



RÉALISE PAR :

Ayoub Rouiha

Mouhsine Ech-choufi

Yassine Elbouanani

Mouna Eddibi

Kawtar Arkoubi

RAPPORT BRIEF PROJET

Scenario 1

FIRST STEP : IMMERSION

- La Définition git :**

Git est un logiciel de gestion de versions décentralisé .

- **le fonctionnement du fichier caché « .git » :**

Le fichier caché «.git » indique que la répertoire créer est devenu lié avec le Git et c'est grâce à la commande « git init » .

- **Sur votre bureau créez un répertoire nommée « /projects » .**
- **Créez un Repository Local sous le nom « /demo » .**
- **Vérifiez le statut du répertoire « /demo »**

```
Youcode@DESKTOP-STUJ4PP MINGW64 ~/Desktop (master)
$ mkdir projects

Youcode@DESKTOP-STUJ4PP MINGW64 ~/Desktop (master)
$ cd projects/

Youcode@DESKTOP-STUJ4PP MINGW64 ~/Desktop/projects (master)
$ mkdir demo

Youcode@DESKTOP-STUJ4PP MINGW64 ~/Desktop/projects (master)
$ cd demo/
Youcode@DESKTOP-STUJ4PP MINGW64 ~/Desktop/projects/demo (master)
$ git init
Initialized empty Git repository in C:/Users/Youcode/Desktop/projects/demo/.git/
Youcode@DESKTOP-STUJ4PP MINGW64 ~/Desktop/projects/demo (master)
$ git status
On branch master

No commits yet

nothing to commit (create/copy files and use "git add" to track)
```

- **Expliquez le commentaire.**

Le message affiché indique que ce repo local contient une branche master, et qu'il y'a pas des fichiers ou modification à enregistrer.

- **Créez le fichier README.md et ajouter la ligne suivante : '#Demo project un simple fichier'.**
- **Fait es le staging et le committing avec un commentaire.**

```
Youcode@DESKTOP-STUJ4PP MINGW64 ~/Desktop/projects/demo (master)
$ touch README.md

Youcode@DESKTOP-STUJ4PP MINGW64 ~/Desktop/projects/demo (master)
$ git commit -am "premier commit crier par readme"
On branch master
Initial commit

Untracked files:
  README.md

nothing added to commit but untracked files present

Youcode@DESKTOP-STUJ4PP MINGW64 ~/Desktop/projects/demo (master)
$ git add .

Youcode@DESKTOP-STUJ4PP MINGW64 ~/Desktop/projects/demo (master)
$ git commit -m "premier commit crier par readme"
[master (root-commit) f4ba609] premier commit crier par readme
 1 file changed, 0 insertions(+), 0 deletions(-)
 create mode 100644 README.md

Youcode@DESKTOP-STUJ4PP MINGW64 ~/Desktop/projects/demo (master)
$ vim README.md

Youcode@DESKTOP-STUJ4PP MINGW64 ~/Desktop/projects/demo (master)
$ git status
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")

Youcode@DESKTOP-STUJ4PP MINGW64 ~/Desktop/projects/demo (master)
$ git commit -am "premier modification sur readme.md"
warning: LF will be replaced by CRLF in README.md.
The file will have its original line endings in your working directory
[master 165f6af] premier modification sur readme.md
 1 file changed, 2 insertions(+)
```

SECOND STEP : LA DÉCOUVERTE

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

```
Youcode@DESKTOP-STUJ4PP MINGW64 ~/Desktop/projects/demo (master)
$ cd .git

Youcode@DESKTOP-STUJ4PP MINGW64 ~/Desktop/projects/demo/.git (GIT_DIR!)
$ ls -al
total 21
drwxr-xr-x 1 Youcode 197121 0 déc. 10 19:44 .
drwxr-xr-x 1 Youcode 197121 0 déc. 10 19:43 ../
-rw-r--r-- 1 Youcode 197121 35 déc. 10 19:44 COMMIT_EDITMSG
-rw-r--r-- 1 Youcode 197121 130 déc. 10 19:39 config
-rw-r--r-- 1 Youcode 197121 73 déc. 10 19:39 description
-rw-r--r-- 1 Youcode 197121 23 déc. 10 19:39 HEAD
drwxr-xr-x 1 Youcode 197121 0 déc. 10 19:39 hooks/
-rw-r--r-- 1 Youcode 197121 137 déc. 10 19:44 index
drwxr-xr-x 1 Youcode 197121 0 déc. 10 19:39 info/
drwxr-xr-x 1 Youcode 197121 0 déc. 10 19:43 logs/
drwxr-xr-x 1 Youcode 197121 0 déc. 10 19:44 objects/
drwxr-xr-x 1 Youcode 197121 0 déc. 10 19:39 refs/

Youcode@DESKTOP-STUJ4PP MINGW64 ~/Desktop/projects/demo/.git (GIT_DIR!)
$ ls
COMMIT_EDITMSG config description HEAD hooks/ index info/ logs/ objects/ refs/
```

Expliquer les clauses suivantes : HEAD, LOGS, BRANCHES :

LES BRANCHES : permettent de créer une version en parallèle du code à un moment donné dans l'historique .

HEAD : c'est une référence symbolique à la branche courante (pointeur) .

LOGS : c'est un journal ou historique d'évènements effectués qui sont datés et classés par ordre chronologique .

- Retournez vers votre Repo et créez le fichier « Licence.md » .
- Faites-le Commit .
- Affichez les fichiers traqués .

```
Youcode@DESKTOP-STUJ4PP MINGW64 ~/Desktop/projects/demo/.git (GIT_DIR!)
$ cd ..

Youcode@DESKTOP-STUJ4PP MINGW64 ~/Desktop/projects/demo (master)
$ touch Licence.md

Youcode@DESKTOP-STUJ4PP MINGW64 ~/Desktop/projects/demo (master)
$ git add .

Youcode@DESKTOP-STUJ4PP MINGW64 ~/Desktop/projects/demo (master)
$ git commit -m "premier commit licence.md"
error: pathspec 'premier commit licence.md' did not match any file(s) known to git
error: pathspec 'premier commit licence.md' did not match any file(s) known to git

Youcode@DESKTOP-STUJ4PP MINGW64 ~/Desktop/projects/demo (master)
$ git commit -m "premier commit licence.md"
[master 634c638] premier commit licence.md
 1 file changed, 0 insertions(+), 0 deletions(-)
 create mode 100644 Licence.md

Youcode@DESKTOP-STUJ4PP MINGW64 ~/Desktop/projects/demo (master)
$ git status
On branch master
nothing to commit, working tree clean

Youcode@DESKTOP-STUJ4PP MINGW64 ~/Desktop/projects/demo (master)
$ ls
Licence.md  README.md
```

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-STUJ4PP MINGW64 ~/Desktop/projects/demo (master)
$ git log -1
commit 634c638b9d83677600fdbe9fdf2e74ac8a397ebb (HEAD -> master)
Author: ech-mou <echmouhsine1994@gmail.com>
Date:   Tue Dec 10 19:52:55 2019 +0100

    premier commit licence.md

Youcode@DESKTOP-STUJ4PP MINGW64 ~/Desktop/projects/demo (master)
$ git log -1 --graph
* commit 634c638b9d83677600fdbe9fdf2e74ac8a397ebb (HEAD -> master)
| Author: ech-mou <echmouhsine1994@gmail.com>
| Date:   Tue Dec 10 19:52:55 2019 +0100
|
|     premier commit licence.md
```

- Créez un alias de la commande précédente le nom de l'alias est : **historique**
- Affichez la liste des alias
- Affichez l'historique des commit du fichier README.md avec l'alias.

```

Youcode@DESKTOP-FIDNSP2 MINGW64 ~/Desktop/projects/demo (master)
$ git config --global alias.historique 'log -1'

Youcode@DESKTOP-FIDNSP2 MINGW64 ~/Desktop/projects/demo (master)
$ git config --global --list
user.name=BlackDiamond-UX
user.email=rouiha594@gmail.com
alias.historique=log -1

Youcode@DESKTOP-FIDNSP2 MINGW64 ~/Desktop/projects/demo (master)
$ git historique README.md
expansion of alias 'historique' failed; 'log-1' is not a git command

Youcode@DESKTOP-FIDNSP2 MINGW64 ~/Desktop/projects/demo (master)
$ git config --global alias.historique 'log -1'

Youcode@DESKTOP-FIDNSP2 MINGW64 ~/Desktop/projects/demo (master)
$ git historique README.md
commit 8165ba4b72334fbfb785dd4514624ca89d2af7
Author: BlackDiamond-UX <rouiha594@gmail.com>
Date:   Wed Dec 11 11:54:52 2019 +0100

    premier modification

```

FOURTH STEP : EXCLUDING FILES

- **Renommer le fichier Licence.md à Licence.txt**
- **Faites le staging avec mise à jour (ne pas faire « git add . »)**
- **Faites-le commit.**
- **Supposons que nous développons sur notre plateforme, sûrement on a des fichiers qu'on veut exclure de notre arborescence du repository Local**
 - ✓ Créez un fichier nommé application.log
 - ✓ Ne faites pas le staging mais créez un fichier nommé « .gitignore »
 - ✓ Sur le fichier .gitignore Ajoutez la ligne suivante « *.log » d. Faites le staging et le commit

```

Youcode@DESKTOP-FIDNSP2 MINGW64 ~/Desktop/projects/demo (master)
$ git mv licence.md Licence.txt
Youcode@DESKTOP-FIDNSP2 MINGW64 ~/Desktop/projects/demo (master)
$ git add Licence.txt
Youcode@DESKTOP-FIDNSP2 MINGW64 ~/Desktop/projects/demo (master)
$ git commit -m "deuxieme modification sur licence.txt"
[master a3e81d5] deuxieme modification sur licence.txt
 1 file changed, 0 insertions(+), 0 deletions(-)
 rename Licence.md => Licence.txt (100%)
Youcode@DESKTOP-FIDNSP2 MINGW64 ~/Desktop/projects/demo (master)
$ touch application.log
Youcode@DESKTOP-FIDNSP2 MINGW64 ~/Desktop/projects/demo (master)
$ touch .gitignore
Youcode@DESKTOP-FIDNSP2 MINGW64 ~/Desktop/projects/demo (master)
$ vim .gitignore
Youcode@DESKTOP-FIDNSP2 MINGW64 ~/Desktop/projects/demo (master)
$ git add .
warning: LF will be replaced by CRLF in .gitignore.
The file will have its original line endings in your working directory
Youcode@DESKTOP-FIDNSP2 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-FIDNSP2 MINGW64 ~/Desktop/projects/demo (master)
$ git commit -m "commit file .gitignore"
[master caae8a5] commit file .gitignore
 1 file changed, 1 insertion(+)
 create mode 100644 .gitignore

```

- ✓ Qu'est-ce que vous constatez ?

On constate que si un fichier mentionné sur le fichier '.gitignore', il va pas être ajouté et enregistré dans une commit puis il n'appartient pas à l'arborescence.

FIFTH STEP : BRANCHING AND MERGING

- **Modifiez le fichier README.md**
- **Ne faites pas le commit**
- **Créez une branche pour la modification du fichier README.md du nom ‘updates’Faites le staging et le commit en une seule ligne.**
- **Affichez l'historique avec l'alias**
- **Qu'est-ce que vous constatez ?**
- **Retournez vers la branche Master.**
- **Affichez l'historique avec l'alias**

```
Youcode@DESKTOP-FIDNSP2 MINGW64 ~/Desktop/projects/demo (master)
$ vim README.md

Youcode@DESKTOP-FIDNSP2 MINGW64 ~/Desktop/projects/demo (master)
$ git checkout -b updates
Switched to a new branch 'updates'

Youcode@DESKTOP-FIDNSP2 MINGW64 ~/Desktop/projects/demo (updates)
$ git commit -am "premier commit sur br updates"
warning: LF will be replaced by CRLF in README.md.
The file will have its original line endings in your working directory
[updates ca2ea90] premier commit sur br updates
 1 file changed, 1 insertion(+)

Youcode@DESKTOP-FIDNSP2 MINGW64 ~/Desktop/projects/demo (updates)
$ git historique
commit ca2ea9025254a27d6fab1b25ab9cffbc10be639d (HEAD -> updates)
Author: BlackDiamond-UX <rouriha594@gmail.com>
Date:   Wed Dec 11 13:32:01 2019 +0100

    premier commit sur br updates

Youcode@DESKTOP-FIDNSP2 MINGW64 ~/Desktop/projects/demo (updates)
$ git switch master
Switched to branch 'master'

Youcode@DESKTOP-FIDNSP2 MINGW64 ~/Desktop/projects/demo (master)
$ git historique
commit caae8a5143a7ed26b889cca6765ad96bf998f775 (HEAD -> master)
Author: BlackDiamond-UX <rouriha594@gmail.com>
Date:   Wed Dec 11 12:05:34 2019 +0100

    commit file .gitignore

Youcode@DESKTOP-FIDNSP2 MINGW64 ~/Desktop/projects/demo (master)
$ git merge updates
Updating caae8a5..ca2ea90
Fast-forward
 README.md | 1 +
 1 file changed, 1 insertion(+)
```

- **Qu'est-ce que vous constatez**

L'historique de la branche updates affiche la nouvelle commit plus les commits faites au niveau de branche master.

- **Maintenant faites le merge.**

SIXTH STEP : CONFLICT RESOLUTION

- **Créez une branche avec le nom ‘BAD’**
- **Modifiez le fichier README.md et ajouter la ligne ‘Trouble’**
- **Faites le staging et le commit en une seule ligne 4. Switchez vers la branche principale ‘master’**
- **Maintenant modifiez le fichier README.md et ajoutez la ligne ‘Troubleshooting’**

- **Staging/committing avec commentaire ‘branche bien faite’**
- **Oups !! corriger le commentaire du dernier commit**
- **Affichez l'historique**

```
Youcode@DESKTOP-FIDNSP2 MINGW64 ~/Desktop/projects/demo (master)
$ git checkout -b BAD
Switched to a new branch 'BAD'

Youcode@DESKTOP-FIDNSP2 MINGW64 ~/Desktop/projects/demo (BAD)
$ vim README.md

Youcode@DESKTOP-FIDNSP2 MINGW64 ~/Desktop/projects/demo (BAD)
$ git commit -am "ecrit trouble sur readme"
[BAD 699e3f9] écrit trouble sur readme
 1 file changed, 1 insertion(+)

Youcode@DESKTOP-FIDNSP2 MINGW64 ~/Desktop/projects/demo (BAD)
$ git switch master
Switched to branch 'master'

Youcode@DESKTOP-FIDNSP2 MINGW64 ~/Desktop/projects/demo (master)
$ vim README.md

Youcode@DESKTOP-FIDNSP2 MINGW64 ~/Desktop/projects/demo (master)
$ git commit -am "branche bien fait"
[master 9b736a7] branche bien fait
 1 file changed, 1 insertion(+)

Youcode@DESKTOP-FIDNSP2 MINGW64 ~/Desktop/projects/demo (master)
$ git commit --amend -m "écrit Troubleshooting sur readme"
[master 8adcc48] écrit Troubleshooting sur readme
 Date: Wed Dec 11 13:39:45 2019 +0100
 1 file changed, 1 insertion(+)

Youcode@DESKTOP-FIDNSP2 MINGW64 ~/Desktop/projects/demo (master)
$ git h
help      historique

Youcode@DESKTOP-FIDNSP2 MINGW64 ~/Desktop/projects/demo (master)
```

```
Youcode@DESKTOP-FIDNSP2 MINGW64 ~/Desktop/projects/demo (master)
$ git historique
commit 8adcc48721e9d91558cda700balea0480a564e2e (HEAD -> master)
Author: BlackDiamond-UX <rouriha594@gmail.com>
Date:   Wed Dec 11 13:39:45 2019 +0100

    écrit Troubleshooting sur readme

Youcode@DESKTOP-FIDNSP2 MINGW64 ~/Desktop/projects/demo (master)
$ git log
commit 8adcc48721e9d91558cda700balea0480a564e2e (HEAD -> master)
Author: BlackDiamond-UX <rouriha594@gmail.com>
Date:   Wed Dec 11 13:39:45 2019 +0100

    écrit Troubleshooting sur readme

commit caae8a5143a7ed26b889cca6765ad96bf998f775
Author: BlackDiamond-UX <rouriha594@gmail.com>
Date:   Wed Dec 11 12:05:34 2019 +0100

    premier commit sur br updates

commit caae8a5143a7ed26b889cca6765ad96bf998f775
Author: BlackDiamond-UX <rouriha594@gmail.com>
Date:   Wed Dec 11 12:05:34 2019 +0100

    commit file .gitignore

commit a3e81d57be63575e7c6ca57949cb8d19642911a2
Author: BlackDiamond-UX <rouriha594@gmail.com>
Date:   Wed Dec 11 12:00:24 2019 +0100

    deuxieme modification sur licence.txt

commit 65d5a5c727322929a3f456ef594346ff81bf4e
Author: BlackDiamond-UX <rouriha594@gmail.com>
Date:   Wed Dec 11 11:56:10 2019 +0100

    premier commit licence

commit 8165ba4b72334fbfbb785dd4514624ca89d2af7
Author: BlackDiamond-UX <rouriha594@gmail.com>
Date:   Wed Dec 11 11:54:52 2019 +0100

    premier modification

commit 65d1cb9d9c3e301968d06c15050bdb9abf3dbc8d
Author: BlackDiamond-UX <rouriha594@gmail.com>
```

```
Youcode@DESKTOP-FIDNSP2 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-FIDNSP2 MINGW64 ~/Desktop/projects/demo (master|MERGING)
$ cat README.md
#Demo project un simple fichier
Youcode
<===== HEAD
Troubleshooting
=====
Trouble
>>>>> BAD

Youcode@DESKTOP-FIDNSP2 MINGW64 ~/Desktop/projects/demo (master|MERGING)
$ git mergetool
git: 'mergetool' is not a git command. See 'git --help'.
The most similar command is
    mergetool

Youcode@DESKTOP-FIDNSP2 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 emerge p4merge araxis bc codecompare smerge emerge viendiff
Merging:
README.md

Normal merge conflict for 'README.md':
[local]: modified file
[remote]: modified file
Hit return to start merge resolution tool (viendiff):
4 fichiers à éditer
```

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

On constate que les modifications faites sur une branche restant indépendantes par rapport master et l'historique de master n'affiche pas les commit de la branche.

- Faites le merge de la branche 'BAD'**
- Expliquez le message ?**

Le message indique qu'il y a un conflit grâce à des modifications sur même ligne.

- Exécutez la commande suivante : cat README.md ? Expliquez ?**

Commande affiche deux modifications sur même ligne (trouble et troubleshooting) l'une au niveau de master et l'autre sur branch

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

On constate qu'il faut configurer et déclarer un gestionnaire de conflits avec le git.

SEVENTH STEP: MERGE TOOLS

- Installez p4merge sur votre PC**
- Sur git bash tapez la commande : git config --global -list**
- Git config --global merge.tool p4merge**
- Git config --global mergetool.p4merge.path "lien d'installation" (.exe)**
- Tapez la commande git mergetool**
- Analysez ce que vous voyez ? et expliquez la plateforme ouverte en temps réel ?**

```
Youcode@DESKTOP-FIDNSP2 MINGW64 ~/Desktop/projects/demo (master|MERGING)
$ git config --global --list
user.name=BlackDiamond-UX
user.email=rourih594@gmail.com
alias.historique=log -1

Youcode@DESKTOP-FIDNSP2 MINGW64 ~/Desktop/projects/demo (master|MERGING)
$ Git config --global merge.tool p4merge

Youcode@DESKTOP-FIDNSP2 MINGW64 ~/Desktop/projects/demo (master|MERGING)
$ git config --global merge_tool p4merge
user.name=Blackdiamond-UX
user.email=rourih594@gmail.com
alias.historique=log -1
merge.tool=p4merge

Youcode@DESKTOP-FIDNSP2 MINGW64 ~/Desktop/projects/demo (master|MERGING)
$ Git config --global merge.tool.p4merge.path 'C:/ProgramData/Microsoft/Windows/Start Menu/Programs/Perforce'
error: key does not contain a section: -global

Youcode@DESKTOP-FIDNSP2 MINGW64 ~/Desktop/projects/demo (master|MERGING)
$ Git config --global merge.tool.p4merge.path 'C:/ProgramData/Microsoft/Windows/Start Menu/Programs/Perforce'

Youcode@DESKTOP-FIDNSP2 MINGW64 ~/Desktop/projects/demo (master|MERGING)
$ mergertool
bash: mergertool: command not found

Youcode@DESKTOP-FIDNSP2 MINGW64 ~/Desktop/projects/demo (master|MERGING)
$ git mergertool
git config option merge.tool set to unknown tool: p4merge
Resetting to default...

This message is displayed because 'merge.tool' is not configured.
See 'git mergertool --tool-help' or 'git help config' for more details.
'git mergertool' 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
No files need merging
```

Puisque le logiciel p4merge est configuré sur le git, il va se lancer par défaut au moment de taper la commande mergetool pour afficher l'emplacement de conflits. A ce moment-là nous devons faire un choix entre les modifications comme une solution.

EIGHTH STEP: CHALLENGE

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

```
Youcode@DESKTOP-FIDNSP2 MINGW64 ~/Desktop/projects/demo (master|MERGING)
$ cat .gitignore
*.log

Youcode@DESKTOP-FIDNSP2 MINGW64 ~/Desktop/projects/demo (master)
$ ls
application.log licence.txt README.md README.md.orig

Youcode@DESKTOP-FIDNSP2 MINGW64 ~/Desktop/projects/demo (master)
$ ls
application.log licence.txt README.md README.md.orig

Youcode@DESKTOP-FIDNSP2 MINGW64 ~/Desktop/projects/demo (master)
$ echo "*.orig" >> .gitignore

Youcode@DESKTOP-FIDNSP2 MINGW64 ~/Desktop/projects/demo (master)
$ cat .gitignore
*.log
*.orig
```

- **On laisse que les fichiers : licence.txt Readme.md .gitignore (exemple *.log pour application.log) .**

Tous les fichiers avec extension ‘.log’ sont exclues

Scénario 2

FIRST STEP : TAGGING

- **Déplacez-vous sur la branche Principale**
- **Créez un TAG avec un nom V1.0 et un commentaire ‘ RELEASE 1.0 ‘**
- **Affichez les informations sur le TAG**

```
Youcode@DESKTOP-FIDNSP2 MINGW64 ~/Desktop/projects/demo (master)
$ git tag -a v1.0 -m "RELEASE 1.0"

Youcode@DESKTOP-FIDNSP2 MINGW64 ~/Desktop/projects/demo (master)
$ git tag
v1.0

Youcode@DESKTOP-FIDNSP2 MINGW64 ~/Desktop/projects/demo (master)
$ git show v1.0
tag v1.0
Tagger: BlackDiamond-UX <rouriha594@gmail.com>
Date:   Wed Dec 11 16:34:38 2019 +0100

RELEASE 1.0

commit 8adcc48721e9d91558cda700ba1ea0480a564e2e (HEAD -> master, tag: v1.0)
Author: BlackDiamond-UX <rouriha594@gmail.com>
Date:   Wed Dec 11 13:39:45 2019 +0100

    écrit Troubleshooting sur readme

diff --git a/README.md b/README.md
index 01f6880..85de78b 100644
--- a/README.md
+++ b/README.md
@@ -1,2 +1,3 @@
 '#Demo project un simple fichier'
 moushine
+ Troubleshooting
```

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

Les gens utilisent cette fonctionnalité pour marquer les points de publication (v1.0, v2.0, etc.). Nous avons utilisé ici les annotated tags. Ils sont stockés en tant qu'objets complets dans la base de données Git. Ils sont contrôlés; contient le nom, l'adresse email et la date du tagueur; avoir un message de marquage .

SECOND STEP: STASHING AND SAVING WORK IN PROGRESS

- **Modifiez le fichier README.md , ajoutez une ligne .**

```
Youcode@DESKTOP-FIDNSP2 MINGW64 ~/Desktop/projects/demo (master)
$ vim README.md
```

- **A vrai dire, l'équipe a décidé que cette tâche sera en standby et que les modifications progressent avec le timeline du projet .**
 - Tapez la commande git Stash

```
Youcode@DESKTOP-FIDNSP2 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: 8adcc48 écrit Troubleshooting sur readme

Youcode@DESKTOP-FIDNSP2 MINGW64 ~/Desktop/projects/demo (master)
$ git status
On branch master
Untracked files:
  (use "git add <file>..." to include in what will be committed)
    README.md.orig

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

- Expliquez le fonctionnement de la commande git Stash :
 - ✓ Souvent, lorsque vous travaillez sur une partie de votre projet, la situation est dans un état désordonné et vous souhaitez changer de branche pour travailler sur autre chose. Le problème, c'est que vous ne voulez pas faire un commit à moitié fait pour pouvoir revenir à ce point plus tard. La réponse à ce problème est la commande git stash.
- Tapez la commande git stash list
- Exécutez la commande git status ?

```
Youcode@DESKTOP-FIDNSP2 MINGW64 ~/Desktop/projects/demo (master)
$ git stash list
stash@{0}: WIP on master: 8adcc48 écrit Troubleshooting sur readme

Youcode@DESKTOP-FIDNSP2 MINGW64 ~/Desktop/projects/demo (master)
$ git status
On branch master
Untracked files:
  (use "git add <file>..." to include in what will be committed)
    README.md.orig

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

- Qu'est-ce que vous constatez ?
 - ✓ la cachette est créée et la modification que nous avons faite désespérée
- **Modifiez le fichier « Licence.txt », ajoutez la ligne « APACHE 2.0 »**
- **Faites staging et le commit en une seule ligne**

- **Exécutez la commande « git stash pop »**

```
Youcode@DESKTOP-FIDNSP2 MINGW64 ~/Desktop/projects/demo (master)
$ vim licence.txt

Youcode@DESKTOP-FIDNSP2 MINGW64 ~/Desktop/projects/demo (master)
$ git commit -am "premier modification sur licence.txt"
warning: LF will be replaced by CRLF in licence.txt.
The file will have its original line endings in your working directory
[master 4b98fce] premier modification sur licence.txt
 1 file changed, 1 insertion(+)

Youcode@DESKTOP-FIDNSP2 MINGW64 ~/Desktop/projects/demo (master)
$ 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:   .gitignore
    modified:   README.md

no changes added to commit (use "git add" and/or "git commit -a")
Dropped refs/stash@{0} (9f606eee5b59a09c91bb472e2cdac01d7caa7730)
```

- **Qu'est-ce que vous constatez ?**

- ✓ après avoir exécuté la commande git stash pop, nous avons appliqué la modification cachée que nous avons effectuée, puis le stash a été supprimée.

THIRD STEP: VOYAGE SUR GITHUB, LOCAL REPO TO GITHUB REPO

- **Pour cette phase vous devez avoir un compte GITHUB**
- **Créez un repo github public sans ajouter le fichier README.md**

Owner: kawtar-ar / Repository name: * github-scenario-2

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

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.

Add .gitignore: None | Add a license: None

Create repository

- **Créez le remote en https**
- **Examinez le remote**
- **Pushez le tout à travers la commande : git push -u origin master --tags**

```
Youcode@DESKTOP-FIDNSP2 MINGW64 ~/Desktop/projects/demo (master)
$ git remote add origin https://github.com/BlackDiamond-UX/github-Scenario-2.git

Youcode@DESKTOP-FIDNSP2 MINGW64 ~/Desktop/projects/demo (master)
$ git remote -v
origin  https://github.com/BlackDiamond-UX/github-Scenario-2.git (fetch)
origin  https://github.com/BlackDiamond-UX/github-Scenario-2.git (push)

Youcode@DESKTOP-FIDNSP2 MINGW64 ~/Desktop/projects/demo (master)
$ git push -u origin master --tags
Enumerating objects: 23, done.
Counting objects: 100% (23/23), done.
Delta compression using up to 8 threads
Compressing objects: 100% (16/16), done.
Writing objects: 100% (23/23), 2.33 KiB | 298.00 KiB/s, done.
Total 23 (delta 1), reused 0 (delta 0)
remote: Resolving deltas: 100% (1/1), done.
To https://github.com/BlackDiamond-UX/github-Scenario-2.git
 * [new branch]      master -> master
 * [new tag]         v1.0 -> v1.0
Branch 'master' set up to track remote branch 'master' from 'origin'.
```

- Expliquez la commande ?
 - ✓ La commande utilisé pour transférer un dépôt local vers le github
- Expliquez les options –u et --tags ?
 - ✓ -u : signifie upstream c'est le dépôt Référentiel à partir duquel nous avons copié le projet.
 - ✓ --tags : ce sont des étiquettes pour marquer un certain état dans l'historique comme important
- **Sur Github Vérifiez la liste des commits , les branches , les releases et les tags**
 - ✓ les branches n'ont pas poussé pour faire ça on doit exécuter le command suivant git push --all

FOURTH STEP: MINI CHALLENGE (OPTIONNEL)

- **Créez un dossier sous git nommé .SSH**
- **Déplacez-vous dans le fichier .SSH**
- **Maintenant à vos mains : Créez une authentification sécurisé SSH entre votre repo local et votre repo distant (aide : Ssh –keygen -t rsa –C 'email'). Fifth Step: Création d'une local copy**

```
Youcode@DESKTOP-FIDNSP2 MINGW64 ~/Desktop/projects/demo (master)
$ mkdir .SSH

Youcode@DESKTOP-FIDNSP2 MINGW64 ~/Desktop/projects/demo (master)
$ cd SSH
bash: cd: SSH: No such file or directory

Youcode@DESKTOP-FIDNSP2 MINGW64 ~/Desktop/projects/demo (master)
$ cd .SSH

Youcode@DESKTOP-FIDNSP2 MINGW64 ~/Desktop/projects/demo/.SSH (master)
$ ssh-keygen
Generating public/private rsa key pair.
Enter file in which to save the key (/c/Users/Youcode/.ssh/id_rsa): ssh
Enter passphrase (empty for no passphrase):
Enter same passphrase again:
Your identification has been saved in ssh.
Your public key has been saved in ssh.pub.
The key fingerprint is:
SHA256:/M1Mpem0G6TCCwOLpcx+E2zkG1/6KwCFaGqPTU181A Youcode@DESKTOP-FIDNSP2
The key's randomart image is:
+---[RSA 3072]---+
| ..E
| ...
| . O...
| o o+ o . o
| += o+ S o
| *oB.o. *
| ++oX * . =
| o.=o@ o..
| ..o.=++ o.
+---[SHA256]---
```

FIFTH STEP: CRÉATION 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 alicence.txt ‘APACHE 2.0‘**

No description, website, or topics provided.

Manage topics

6 commits	1 branch	0 packages	0 releases	1 contributor
-----------	----------	------------	------------	---------------

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

BlackDiamond-UX Merge branch 'master' of https://github.com/BlackDiamond-UX/Monsiteweb ...	Latest commit e274ae3 44 minutes ago
copy initializr repo	2 hours ago
.gitignore	3 hours ago
APACHE 2.0	3 hours ago
Premier commit Readme 2 sur monsiteweb-local	1 hour ago
ALICENCE.TXT	3 hours ago
premier commit sur index.html	1 hour ago

- **Déplacez-vous dans le répertoire projets sous GIT**

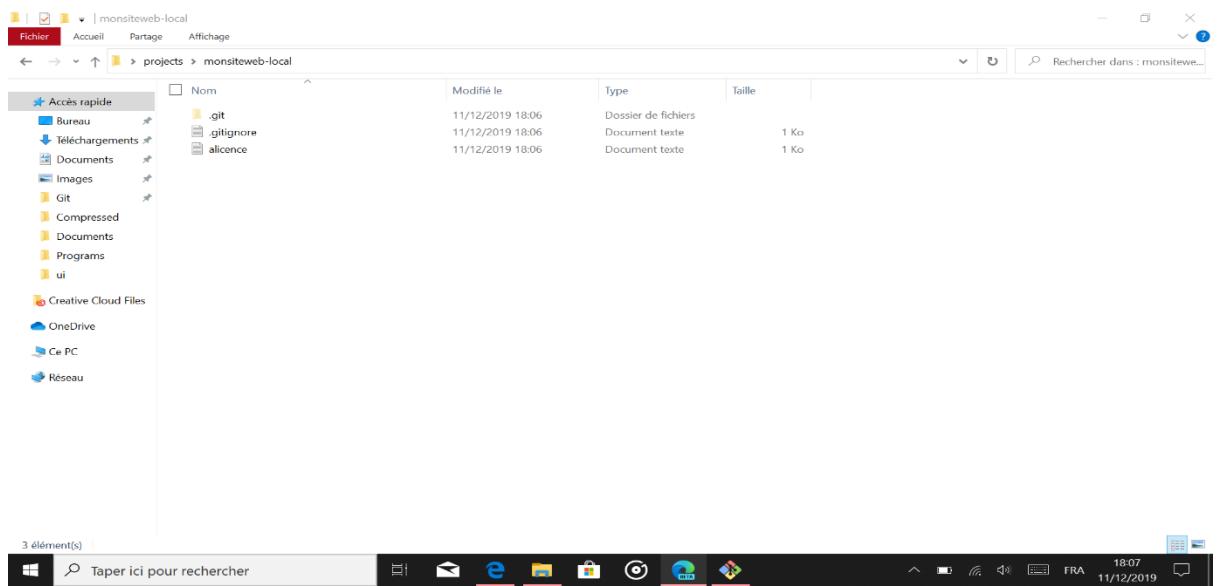
```
Youcode@DESKTOP-FIDNSP2 MINGW64 ~/Desktop/projects/demo/.SSH (master)
$ cd ..

Youcode@DESKTOP-FIDNSP2 MINGW64 ~/Desktop/projects/demo (master)
$ cd ..
```

- **Créez un clone Github vers le local sous le nom (Monsiteweb-local)**

```
Youcode@DESKTOP-FIDNSP2 MINGW64 ~/Desktop/projects
$ git clone https://github.com/BlackDiamond-UX/Monsiteweb.git monsiteweb-local
Cloning into 'monsiteweb-local'...
remote: Enumerating objects: 5, done.
remote: Counting objects: 100% (5/5), done.
remote: Compressing objects: 100% (3/3), done.
remote: Total 5 (delta 0), reused 0 (delta 0), pack-reused 0
Unpacking objects: 100% (5/5), done.
```

- **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

HTML/CSS Template HTML5 Polyfills jQuery

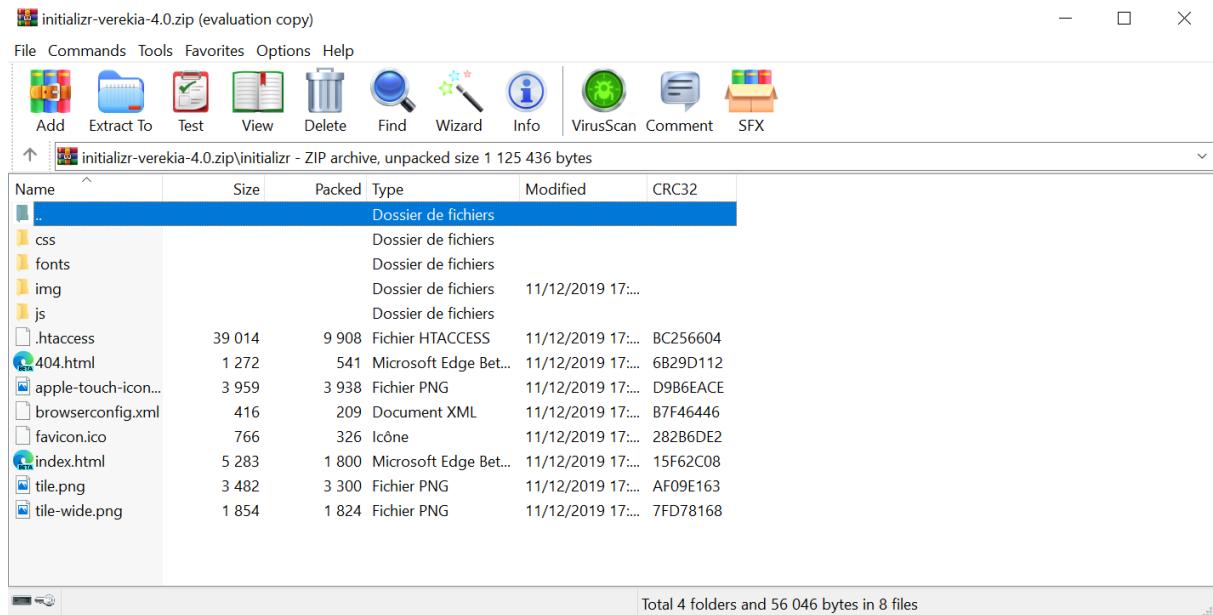
<input type="radio"/> No template	<input checked="" type="radio"/> Modernizr	<input checked="" type="checkbox"/> Minified
<input type="radio"/> Mobile-first Responsive	<input type="radio"/> Just HTML5shiv	<input type="checkbox"/> Development
<input checked="" type="radio"/> Twitter Bootstrap	<input checked="" type="checkbox"/> Respond - Alternatives	

H5BP Optional

<input type="checkbox"/> IE Classes	<input checked="" type="checkbox"/> Favicon	<input type="checkbox"/> Humans.txt
<input checked="" type="checkbox"/> Old browser warning	<input checked="" type="checkbox"/> Apple and Windows Icons	<input checked="" type="checkbox"/> 404 Page
<input checked="" type="checkbox"/> Google Analytics	<input type="checkbox"/> plugins.js	<input type="checkbox"/> Adobe Cross Domain
<input checked="" type="checkbox"/> .htaccess	<input type="checkbox"/> Robots.txt	

Download it! **What's inside?**

- **Analysez l'arborescence.**
- **Expliquez ?**



- **Copiez le site téléchargé dans votre repo local à travers une seule commande :**
- **Hint : cp -R ~/DESKTOP/initializr/* .**
- **Git status**

```
Youcode@DESKTOP-FIDNSP2 MINGW64 ~/Desktop/projects
$ cd monsiteweb-local/
Youcode@DESKTOP-FIDNSP2 MINGW64 ~/Desktop/projects/monsiteweb-local (master)
$ cp -r 'C:\Users\Youcode\Desktop\initializr' 'C:\Users\Youcode\Desktop\projects\monsiteweb-local'
Youcode@DESKTOP-FIDNSP2 MINGW64 ~/Desktop/projects/monsiteweb-local (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)
    initializr/

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-FIDNSP2 MINGW64 ~/Desktop/projects/monsiteweb-local (master)
$ git add . | git commit -m"copy initializr repo"
[master 3ef9d11] copy initializr repo
 25 files changed, 10534 insertions(+)
  create mode 100644 initializr/.htaccess
  create mode 100644 initializr/404.html
  create mode 100644 initializr/apple-touch-icon.png
  create mode 100644 initializr/browserconfig.xml
  create mode 100644 initializr/css/bootstrap-theme.css
  create mode 100644 initializr/css/bootstrap-theme.css.map
  create mode 100644 initializr/css/bootstrap-theme.min.css
  create mode 100644 initializr/css/bootstrap.css
  create mode 100644 initializr/css/bootstrap.css.map
  create mode 100644 initializr/css/bootstrap.min.css
  create mode 100644 initializr/css/main.css
  create mode 100644 initializr/favicon.ico
  create mode 100644 initializr/fonts/glyphicons-halflings-regular.eot
  create mode 100644 initializr/fonts/glyphicons-halflings-regular.svg
  create mode 100644 initializr/fonts/glyphicons-halflings-regular.ttf
  create mode 100644 initializr/fonts/glyphicons-halflings-regular.woff
  create mode 100644 initializr/index.html
  create mode 100644 initializr/js/main.js
  create mode 100644 initializr/js/vendor/bootstrap.js
  create mode 100644 initializr/js/vendor/bootstrap.min.js
  create mode 100644 initializr/js/vendor/jquery-1.11.2.min.js
  create mode 100644 initializr/js/vendor/modernizr-2.8.3-respond-1.4.2.min.js
  create mode 100644 initializr/js/vendor/npm.js
  create mode 100644 initializr/tile-wide.png
  create mode 100644 initializr/tile.png

Youcode@DESKTOP-FIDNSP2 MINGW64 ~/Desktop/projects/monsiteweb-local (master)
$ git log
commit 3ef9d11eaf9525040d53483f2528c218b144bd4a (HEAD -> master)
Author: BlackDiamond-UX <rouriha594@gmail.com>
Date:   Wed Dec 11 19:04:37 2019 +0100

    copy initializr repo
```

- Fait le push à github**

```
Youcode@DESKTOP-FIDNSP2 MINGW64 ~/Desktop/projects/monsiteweb-local (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-FIDNSP2 MINGW64 ~/Desktop/projects/monsiteweb-local (master)
$ git push
Enumerating objects: 32, done.
Counting objects: 100% (32/32), done.
Delta compression using up to 8 threads
Compressing objects: 100% (31/31), done.
Writing objects: 100% (31/31), 284.21 Kib | 5.57 MiB/s, done.
Total 31 (delta 3), reused 0 (delta 0)
remote: Resolving deltas: 100% (3/3), done.
To https://github.com/BlackDiamond-UX/Monsiteweb.git
  4d0d56b..3ef9d11  master -> master
```

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

No description, website, or topics provided.

Manage topics

2 commits	1 branch	0 packages	0 releases	2 contributors	Apache-2.0
Branch: master	New pull request	Create new file	Upload files	Find file	Clone or download
135579 committing website					
css	committing website	27 minutes ago			
fonts	committing website	27 minutes ago			
js	committing website	27 minutes ago			
.gitignore	Initial commit	2 hours ago			
404.html	committing website	27 minutes ago			
LICENSE	Initial commit	2 hours ago			
apple-touch-icon.png	committing website	27 minutes ago			
browserconfig.xml	committing website	27 minutes ago			
favicon.ico	committing website	27 minutes ago			
index.html	committing website	27 minutes ago			
tile-wide.png	committing website	27 minutes ago			
tile.png	committing website	27 minutes ago			

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

Screenshot of a GitHub commit interface showing the file content and commit message fields.

File Content:

```
<title>mon premier site web</title>
```

Commit Message:

premier commit sur index.html

Branch Selection:

Commit directly to the `master` branch.

Create a new branch for this commit and start a pull request. [Learn more about pull requests.](#)

Buttons:

Commit new file (green button) | **Cancel**

© 2019 GitHub, Inc. Terms Privacy Security Status Help Contact GitHub Pricing API Training Blog About

- **Sur git et sur le repo local, Editez le fichier README.md**
- **Faites le staging et le commit en une seule ligne**

```
Youcode@DESKTOP-FIDNSP2 MINGW64 ~/Desktop/projects/monsiteweb-local (master)
$ touch README.md

Youcode@DESKTOP-FIDNSP2 MINGW64 ~/Desktop/projects/monsiteweb-local (master)
$ vim README.md

Youcode@DESKTOP-FIDNSP2 MINGW64 ~/Desktop/projects/monsiteweb-local (master)
$ git add . | git commit -m"Premier commit Readme 2 sur monsiteweb-local"
warning: LF will be replaced by CRLF in README.md.
The file will have its original line endings in your working directory
[master 1877611] Premier commit Readme 2 sur monsiteweb-local
 1 file changed, 1 insertion(+)
 create mode 100644 README.md

Youcode@DESKTOP-FIDNSP2 MINGW64 ~/Desktop/projects/monsiteweb-local (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
```

- **Qu'est-ce que vous constatez ?**

- ✓ nous avons apporté des modifications à la fois dans notre repo distant et local
Donc, fondamentalement, nous avons deux commits différents

- **Trouble shooting :**

- **Executez la commande « git fetch »**
- **Git status**

```
Youcode@DESKTOP-FIDNSP2 MINGW64 ~/Desktop/projects/monsiteweb-local (master)
$ git fetch
remote: Enumerating objects: 4, done.
remote: Counting objects: 100% (4/4), done.
remote: Compressing objects: 100% (2/2), done.
remote: Total 3 (delta 0), reused 0 (delta 0), pack-reused 0
Unpacking objects: 100% (3/3), done.
From https://github.com/BlackDiamond-UX/Monsiteweb
  3ef9d11..fa27c10  master      -> origin/master

Youcode@DESKTOP-FIDNSP2 MINGW64 ~/Desktop/projects/monsiteweb-local (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-FIDNSP2 MINGW64 ~/Desktop/projects/monsiteweb-local (master)
$ ls
allicence.txt  initializr/  README.md
```

- **Expliquez?**

- ✓ Git fetch permet d'importer les modifications de dépôt distant sur le dépôt local ; mais ils nécessitent faire un merge .

- **Git pull**

```
Youcode@DESKTOP-FIDNSP2 MINGW64 ~/Desktop/projects/monsiteweb-local (master)
$ git pull
Merge made by the 'recursive' strategy.
 index.html | 1 +
 1 file changed, 1 insertion(+)
 create mode 100644 index.html
```

- **Git push**

```
Youcode@DESKTOP-FIDNSP2 MINGW64 ~/Desktop/projects/monsiteweb-local (master)
$ git push
Enumerating objects: 7, done.
Counting objects: 100% (7/7), done.
Delta compression using up to 8 threads
Compressing objects: 100% (4/4), done.
Writing objects: 100% (5/5), 580 bytes | 580.00 KiB/s, done.
Total 5 (delta 2), reused 0 (delta 0)
remote: Resolving deltas: 100% (2/2), completed with 1 local object.
To https://github.com/BlackDiamond-UX/Monsiteweb.git
  fa27c10..e274ae3 master -> master
```

- **Verifiez les commit sur github**

The screenshot shows a GitHub repository page for 'BlackDiamond-UX / Monsiteweb'. The repository has 6 commits, 1 branch, 0 packages, 0 releases, and 1 contributor. The latest commit is 'BlackDiamond-UX Merge branch 'master' of https://github.com/BlackDiamond-UX/Monsiteweb' made 44 minutes ago. Below the commit list, there are links for 'Create new file', 'Upload files', 'Find file', and a green 'Clone or download' button.

File	Commit Message	Time Ago
initializr	copy initializr repo	2 hours ago
.gitignore	APACHE 2.0	3 hours ago
README.md	Premier commit Readme 2 sur monsiteweb-local	1 hour ago
alience.txt	APACHE 2.0	3 hours ago
index.html	premier commit sur index.html	1 hour ago

- ✓ L'objectif de cette phase : c'est lorsqu'il y'a deux changements sur le repo local et sur le repo GitHub, donc les démarches décrites c'est pour la gestion de ce conflit

Scénario 3

FIRST STEP: CHANGES ON GITHUB

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

	bootstrap-theme.css	committing website	2 days ago
	bootstrap-theme.min.css	committing website	2 days ago
	bootstrap.css	committing website	2 days ago
	bootstrap.min.css	committing website	2 days ago
	main.css	update main.css	22 hours ago
	style.css	Create style.css	2 days ago

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

Monsiteweb / initializr / index.html Cancel

< Edit file Preview changes Spaces 4 No wrap

```
21     <link rel="stylesheet" href="css/bootstrap-theme.min.css">
22     <link rel="stylesheet" href="css/main.css">
23
24     <script src="js/vendor/modernizr-2.8.3respond-1.4.2.min.js"></script>
25 </head>
26 <body>
27     <!--[if lt IE 8]>
28         <p class="browserupgrade">You are using an <strong>outdated</strong> browser. Please <a href="http://browsehappy.com/">upgrade</a>
29     <![endif]-->
30     <h1>Modification récente </h1>
31 <nav class="navbar navbar-inverse navbar-fixed-top" role="navigation">
32     <div class="container">
33         <div class="navbar-header">
34             <button type="button" class="navbar-toggle collapsed" data-toggle="collapse" data-target="#navbar" aria-expanded="false" aria-
35             <span class="sr-only">Toggle navigation</span>
36             <span class="icon-bar"></span>
37             <span class="icon-bar"></span>
38             <span class="icon-bar"></span>
39         </button>
40         <a class="navbar-brand" href="#">Project name</a>
41     </div>
42     <div id="navbar" class="navbar-collapse collapse collapse">
43         <form class="navbar-form navbar-right" role="form">
44             <div class="form-group">
45                 <input type="text" placeholder="Email" class="form-control">
46             </div>
47             <div class="form-group">
48                 <input type="password" placeholder="Password" class="form-control">
49             </div>
50             <button type="submit" class="btn btn-success">Sign in</button>
51         </form>
52     </div><!--/.navbar-collapse -->
53 </div>
54 </body>
55
56 <!-- Main jumbotron for a primary marketing message or call to action -->
57 <div class="jumbotron">
58     <div class="container">
```

- Renommer le fichier 404.html en error404.html

Monsiteweb / [error404.html](#) Cancel

<> Edit file Preview changes Spaces 4 No wrap

```
1 <!doctype html>
2 <html lang="en">
3 <head>
4   <meta charset="utf-8">
5   <title>Page Not Found</title>
6   <meta name="viewport" content="width=device-width, initial-scale=1">
7   <style>
8
9     * {
```

- Sur gitHub vérifiez la liste des commits .

Branch: master ▾

- ↳ Commits on Dec 12, 2019

Rename File 404.html to error404.html BlackDiamond-UX committed now	Verified		f96f498	
Premiere commit sur index.html BlackDiamond-UX committed 6 minutes ago	Verified		1f7b219	
creation du fichier style.css BlackDiamond-UX committed 14 minutes ago	Verified		bf26a00	

- **Sur GITBASH, vérifiez avec « git status »**

```
Youcode@DESKTOP-FIDNSP2 MINGW64 ~/Desktop/projects/monsiteweb-local (master)
$ git status
On branch master
Your branch is up to date with 'origin/master'.

nothing to commit, working tree clean
```

- **Visualisez et téléchargez les fichiers distants sur GITBASH**

```
$ 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.
```

- **Maintenant faites le pull et fusionnez les changements distants avec le repo local 5.**

```
$ git hist
* 5dcbb3c (origin/master, origin/HEAD) Rename 404.html to error404.html
* 8415c7b Update index.html
* aa7dd9f Create style.css
* 6b9d899 (HEAD -> master) Merge branch 'master' of https://github.com/
|\ 
| * a7db2f8 changed the title
| * 71417c9 more changes to readme.md
|/
* c60efa3 committing website
* 56f7a00 Initial commit
```

- **Vérifiez votre repo Local**
- **Sur votre repo Distant GITHUB, laissez un commentaire sur les fichiers que vous avez modifiiez en regardant la liste des commits.**

Rename File 404.html to error404.html

master

BlackDiamond-UX committed 40 minutes ago Verified

1 parent 1f7b219 commit f96f498b05094ee5e38ac47c0cba70e53fd9cb3

Showing 1 changed file with 0 additions and 0 deletions.

File renamed without changes.

1 comment on commit f96f498

BlackDiamond-UX commented on f96f498 now

modification le nom de cette fichier

- **Sur GITBASH affichez les informations du commit de l'ajout de la balise <h1></h1>**

```
$ git show
commit ff43f91e8085c5a9a4191696b506235fa77fc3c7 (HEAD -> master, origin/master, origin/HEAD)
Author: ech-mou <58331377-ech-mou@users.noreply.github.com>
Date:   Thu Dec 12 14:13:25 2019 +0100

    Update index.html

diff --git a/initializr/index.html b/initializr/index.html
index 8adde02..814b54a 100644
--- a/initializr/index.html
+++ b/initializr/index.html
@@ -27,6 +27,7 @@
<!--[if lt IE 8]>
    <p class="browserupgrade">You are using an <strong>outdated</strong> browser. Please <a href="http://bro
experience.</p>
<![endif]-->
+<h1> modification récente </h1>
<nav class="navbar navbar-inverse navbar-fixed-top" role="navigation">
    <div class="container">
        <div class="navbar-header">
```

SECOND STEP: BRANCHING AND MERGING SUR GITHUB

REPO DISTANT :

- **Sur GitHub créez une branche du nom 'Example'**

Branch: master ▾ Monsiteweb / initializr / i

Switch branches/tags

Find or create a branch...

Branches Tags

✓ master default

Example

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

toujours sur GITHUB

Example

BlackDiamond-UX committed 2 minutes ago Verified

1 parent f96f498 commit e17003dbd52ec64a7494f682bc26b0ee86a9c6cc

Showing 1 changed file with 1 addition and 0 deletions.

Unified Split

1 README.md

@@ -1 +1,2 @@
 1 New Titles
 2 + Ayoub et mouhssin

- **Sur github et sur le menu ‘branch’ ; Expliquez ce que vous voyez comme message ?**
Dans le menu des branches nous avons toutes les branches que nous avons et dans l’entrée, nous pouvons soit rechercher une branche, soit en créer une nouvelle

REPO LOCAL :

- **Sur votre repo local, créez une branche nommée ‘annulation’.**
- **Sur votre repo local éditez le fichier « index.html » et supprimez la balise H1.**

```
<!-- Main jumbotron for a primary marketing message or call to action -->
<div class="jumbotron">
  <div class="container">
    <h1>Hello, world!</h1>
    <p>This is a template for a simple marketing or informational website. It includes a large callout called a jumbotron and three supporting pieces of content. Use it as a starting point to create something more unique.</p>
    <p><a class="btn btn-primary btn-lg" href="#" role="button">Learn more &gt;</a></p>
  </div>
</div>
```

index.html [dos] (13:10 27/11/2019)
1 more line; before #1 14 seconds ago

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

```
Youcode@DESKTOP-FIDNSP2 MINGW64 ~/Desktop/projects/monsiteweb-local (annulation)
$ git commit -am "annulation de <h1>"
[annulation 1ae55dd] annulation de <h1>
1 file changed, 1 deletion(-)
```

- **« Git show » pour vérifier vos changements.**

```
Youcode@DESKTOP-FIDNSP2 MINGW64 ~/Desktop/projects/monsiteweb-local (annulation)
$ git show
commit 1a65ddd8d2b792352b1242c40ac4fa495e34dac (HEAD -> annulation)
Author: BlackDiamond-UX <rourha594@gmail.com>
Date:   Thu Dec 12 15:29:31 2019 +0100

    annulation de <h1>

diff --git a/initializr/index.html b/initializr/index.html
index 3209919..09d54e8 100644
--- a/initializr/index.html
+++ b/initializr/index.html
@@ -56,7 +56,6 @@ 
<!-- Main jumbotron for a primary marketing message or call to action -->
<div class="jumbotron">
  <div class="container">
-    <h1>Hello, world!

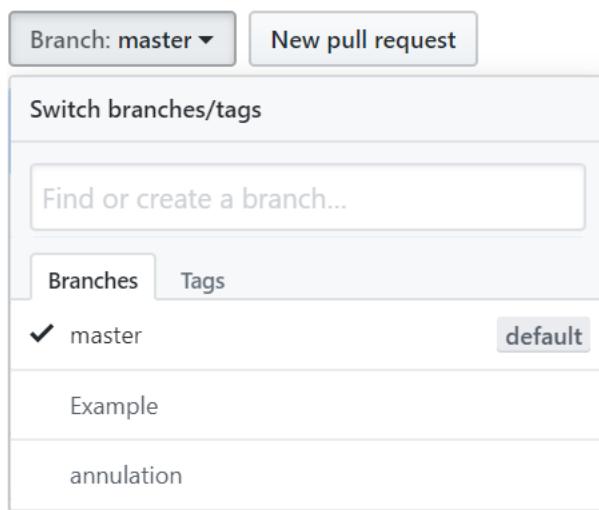
```

VÉRIFICATION :

- **Sur GITBASH Pushez les changements que vous avez sur la branche 'annulation' à votre repo distant**

```
Youcode@DESKTOP-FIDNSP2 MINGW64 ~/Desktop/projects/monsiteweb-local (annulation)
$ git push origin annulation
Enumerating objects: 7, done.
Counting objects: 100% (7/7), done.
Delta compression using up to 8 threads
Compressing objects: 100% (4/4), done.
Writing objects: 100% (4/4), 358 bytes | 358.00 KiB/s, done.
Total 4 (delta 3), reused 0 (delta 0)
remote: Resolving deltas: 100% (3/3), completed with 3 local objects.
remote:
remote: Create a pull request for 'annulation' on GitHub by visiting:
remote:     https://github.com/BlackDiamond-UX/Monsiteweb/pull/new/annulation
remote:
To https://github.com/BlackDiamond-UX/Monsiteweb.git
 * [new branch]      annulation -> annulation
```

- **Maintenant vérifiez votre repo distant.**



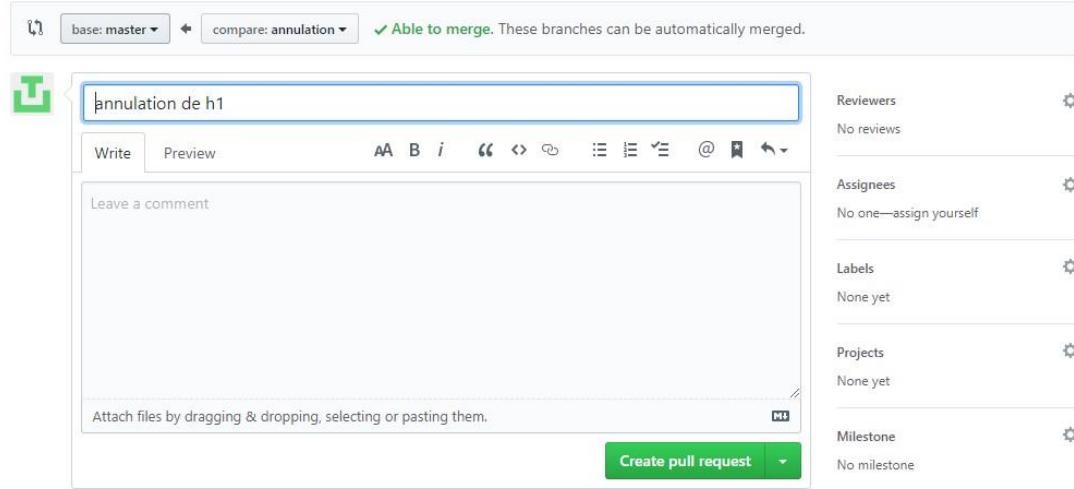
- **Qu'est-ce que vous constatez.**
- Nous avons apporté des modifications à la branche "annulation" et nous avons transféré notre nouvelle branche vers le dépôt distant. Il ne nous reste plus qu'à comparer dans GitHub et finaliser la demande de « pull »

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 ?**

Open a pull request

Create a new pull request by comparing changes across two branches. If you need to, you can also compare across forks.



Nous avons la possibilité de créer une nouvelle demande de tirage en comparant les changements sur deux branches.

- **Créez un pull request avec le commentaire ‘annulation à vérifier’**
- **Une page s’ouvre, explorez la page ? qu'est-ce que vous constatez ?**

annulation de <h1> #1

Open BlackDiamond-UX wants to merge 1 commit into `master` from `annulation`

Conversation 0 Commits 1 Checks 0 Files changed 1

BlackDiamond-UX commented now
annulation à vérifier

Owner + ...

annulation de <h1> 1ae55dd

Add more commits by pushing to the `annulation` branch on BlackDiamond-UX/Monsiteweb.

The screenshot shows a GitHub pull request interface. At the top, it says "BlackDiamond-UX wants to merge 1 commit into master from annulation". Below that are tabs for Conversation (0), Commits (1), Checks (0), and Files changed (1). A comment from "BlackDiamond-UX" is shown: "annulation à vérifier". Under the commits, there's a commit titled "annulation de <h1>" with hash "1ae55dd". Below the commit list, it says "Add more commits by pushing to the annulation branch on BlackDiamond-UX/Monsiteweb.". A green box highlights the CI status: "Continuous integration has not been set up. GitHub Actions and several other apps can be used to automatically catch bugs and enforce style." Another section says "This branch has no conflicts with the base branch. Merging can be performed automatically." At the bottom of the green box is a "Merge pull request" button and a note: "You can also open this in GitHub Desktop or view command line instructions."

GitHub nous dit que cette branche n'a aucun conflit avec la branche de base La fusion peut être effectuée automatiquement

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

The screenshot shows the GitHub conversation interface. At the top, there are tabs for "Write" and "Preview". The "Write" tab contains a text area with the placeholder "vous pouvez aussi laisser des commentaires à la personne qui demande le pull request". Below the text area is a note: "Attach files by dragging & dropping, selecting or pasting them." At the bottom are two buttons: "Close and comment" and "Comment".

- **Maintenant si tout va bien ; aucun message d'erreur effectuez le merging ?**
- **Supprimez la branche ‘annulation’ une fois tout est OK.**

```
Youcode@DESKTOP-FIDNSP2 MINGW64 ~/Desktop/projects/monsiteweb-local (master)
$ git push origin :annulation
To https://github.com/BlackDiamond-UX/Monsiteweb.git
 - [deleted]           annulation

Youcode@DESKTOP-FIDNSP2 MINGW64 ~/Desktop/projects/monsiteweb-local (master)
$ git branch -D annulation
Deleted branch annulation (was 190dfac).

Youcode@DESKTOP-FIDNSP2 MINGW64 ~/Desktop/projects/monsiteweb-local (master)
$ git branch
* master
```

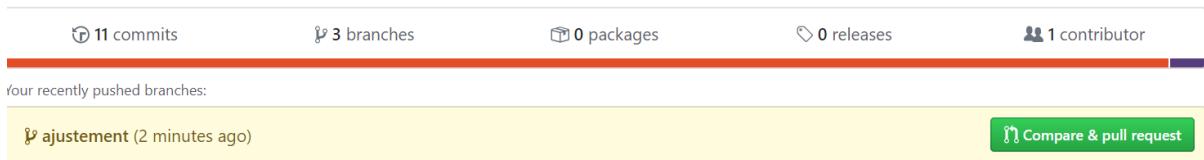
FOURTH STEP: MERGING EN LOCAL

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

```
> # git checkout –b ajustement
> #vim css/main.css
> #git commit –am “update main.css”
> #git push ajustement
```

➤

- **Sur votre GITHUB, un pull request est demandé !**



- **Maintenant, essayez de faire le merge en local ! @ vos mains !**

```
Youcode@DESKTOP-STUJ4PP MINGW64 ~/Desktop/projects/Monsiteweb-local (master)
$ git merge ajustement
Updating ff43f91..5bb7f1c
Fast-forward
 body      |  0
 css/main.css |  5 +++++
 {          |  0
 3 files changed, 5 insertions(+)
 create mode 100644 body
 create mode 100644 css/main.css
 create mode 100644 {
```

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

```
Youcode@DESKTOP-FIDNSP2 MINGW64 ~/Desktop/projects/monsiteweb-local (ajustement)
$ git branch
* ajustement
 master
```

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

Premier commit sur fichier main.css #3

Merged BlackDiamond-UX merged 1 commit into master from ajustement 23 minutes ago

Conversation 0 Commits 1 Checks 0 Files changed 2

BlackDiamond-UX commented 23 minutes ago

No description provided.

- **Sur GITBASH, revenez vers la branche master ! Expliquez le rendu de la commande ?**

- `# git checkout master`
- Ces commandes nous permettent de basculer entre les branches et dans cette situation, la branche vers laquelle nous voulons basculer est la branche principale .
- **Maintenant demandez le pull ?**

```
s git pull
Merge made by the 'recursive' strategy.
css/style.css      | 1 +
404.html => error404.html | 0
index.html         | 2 ++
3 files changed, 2 insertions(+), 1 deletion(-)
create mode 100644 css/style.css
rename 404.html => error404.html (100%)
```

- **Une fois la commande du pull est exécutée, à votre avis quel type de merge on aura besoin pour faire un merge sans conflits**
- besoin pour faire un merge sans conflits
- Fast-Forward Merge or automatique merge
- **Faites le merge à travers GITBASH**

- `# git merge`
- **Une fenêtre s'ouvre après l'exécution de la commande du merge ; ce qui signifie le message délivré ? à quoi sert cette fenêtre ?**
- La fenêtre qui s'est ouverte est simplement pour nous de commenter ou de décrire la fusion qui se produit
- **Git status**
- `# git status`

```
Youcode@DESKTOP-FIDNSP2 MINGW64 ~/Desktop/projects/monsiteweb-local/css (master)
$ git status
On branch master
Your branch is up to date with 'origin/master'.

nothing to commit, working tree clean
```

- **Si tout est OK ! Pushez les changements ?**

```
$ git push
git: 'credential-cache' is not a git command. See 'git --help'.
Enumerating objects: 7, done.
Counting objects: 100% (7/7), done.
Delta compression using up to 8 threads
Compressing objects: 100% (3/3), done.
Writing objects: 100% (3/3), 388 bytes | 194.00 KiB/s, done.
Total 3 (delta 2), reused 0 (delta 0)
remote: Resolving deltas: 100% (2/2), completed with 2 local objects.
To https://github.com/Ned-med/Monsiteweb.git
  2545998..67b0daf master -> master
```

- **Si tout est OK ! sur GITHUB le pull sera vérifier ? explorez le pull pour visualiser les changements ?**
- **Sur GITHUB supprimez les branches que vous avez créé.**

Your branches		
ajustement	Deleted just now by BlackDiamond-UX	<button>Restore</button>
Example	Deleted just now by BlackDiamond-UX	<button>Restore</button>

Active branches		
ajustement	Deleted just now by BlackDiamond-UX	<button>Restore</button>
Example	Deleted just now by BlackDiamond-UX	<button>Restore</button>

- **Sur le local on doit apporter les changements qu'on a fait sur GITHUB :**

- **Tapez la commande git branch -a ? expliquez le message ?**

```
Youcode@DESKTOP-FIDNSP2 MINGW64 ~/Desktop/projects/monsiteweb-local/css (master)
$ git branch -a
  ajustement
* master
  remotes/origin/Example
  remotes/origin/HEAD -> origin/master
  remotes/origin/ajustement
  remotes/origin/master
```

- Répertorie les branches de suivi à distance et les branches locales. Combiner avec -list pour

```
# git branch -a
```

- **Sur GITBASH supprimez les branches que vous avez créé.**

```
Youcode@DESKTOP-FIDNSP2 MINGW64 ~/Desktop/projects/monsiteweb-local/css (master)
$ git branch -d ajustement
Deleted branch ajustement (was fb9ba22).
```

- **Maintenant ! Tapez la commande git bash -a ? qu'est-ce que vous constatez ?**
- Maintenant nous voyons que la branche "ajustement" a été supprimée

- Ce que nous devons faire maintenant, est de faire appelez la suppression faites sur GITHUB et l'appliquez sur le local ! Trouvez la commande et exécutez ?**

```
# git push origin :<branch-name>
```

- Tapez la commande git bash -a ? qu'est-ce que vous constatez ?**
- Nous voyons que les branches ont été supprimées à la fois dans le dépôt distant et local

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**

2 contributors

201 lines (169 sloc) | 11.1 KB

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

Raw Blame History

- Sur GITBASH faites un pull global : git pull --all ; Maintenant faites le merge de la branche ‘updatelicence’.**

```
$ 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/Ned-med/Monsiteweb
 * [new branch]      updatelicence -> origin/updatelicence
Already up to date.
```

- Executez Git push !**

Branch: updatelicence * New pull request Create new file Upload files Find file Close or download ▾

This branch is even with abdellmum:updatelicence

Youcode@KIM MINGW64 ~/Desktop/projects/monsiteweb-local

```
$ git merge updatelicence
Updating 116d5c3..64e4061
Fast-forward
  licence.txt | 2 +-
  1 file changed, 1 insertion(+), 1 deletion(-)

Youcode@KIM MINGW64 ~/Desktop/projects/monsiteweb-local (master)
$ git push
Total 0 (delta 0), reused 0 (delta 0)
To https://github.com/abdellmum/monsiteweb.git
  116d5c3..64e4061  master -> master

Youcode@KIM MINGW64 ~/Desktop/projects/monsiteweb-local (master)
```

- **Git branch -d 'updatelicence'**

```
Youcode@DESKTOP-FIDNSP2 MINGW64 ~/Desktop/projects/monsiteweb-local (master)
$ git branch -d updatelicence
error: branch 'updatelicence' not found.
```

- **Exécutez la commande git bash -a ? qu'est-ce que vous constatez sur le remote et aussi sur GTHUB?**

```
# git branch -a
```

- la branche n'a pas été supprimée du dépôt distant, et elle n'existe pas dans le local
- **Maintenant sur GITBASH, trouvez la commande qui va nous permettre de supprimer la branche depuis le remote et ainsi de pushez les changements de la suppression simultanément ? Vérifiez vos Repo (local et distant)**

- `#git push origin :updatelicence`

```
Youcode@DESKTOP-FIDNSP2 MINGW64 ~/Desktop/projects/monsiteweb-local (master)
$ git push origin :updatelicence
To https://github.com/BlackDiamond-UX/Monsiteweb.git
 - [deleted]          updatelicence
```

- **Pourriez-vous expliquez 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**



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

```
<!--[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]><!-->
```

- **Et modifiez le titre du site : choisissez le titre que vous voulez**
- **Faites le staging et le commit en une seule ligne**
- `# git commit -am "updating index.html"`
`$ git commit -am "updating index.html"`
`[master 5649187] updating index.html`
 `1 file changed, 4 deletions(-)`
- **Maintenant apportez les changements faits sur le Repo distant : fetch**

- `# git fetch`

```
$ 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/Ned-med/Monsiteweb
  de8b938..80c77a7  master      -> origin/master
```

- **Git status ? lisez les commentaires**

```
$ 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**

```
$ git pull --rebase
First, rewinding head to replay your work on top of it...
Applying: updating 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 ?**

```
$ git hist
* 5dcbb3c (origin/master, origin/HEAD) Rename 404.html to error404.html
* 8415c7b Update index.html
* aa7dd9f Create style.css
* 6b9d899 (HEAD -> master) Merge branch 'master' of https://github.com/
  \ 
  | * a7db2f8 changed the title
  | * 71417c9 more changes to readme.md
  |
  * c60efa3 committing website
* 56f7a00 Initial commit
```

- **Vérifiez que tous les changements que vous avez faits sont OK ?**

- Après avoir exécuté la commande --rebase, nous voyons que le commit que nous avons fait au local vient après celui que nous avons fait dans github

- **Expliquez l'utilité du REBASE ?**

- 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 aplatis l'historique, car il transfère le travail terminé d'une branche à une autre. Dans le processus, l'historique non désiré est éliminé.

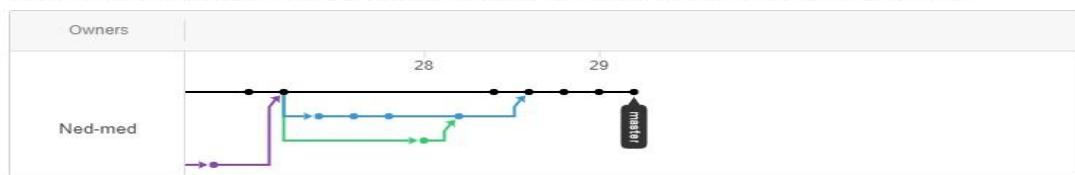
- Les "rebasements" indiquent comment les modifications doivent être transférées du sommet de la hiérarchie vers le bas, et les fusions indiquent comment elles remontent vers le haut.

SEVENTH STEP : GitHub Insights

- **Analysez le Network ou le timeline de votre workflow ?**

Network graph

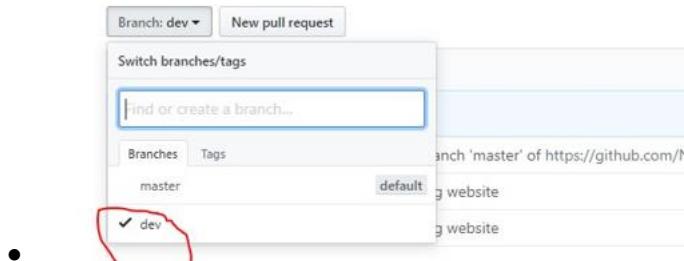
Timeline of the most recent commits to this repository and its network ordered by most recently pushed to.



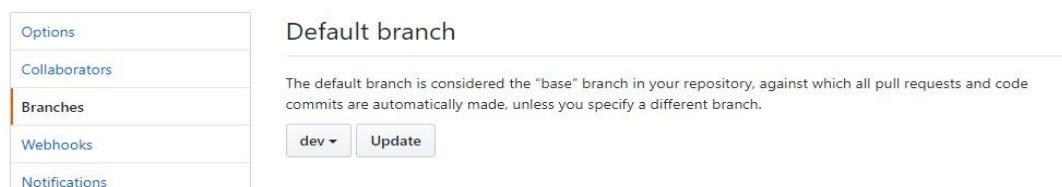
- **Discutez entre vous équipe et formalisez ce que vous voyez sur le Network en une production Timeline sur Tableau à marqueur ou trello : imaginez aussi que chaque membre du groupe à contribuer dans une tache du Timeline Network (Like a Sprint planning) .**

EIGHTH STEP: DEFAULT BRANCH AND CONFLICTS

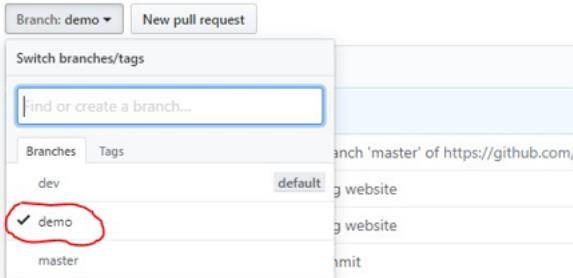
- **Sur github créez une branche du nom “Dev”**



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



- **Revenez à la page d'accueil de votre repo ? à votre avis on a besoin dans ce cas de faire un merge ou bien un rebase ? pourquoi ?**
- Dans ce cas on a besoin de faire un rebase car ce dernière compresse tous les changement sur une seule patch et après integrer le dans le branch destinataire et de plus il élimine l'historique non désiré
- **Ne faites aucun merge ni Rebase mais plutôt supprimez la branche feature depuis GITHUB**

Your branches

Branch	Last Update	Pull Requests	Actions
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

Branch	Last Update	Pull Requests	Actions
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 website**

```
$ rm -rf Monsiteweb-local/
```

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

```
git clone https://github.com/Ned-med/Monsiteweb.git
Cloning into 'Monsiteweb'...
remote: Enumerating objects: 70, done.
remote: Counting objects: 100% (70/70), done.
remote: Compressing objects: 100% (54/54), done.
remote: Total 70 (delta 27), reused 45 (delta 13), pack-reused 0
Unpacking objects: 100% (70/70), done.
```

- **Vérifiez si le contenu est récupéré**

```
$ ls
apple-touch-icon.png  css/      favicon.ico  index.html  LICENSE  tile.png
browserconfig.xml    error404.html  fonts/     js/        readme.md  tile-wide.png
```

- **Vérifiez les branches et où pointe le pointeur : HEAD**

```
$ git hist
* 80c77a7 [HEAD -> dev] origin/master, origin/dev, origin/HEAD) adding update to readme file
* de8b938 Update From V2.0 to V3.0
* 299dcab update apache version to 3.0
* 67b0daf Merge branch 'master' of https://github.com/Ned-med/Monsiteweb
| \
| * 2545998 Merge pull request #1 from Ned-med/annulation
| |
| * 73d3187 annulation de h1
| * 5dcbb3c Rename 404.html to error404.html
| * 8415c7b Update index.html
| * aa7dd9f Create style.css
|
| * 7a4808d update main.css
|
* 6b9d899 Merge branch 'master' of https://github.com/Ned-med/Monsiteweb
| \
| * a7db2f8 changed the title
* | 71417c9 more changes to readme.md
| /
* c60efa3 committing website
* 56f7a00 Initial commit
```

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

```
$ git checkout master
Switched to a new branch 'master'
Branch 'master' set up to track remote branch 'master' from 'origin'.
```

#Quick Challenge: Try and Catch ()

On va imaginer le mini Scénario suivant :

2 contributors

3 lines (3 sloc) | 60 Bytes

Raw Blame History

#Made More Changes made more changes to readme file Updates

2. Localement on est sur la branche DEV

3. on a faist des changement sur le REPO distant surement on a besoin d'un fetch

```
#git 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/Ned-med/Monsiteweb
  80c77a7..9481b38  dev      -> origin/dev
```

4. 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.

```
#echo "##my local changes to readme" >> readme.md
```

```
$ echo "##my local changes to readme" >> readme.md
```

```
$ git status
On branch dev
Your branch is behind 'origin/dev' 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")
```

5. On va faire le commit

```
#git commit -am "adding one line to readme"
$ git commit -am "adding one line to readme"
warning: LF will be replaced by CRLF in readme.md.
The file will have its original line endings in your working directory
[dev 65d6d76] adding one line to readme
 1 file changed, 1 insertion(+)
```

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

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

7. 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

Nous n'avons pas apporté de modifications sur la même ligne du fichier readme.md, c'est pourquoi au lieu d'avoir un conflit, nous avons eu une fusion automatique avec une stratégie récursive.

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

Conclusion

En conclusion, Git est un outil performant et simple d'utilisation, qui facilitera grandement le travail en équipe au sein de votre groupe durant le développement de votre site web. Les principales difficultés liées à l'utilisation

de Git viennent de la résolution des éventuels conflits entre vos modifications. Ainsi que le site GitHub permet de se servir plus facilement des fonctionnalités de git.

Ce Brief nous a permis de mieux comprendre le flux de travail de git et de GitHub. Sur le chemin de la réalisation des scénarios, nous avons rencontré quelques problèmes mineurs, mais nous les avons surmontés en utilisant des documents git et en nous expliquant les choses les uns aux autres.