar-fixed-top" role="navigation">
        <div class="container">
            <div class="navbar-header">
                <button type="button" class="navbar-toggle collapsed" data-toggle="collapse" data-target="#navbar" aria-expanded="false" aria-controls="navbar">
                    <span class="icon-bar"></span>
                    <span class="icon-bar"></span>
                    <span class="icon-bar"></span>
                </button>
                <a class="navbar-brand" href="#">Project name</a>
            </div>
            <div id="navbar" class="navbar-collapse collapse">
                <form class="navbar-form navbar-right" role="form">
                    <div class="form-group">
                        <input type="text" placeholder="Email" class="form-control">
                    </div>
                    <div class="form-group">
                        <input type="password" placeholder="Password" class="form-control">
                    </div>
                    <button type="submit" class="btn btn-success">Sign in</button>
                </form>
            </div>
        </div>
    </nav>
```

- Faites-le commit et le staging en une seule ligne.

```
Youcode@DESKTOP-R41R7JE MINGW64 ~/Desktop/projects/Monsiteweb (annulation)
$ git commit -am "supprimez la balise H1"
[annulation b193b43] supprimez la balise H1
 1 file changed, 1 insertion(+), 1 deletion(-)
```

➤ « Git show » pour vérifier vos changements

```
Youcode@DESKTOP-R41R7JE MINGW64 ~/Desktop/projects/Monsiteweb (annulation)
$ git show
commit b193b4335abc9c73a80ee283b185007ec103bf48 (HEAD -> annulation)
Author: Salma Belassiri <salmabelassiri@gmail.com>
Date:   Fri Nov 29 10:17:06 2019 +0100

    supprimez la balise H1

diff --git a/index.html b/index.html
index e306481..5575920 100644
--- a/index.html
+++ b/index.html
@@ -23,7 +23,7 @@
     <script src="js/vendor/modernizr-2.8.3.min.js"></script>
 </head>
 <body>
-    <h1>modification récente</h1>
+    <nav class="navbar navbar-inverse navbar-fixed-top" role="navigation">
        <div class="container">
            <div class="navbar-header">
```

➤ Sur GITBASH Pushez les changements que vous avez sur la branche ‘annulation’ à votre repo distant

```
Youcode@DESKTOP-R41R7JE MINGW64 ~/Desktop/projects/Monsiteweb (annulation)
$ git push -u origin annulation
Enumerating objects: 9, done.
Counting objects: 100% (9/9), done.
Delta compression using up to 8 threads
Compressing objects: 100% (5/5), done.
Writing objects: 100% (6/6), 588 bytes | 588.00 KiB/s, done.
Total 6 (delta 3), reused 0 (delta 0)
remote: Resolving deltas: 100% (3/3), completed with 2 local objects.
remote:
remote: Create a pull request for 'annulation' on GitHub by visiting:
remote:     https://github.com/Salma-hu/Monsiteweb/pull/new/annulation
remote:
To https://github.com/Salma-hu/Monsiteweb.git
 * [new branch]      annulation -> annulation
Branch 'annulation' set up to track remote branch 'annulation' from 'origin'.
Youcode@DESKTOP-R41R7JE MINGW64 ~/Desktop/projects/Monsiteweb (annulation)
$ |
```

➤ Maintenant vérifiez votre repo distant.

Your recently pushed branches:

Example (28 minutes ago)	Compare & pull request
annulation (2 minutes ago)	Compare & pull request

Branch: Example ▾ [New pull request](#) [Create new file](#) [Upload files](#) [Find file](#) [Clone or download ▾](#)

This branch is 1 commit ahead of master.

	Salma-hu Update README.md	Latest commit e0f4ac2 29 minutes ago
	Create style.css	yesterday
	ajouter initializr	2 days ago
	ajouter initializr	2 days ago
	Initial commit	2 days ago
	Initial commit	2 days ago
	Update README.md	29 minutes ago
	Rename 404.html to error404.html	23 hours ago
	Update index.html	23 hours ago

Third Step: compare pull Requests

- Sur votre repo Distant, un message indiquant qu'une branche a été ajoutée avec un bouton « compare and pull request »
 - Cliquez sur le bouton ? analysez le rendu de la page ? qu'est-ce que vous constatez ?
 - Créez un pull request avec le commentaire ‘annulation à vérifier’

The screenshot shows the GitHub interface for creating a pull request. At the top, there are navigation links: Code, Issues (0), Pull requests (0), Actions, Projects (0), Wiki, Security, Insights, and Settings. The main title is "Open a pull request". Below it, a sub-instruction says "Create a new pull request by comparing changes across two branches. If you need to, you can also compare across forks." A dropdown menu shows "base: master" and "compare: annulation", with a green checkmark indicating "Able to merge. These branches can be automatically merged." The main form area has tabs "Write" (selected) and "Preview", and includes rich text editing tools (bold, italic, code blocks, etc.). The "Write" tab contains the message "annulation à vérifier!". Below the message area is a dashed blue box for attaching files. To the right of the message area are several configuration sections: "Reviewers" (No reviews), "Assignees" (No one—assign yourself), "Labels" (None yet), "Projects" (None yet), and "Milestone" (No milestone). Each section has a gear icon for settings.

- ### ➤ Une page s'ouvre, explorez la page ?

Your recently pushed branches:

Example (37 minutes ago)		Compare & pull request
Branch: master	New pull request	Create new file Upload files Find file Clone or download
Salma-hu Rename 404.html to error404.html		Latest commit 5e1c081 23 hours ago
css	Create style.css	yesterday
fonts	ajouter initializr	2 days ago
js	ajouter initializr	2 days ago
.gitignore	Initial commit	2 days ago
LICENSE	Initial commit	2 days ago
README.md	modifier repo local	yesterday
error404.html	Rename 404.html to error404.html	23 hours ago
index.html	Update index.html	23 hours ago

- Sur l'onglet ‘conversation’ vous pouvez aussi laisser des commentaires à la personne qui demande le pull request ; laissez un commentaire ?

Merging can be performed automatically.

Merge pull request You also can open this in GitHub Desktop or view command line instructions.

Write **Preview**

vérifier

Attach files by dragging & dropping, selecting or pasting them.

Close and comment **Comment**

Unsubscribe
You're receiving notifications because you're watching this repository.

1 participant

Lock conversation

💡 ProTip! Add comments to specific lines under [Files changed](#).

- Maintenant si tout va bien ; aucun message d'erreur effectuez le merging ?

Merged Annulation #1
Salma-hu merged 2 commits into `master` from `annulation` now

Salma-hu merged commit `f3380cc` into `master` now Revert

 Pull request successfully merged and closed Delete branch
You're all set—the `annulation` branch can be safely deleted.

 @Salma-hu Write Preview AA B i “ <> ↻

Leave a comment

Attach files by dragging & dropping, selecting or pasting them. M+

Comment

 **ProTip!** Add comments to specific lines under [Files changed](#).

➤ Supprimez la branche ‘annulation’ une fois tout est OK.

- ⇒ Taper la commande : git push origin : annulation
 - ⇒ Taper la commande : git branch -D annulation

fourth Step: merging en local

- Pour finaliser l'objectif de cette étape, on va créer une branche nommée ‘ajustement’, ou on va modifier le fichier ‘main.css’ (css/main.css) en ajoutant la ligne suivante : body { Width : 100% ; Height : 100% }

```
Youcode@DESKTOP-R41R7JE MINGW64 ~/Desktop/projects/Monsiteweb (annulation)
$ git checkout -b "ajustement"
Switched to a new branch 'ajustement'
```

```
Youcode@DESKTOP-R41R7JE MINGW64 ~/Desktop/projects/Monsiteweb (ajustement)
$ |
```

```
# main.css ×
css > # main.css > ...
1
2
3 /* =====
4 Author's custom styles
5 ===== */
6 body { Width : 100% ; Height : 100% }
7
8
9
10
11
```

- Faites maintenant le nécessaire pour ‘pushez’ les modifications sur GITHUB.

```
Youcode@DESKTOP-R41R7JE MINGW64 ~/Desktop/projects/Monsiteweb (ajustement)
$ git push -u origin ajustement
Enumerating objects: 59, done.
Counting objects: 100% (59/59), done.
Delta compression using up to 8 threads
Compressing objects: 100% (54/54), done.
Writing objects: 100% (59/59), 272.02 KiB | 3.09 MiB/s, done.
Total 59 (delta 20), reused 0 (delta 0)
remote: Resolving deltas: 100% (20/20), done.
remote:
remote: Create a pull request for 'ajustement' on GitHub by visiting:
remote:   https://github.com/Salma-hu/Monsiteweb/pull/new/ajustement
remote:
To https://github.com/Salma-hu/Monsiteweb.git
 * [new branch]      ajustement -> ajustement
Branch 'ajustement' set up to track remote branch 'ajustement' from 'origin'.

Youcode@DESKTOP-R41R7JE MINGW64 ~/Desktop/projects/Monsiteweb (ajustement)
$ |
```

- Sur GITBASH, Vérifiez dans quelle branche vous êtes maintenant !
- surement sur la branche ‘ajustement’

```
Your branch is up to date with 'origin/ajustement'.
Youcode@DESKTOP-R41R7JE MINGW64 ~/Desktop/projects/Monsiteweb (ajustement)
$ git branch
* ajustement
  annulation
  master

Youcode@DESKTOP-R41R7JE MINGW64 ~/Desktop/projects/Monsiteweb (ajustement)
$ |
```

- Sur votre GITHUB, un pull request est demandé !

Manage topics

12 commits	3 branches	0 packages	0 releases	1 contributor	Apache-2.0
------------	------------	------------	------------	---------------	------------

Your recently pushed branches:

ajustement (3 minutes ago)	
Branch: master ▾ New pull request	
Create new file Upload files Find file Clone or download ▾	
Salma-hu Merge pull request #1 from Salma-hu/annulation ... Latest commit f3380cc 28 minutes ago	
style.css Create style.css yesterday	
fonts ajouter initializr 2 days ago	
js ajouter initializr 2 days ago	
.gitignore Initial commit 2 days ago	
LICENSE Initial commit 2 days ago	
README.md new commit yesterday	
error404.html Rename 404.html to error404.html yesterday	
index.html supprimez la balise H1 1 hour ago	

- Un pull Request est demandé sur GITHUB ! N’oubliez pas !

The screenshot shows a GitHub interface with a search bar containing 'is:pr is:open'. A modal at the top right says 'Label issues and pull requests for new contributors' with a 'Dismiss' button. Below the search bar, there's a filter section with '1 Open' and '1 Closed'. A single pull request is listed: '#2 opened 20 minutes ago by Salma-hu'.

💡 ProTip! Type `g` `i` on any issue or pull request to go back to the issue listing page.

➤ Faites le merge à travers GITBASH

```
Youcode@DESKTOP-R41R7JE MINGW64 ~/Desktop/projects/Monsiteweb (master)
$ git merge ajustement
Updating 3d1f221..adec765
Fast-forward
  css/main.css | 3 +-+
  index.html    | 2 +-
  2 files changed, 3 insertions(+), 2 deletions(-)
```

➤ Git status

```
Youcode@DESKTOP-R41R7JE MINGW64 ~/Desktop/projects/Monsiteweb (master)
$ git status
On branch master
Your branch is ahead of 'origin/master' by 2 commits.
  (use "git push" to publish your local commits)

nothing to commit, working tree clean

Youcode@DESKTOP-R41R7JE MINGW64 ~/Desktop/projects/Monsiteweb (master)
$ |
```

➤ Si tout est OK ! Pushez les changements ?

```
Youcode@DESKTOP-R41R7JE MINGW64 ~/Desktop/projects/Monsiteweb (master)
$ git push origin master
Enumerating objects: 1, done.
Counting objects: 100% (1/1), done.
Writing objects: 100% (1/1), 256 bytes | 128.00 KiB/s, done.
Total 1 (delta 0), reused 0 (delta 0)
To https://github.com/Salma-hu/Monsiteweb.git
  f3380cc..94589af  master -> master

Youcode@DESKTOP-R41R7JE MINGW64 ~/Desktop/projects/Monsiteweb (master)
$ |
```

➤ Sur GITHUB supprimez les branches que vous avez créé.

The screenshot shows a GitHub repository page for 'Salma-hu / Monsiteweb'. At the top, there are navigation links: Code, Issues (0), Pull requests (0), Actions, Projects (0), Wiki, Security, Insights, and Settings. Below the navigation, there's a tabs section with 'Overview' (selected), Yours, Active, Stale, and All branches. A search bar for branches is also present. The main content area is divided into sections: 'Default branch' (master), 'Your branches' (listing 'ajustement' and 'Example' with 'Deleted just now by Salma-hu' status and 'Restore' buttons), and 'Active branches' (listing 'ajustement' and 'Example' with 'Deleted just now by Salma-hu' status and 'Restore' buttons).

➤ Tapez la commande git branch -a

```
Youcode@DESKTOP-R41R7JE MINGW64 ~/Desktop/projects/Monsiteweb (master)
$ git branch -a
ajustement
annulation
* master
remotes/origin/Example
remotes/origin/HEAD -> origin/master
remotes/origin/ajustement
remotes/origin/annulation
remotes/origin/master
```

```
Youcode@DESKTOP-R41R7JE MINGW64 ~/Desktop/projects/Monsiteweb (master)
$
```

➤ Sur GITBASH supprimez les branches que vous avez créé.

⇒ Taper la commande : git branch -d ajustement

➤ Maintenant ! Tapez la commande git branch -a ? qu'est-ce que vous constatez ?

⇒ La branche ajustement est supprimée.

fifth Step: The Cleaning up

➤ Sur GITHUB Créez une branche nommée 'updatelicence, éditez le fichier Licence et changer Apache 2.0 par Apache 3.0

Salma-hu / Monsiteweb

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

Overview Yours Active Stale All branches Search branches...

Default branch

master Updated 3 days ago by Salma-hu Default Change default branch

Your branches

updatelicens Updated 3 days ago by Salma-hu 0|0 New pull request ⚡

Active branches

updatelicens Updated 3 days ago by Salma-hu 0|0 New pull request ⚡

Monsiteweb / LICENSE Cancel Choose a license template

Edit file Preview changes Spaces 2 Soft wrap

Apache License
Version 3.0, January 2004
<http://www.apache.org/licenses/LICENSE-3.0>

TERMS AND CONDITIONS FOR USE, REPRODUCTION, DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

➤ Sur GITBASH faites un pull global : git pull --all

```
yc@DESKTOP-ISOQRAB MINGW64 ~/Desktop/projects/Minsiteweb (master)
$ git pull --all
Fetching origin
remote: Enumerating objects: 5, done.
remote: Counting objects: 100% (5/5), done.
remote: Compressing objects: 100% (3/3), done.
remote: Total 3 (delta 2), reused 0 (delta 0), pack-reused 0
Unpacking objects: 100% (3/3), done.
From https://github.com/fatimaezzahrazahid1997/Minsiteweb
 * [new branch]      updatelicens -> origin/updatelicens
Already up to date.
```

➤ Maintenant faites la merge de la branche 'updatelicens'

```
yc@DESKTOP-ISOQRAB MINGW64 ~/Desktop/projects/Minsiteweb (master)
$ git merge origin/updatelicence
Updating 1417b99..1ab08d4
Fast-forward
 LICENSE | 2 +-
 1 file changed, 1 insertion(+), 1 deletion(-)
```

- Executez Git push !

```
Youcode@DESKTOP-R41R7JE MINGW64 ~/Desktop/projects/Monsiteweb (master)
$ git push
fatal: unable to access 'https://github.com/Salma-hu/Monsiteweb.git/': Could not resolve host: github.com
```

- Git branch -d ‘updatelicence’
- Exécuter la commande git bash -a ? qu'est-ce que vous constatez sur le remote et aussi sur GTHUB?

```
Youcode@DESKTOP-R41R7JE MINGW64 ~/Desktop/projects/Monsiteweb (master)
$ git branch -a
* master
  remotes/origin/Example
  remotes/origin/HEAD -> origin/master
  remotes/origin/ajustement
  remotes/origin/annulation
  remotes/origin/master
  remotes/origin/updatelicenc
```

- Pourriez-vous expliquer l'objectif de cette phase ?

Le but de cette étape est de comprendre comment mieux manipuler les branches et surtout comment supprimer une branche distante du terminal Gitbash

sixth step: Rebasing:

- Faites une veille sur REBASE de 30 min intergroupe , visitez le lien suivant
- <https://www.grafikart.fr/tutoriels/amend-rebase-588>
- Sur GITHUB, Editez le fichier README.md, ajoutez la ligne ‘updates’ ! n’oubliez pas le commit



Salma-hu / Monsiteweb

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

Monsiteweb / README.md Cancel

Monsiteweb-local

➤ Sur GITBASH Editez le fichier index.html supprimez les lignes suivantes

```
index.html ×
index.html > html.no-js
1 <!doctype html>
2 <!--[if lt IE 7]>      <html class="no-js lt-ie9 lt-ie8 lt-ie7" lang=""> <![endif]-->
3 <!--[if IE 7]>        <html class="no-js lt-ie9 lt-ie8" lang=""> <![endif]-->
4 <!--[if IE 8]>        <html class="no-js lt-ie9" lang=""> <![endif]-->
5 <!--[if gt IE 8]><!-- <html class="no-js" lang=""> <!--<![endif]-->
6   <head>
7     <meta charset="utf-8">
8     <meta http-equiv="X-UA-Compatible" content="IE=edge,chrome=1">
```

➤ Et modifiez le titre du site : choisissez le titre que vous voulez

```
<head>
  <meta charset="utf-8">
  <meta http-equiv="X-UA-Compatible" content="IE=edge,chrome=1">
  <title> mon site </title>
  <meta name="description" content="">
  <meta name="viewport" content="width=device-width, initial-scale=1">
```

➤ Faites le staging et le commit en une seule ligne

```
Youcode@DESKTOP-R41R7JE MINGW64 ~/Desktop/projects/Monsiteweb (master)
$ git commit -am "Editez le fichier index.html"
[master d94eeae] Editez le fichier index.html
 1 file changed, 2 insertions(+), 5 deletions(-)

Youcode@DESKTOP-R41R7JE MINGW64 ~/Desktop/projects/Monsiteweb (master)
$
```

➤ Maintenant apportez les changements faites sur le Repo distant : fetch

```
Youcode@DESKTOP-R41R7JE MINGW64 ~/Desktop/projects/Monsiteweb (master)
$ git fetch
remote: Enumerating objects: 5, done.
remote: Counting objects: 100% (5/5), done.
remote: Compressing objects: 100% (2/2), done.
remote: Total 3 (delta 1), reused 0 (delta 0), pack-reused 0
Unpacking objects: 100% (3/3), done.
From https://github.com/Salma-hu/Monsiteweb
  c294e19..7cccb42 master      -> origin/master

Youcode@DESKTOP-R41R7JE MINGW64 ~/Desktop/projects/Monsiteweb (master)
$ |
```

➤ Git status ? lisez les commentaires

```
Youcode@DESKTOP-R41R7JE MINGW64 ~/Desktop/projects/Monsiteweb (master)
$ git status
On branch master
Your branch and 'origin/master' have diverged,
and have 1 and 1 different commits each, respectively.
  (use "git pull" to merge the remote branch into yours)

nothing to commit, working tree clean
```

➤ Maintenant si vous comprenez le commentaire, on a besoin de faire un rebase : git pull --rebase

```
Youcode@DESKTOP-R41R7JE MINGW64 ~/Desktop/projects/Monsiteweb (master)
$ git pull --rebase
First, rewinding head to replay your work on top of it...
Applying: Editez le fichier index.html
```

➤ Exécutez l'alias que vous avez créé dans le scénario #1 (historique) ? qu'est-ce que vous constatez sur l'arborescence des branches ?

```
Youcode@DESKTOP-R41R7JE MINGW64 ~/Desktop/projects/Monsiteweb (master)
$ git log --oneline --graph --decorate --all
* 503544d (HEAD -> master) Editez le fichier index.html
* 7cccb42 (origin/master, origin/HEAD) ajoutez updates
* c294e19 changer Apache 2.0 par Apache 3.0
* 94589af (origin/updatelicensc) Merge branch 'master' of https://github.com/salma-hu/Monsiteweb
  \
  * f3380cc Merge pull request #1 from Salma-hu/annulation
  |
  * adec765 (origin/ajustement) modifier le fichier main.css
  |
  * b193b43 (origin/annulation) supprimez la balise H1
  * 3d1f221 new commit
  /
  * e0f4a2c (origin/Example) Update README.md
  /
* 5e1c081 Rename 404.html to error404.html
* 181caca Update index.html
* 1c54e06 Create style.css
* c8ba7bf Merge branch 'master' of https://github.com/Salma-hu/Monsiteweb
  \
  * 01546d2 Update index.html
  * 5045031 modifier repo local
  /
* 83c5701 ajouter initializr
* 20a916a Update README.md
* 13d27af Initial commit

Youcode@DESKTOP-R41R7JE MINGW64 ~/Desktop/projects/Monsiteweb (master)
$ |
```

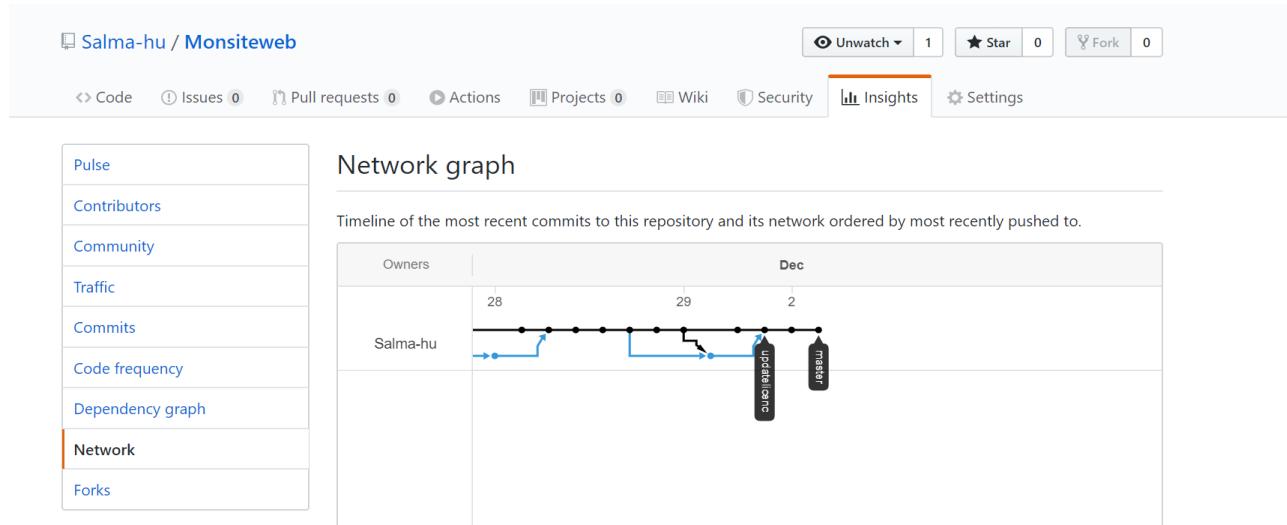
➤ Vérifiez que tous les changements que vous avez faits sont OK

➤ Expliquez l'utilité du REBAS

Rebase est un moyen d'intégrer les changements d'une branche à une autre. Rebase compresse tous les changements en un seul «patch». Ensuite, il intègre le patch sur la branche cible. Contrairement à la fusion, le rebasement aplati l'historique, car il transfère le travail terminé d'une branche à une autre. Dans le processus, l'historique non désiré est éliminé. Les "rebasements" indiquent comment les modifications doivent être transférées du sommet de la hiérarchie vers le bas, et les fusions indiquent comment elles remontent vers le haut.

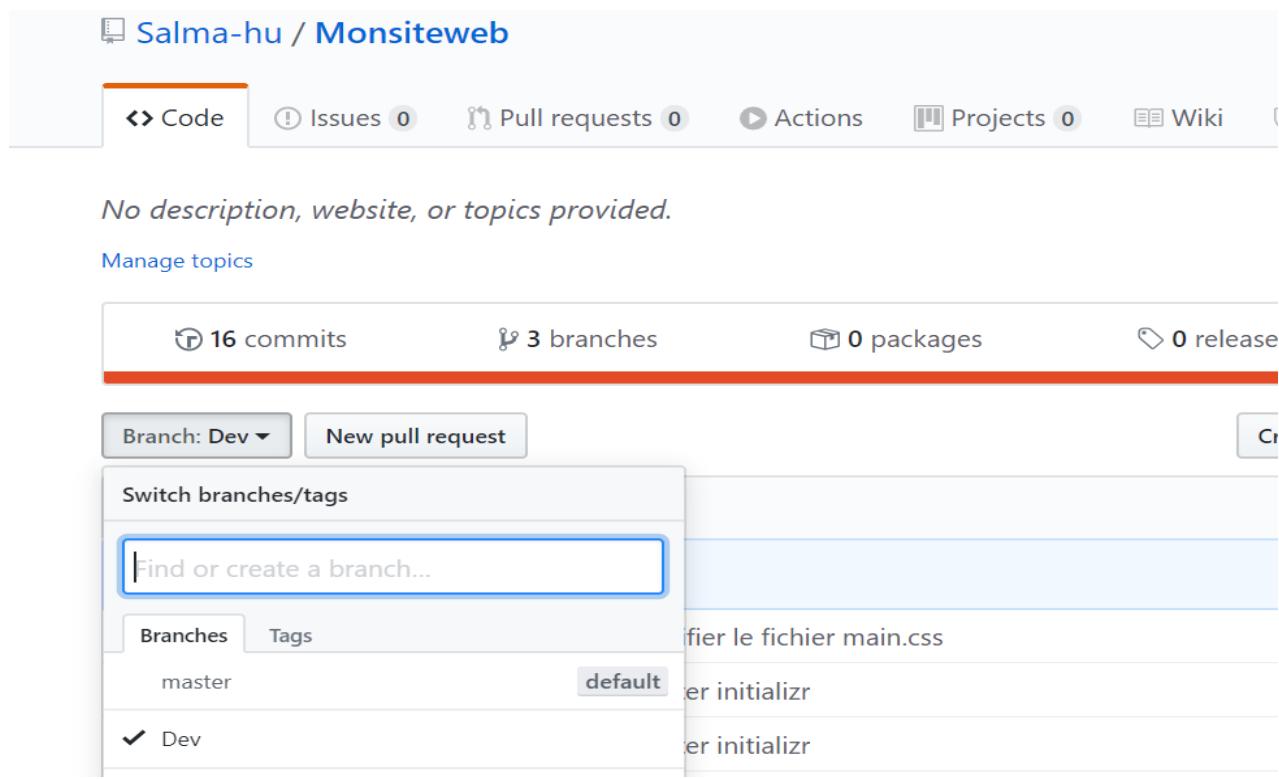
Seventh step : GitHub Insights

- Sur Github vous avez un menu INSIGHTS ; Explorez votre activité à travers insights

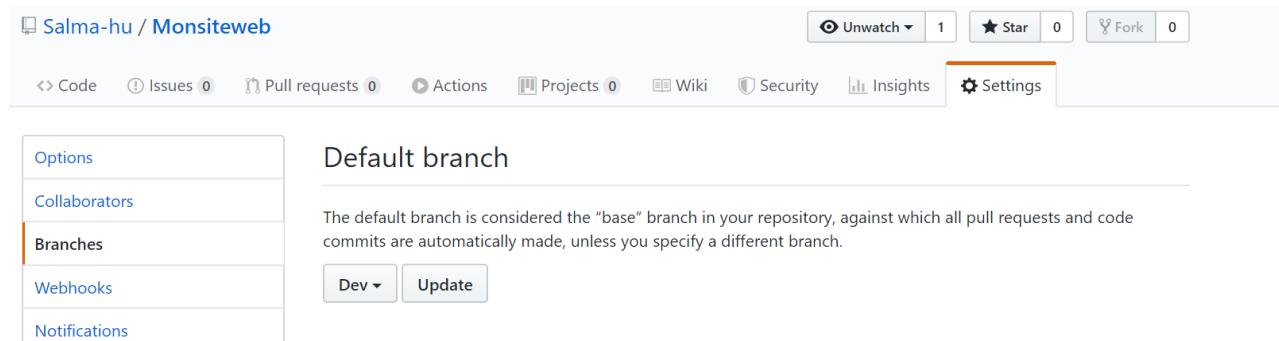


eighth step: Default branch and conflicts

- Sur github créez une branche du nom “Dev”



- La Branche Dev deviendra la branche par défaut ; sur le menu « settings » mettez la branche Dev en mode par défaut



- Créez une autre branche « demo », on va la considérer comme « feature»

No description, website, or topics provided.

[Edit](#)

Manage topics

16 commits 4 branches 0 packages 0 releases 1 contributor Apache-2.0

Branch: demo New pull request

Create new file Upload files Find file Clone or download

Switch branches/tags

Find or create a branch...

Branches	Tags	Commit Message	Date
master	default	fier le fichier main.css	4 days ago
Dev		er initializr	6 days ago
demo		er initializr	6 days ago
		commit	6 days ago

Pull request Compare Latest commit 7cccb42 16 hours ago

- Avant de faire un pull Request sur la branche demo , Editez le fichier « README.md » , Ajoutez une ligne

Monsiteweb / README.md Cancel

Edit file Preview changes

```
1 # Monsiteweb-local
2 premier site web
3 updates
4 updates 3
5
```

updates 3 readme.md #3

[Open](#) Salma-hu wants to merge 1 commit into master from demo

Conversation 0 Commits 1 Checks 0 Files changed 1 +1 -0

Salma-hu commented 2 minutes ago
Editez le fichier README.md

Salma-hu updated 3 readme.md

Owner Verified c7786f4

Reviewers No reviews

Assignees No one—assign yourself

Labels

- Ne faites aucun merge ni Rebbase mais plutôt supprimez la branche feature depuis GITHUB

Your branches

demo	Updated 1 hour ago by Ned-med	0 0	New pull request	
master	Updated 1 hour ago by Ned-med	0 0	New pull request	

Active branches

master	Updated 1 hour ago by Ned-med	0 0	New pull request	
demo	Updated 1 hour ago by Ned-med	0 0	New pull request	

- Sur GIT, supprimez récursivement et localement votre repo websit

⇒ Taper la commande : rm -rf Monsiteweb

- Maintenant Clonez via https ou SSH si vous l'avez réussi

```
Youcode@DESKTOP-R41R7JE MINGW64 ~/Desktop/projects
$ git clone https://github.com/Salma-hu/Monsiteweb.git
Cloning into 'Monsiteweb'...
remote: Enumerating objects: 70, done.
remote: Counting objects: 100% (70/70), done.
remote: Compressing objects: 100% (44/44), done.
remote: Total 70 (delta 25), reused 57 (delta 20), pack-reused 0
Unpacking objects: 100% (70/70), done.

Youcode@DESKTOP-R41R7JE MINGW64 ~/Desktop/projects
$ |
```

- Vérifiez si le contenu est récupéré

```
Youcode@DESKTOP-R41R7JE MINGW64 ~/Desktop/projects/Monsiteweb (master)
$ ls
css/  error404.html  fonts/  index.html  js/  LICENSE  README.md

Youcode@DESKTOP-R41R7JE MINGW64 ~/Desktop/projects/Monsiteweb (master)
```

- Vérifiez les branches et où pointe le pointeur : HEAD

```
Youcode@DESKTOP-R41R7JE MINGW64 ~/Desktop/projects/Monsiteweb (master)
$ git log --oneline --graph --decorate --all
* a54f907 (HEAD -> master, origin/master, origin/HEAD) updates 3 readme.md
* 7ccbcb42 (origin/Dev) ajoutez updates
* c294e19 changer Apache 2.0 par Apache 3.0
* 94589af (origin/updatelicenc) Merge branch 'master' of https://github.com/Salma-hu/Monsiteweb
|\
| * f3380cc Merge pull request #1 from Salma-hu/annulation
| * adec765 modifier le fichier main.css
|/
| * b193b43 supprimez la balise H1
| * 3d1f221 new commit
|
* 5e1c081 Rename 404.html to error404.html
* 181caca Update index.html
* 1c54e06 Create style.css
* c8ba7bf Merge branch 'master' of https://github.com/Salma-hu/Monsiteweb
|\
| * 01546d2 Update index.html
| * 5045031 modifier repo local
|/
| * 83c5701 ajouter initializr
* 20a916a Update README.md
* 13d27af Initial commit

Youcode@DESKTOP-R41R7JE MINGW64 ~/Desktop/projects/Monsiteweb (master)
$ |
```

- Maintenant on a besoin de la branche master comme base : git checkout master

```
Youcode@DESKTOP-R41R7JE MINGW64 ~/Desktop/projects/Monsiteweb (master)
$ git checkout master
Already on 'master'
Your branch is up to date with 'origin/master'.

Youcode@DESKTOP-R41R7JE MINGW64 ~/Desktop/projects/Monsiteweb (master)
$
```



- Sur GitHub, on va éditez le fichier Readme.md

Branch: master ▾ Monsiteweb / README.md

Find file Copy path

Salma-hu éditez le fichier a3a46d7 now

1 contributor

5 lines (5 sloc) | 67 Bytes

Raw Blame History

Monsiteweb-local

premier site web updates updates 3 updates 4

- Localement on est sur la branche DEV
- On a fait des changements sur le REPO distant, surement on a besoin d'un fetch

```
$ git fetch
remote: Enumerating objects: 5, done.
remote: Counting objects: 100% (5/5), done.
remote: Compressing objects: 100% (3/3), done.
remote: Total 3 (delta 1), reused 0 (delta 0), pack-reused 0
Unpacking objects: 100% (3/3), done.
From https://github.com/Salma-hu/Monsiteweb
  a54f907..a3a46d7  master      -> origin/master
```

- On va provoquer un sous-entendu, on va éditez le fichier README.md localement en sachant qu'on a déjà modifié le fichier sur le repo distant.

```
Youcode@DESKTOP-R41R7JE MINGW64 ~/Desktop/projects/Monsiteweb (master)
$ echo "##my local changes to readme" >> readme.md

Youcode@DESKTOP-R41R7JE MINGW64 ~/Desktop/projects/Monsiteweb (master)
$ git status
On branch master
Your branch is behind 'origin/master' by 1 commit, and can be fast-forwarded.
  (use "git pull" to update your local branch)

Changes not staged for commit:
  (use "git add <file>..." to update what will be committed)
  (use "git restore <file>..." to discard changes in working directory)
    modified:   README.md

no changes added to commit (use "git add" and/or "git commit -a")

Youcode@DESKTOP-R41R7JE MINGW64 ~/Desktop/projects/Monsiteweb (master)
$ |
```

➤ On va faire le commit

```
Youcode@DESKTOP-R41R7JE MINGW64 ~/Desktop/projects/Monsiteweb (master)
$ git commit -am "adding one line "
warning: LF will be replaced by CRLF in README.md.
The file will have its original line endings in your working directory
[master 25ec5fe] adding one line
 1 file changed, 1 insertion(+)

Youcode@DESKTOP-R41R7JE MINGW64 ~/Desktop/projects/Monsiteweb (master)
$ |
```

➤ Normalement, un des membres de l'équipe demande un pull : git pull

```
$ git pull
Auto-merging readme.md
Merge made by the 'recursive' strategy.
 readme.md | 1 +
 1 file changed, 1 insertion(+)
```

➤ Un message s'affiche indiquant que GIT n'arrive pas à traquer les informations et qu'il y'a un problème du merging, vu que le merge automatique n'est pas autorisé

```
#Made More Changes
made more changes to readme file
Updates
##my local changes to readme
~
```

conclusion

Pour conclure, ce projet nous a permis de stimuler notre créativité. Celle-ci passe par la création d'un bref projet , et des idées différentes que nous avons essayé de mettre en oeuvre. De plus, nous avons fait un fichier drive pour recueillir des informations sur lequel est retrancrit notre travail de groupe. En outre, afin de nous différencier, nous avons également fait un platforme trello .Outil d'organisation collaboratif

Par ailleurs, lors de ce travail nous avons remarqué qu'au final, nous avons eu une bonne cohésion de groupe. Etant donné qu'au début du projet nous avons tous des idées différentes, nous avons toutefois réussi à nous mettre d'accord sur une trame de projet.

Afin de suivre cette trame, nous avons donc du nous organiser. C'est à dire que nous avons partagé le travail en groupe de 5 personnes afin de gagner du temps. Nous nous sommes par la suite réunis plusieurs fois afin de mettre en commun nos travaux.

Nous avons également fait en sorte de contenter les idées de tous les membres du groupe. C'est pourquoi nous avons

organisé les idées. Afin de les réaliser, nous nous sommes même découvert une âme d'acteur pour chacun d'entre nous. C'est pourquoi Nous avons divisé les rôles de chacun nous avons attribué les taches afin que cela soit le plus cohérent possible.

Pour ce projet, nous nous sommes donc perfectionnés sur des logiciels et des sites tels que «trello » Parmi les outils collaboratifs de gestion de projet, Trello est une référence en la matière. Même s'il lui manque encore des fonctionnalités indispensables, il demeure un gestionnaire de projet intéressant pour tous ceux qui travaillent et gèrent des missions à distance et en équipe.