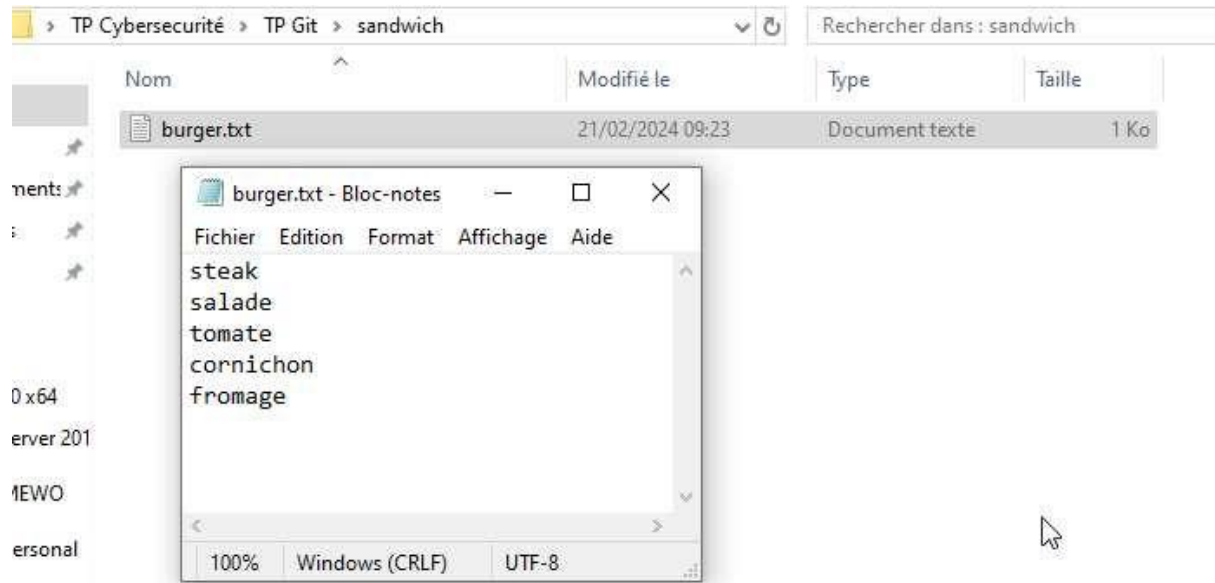


TP 4 – Git

2.1)

Création d'un nouveau dépôt Git nommé "**sandwich**" avec un fichier "**burger.txt**" contenant les ingrédients d'un burger.



2.2)

On vérifie l'état du dépôt avec "**git status**", afin de s'assurer que les modifications sont seulement dans la copie de travail.

```
Maxime QUESSADA@DESKTOP-NAHOR16 MINGW64 ~/OneDrive - MEWO/Bureau/TP Cybersecrété/TP Git/sandwich (master)
$ git status
On branch master

No commits yet

Untracked files:
  (use "git add <file>..." to include in what will be committed)
    burger.txt

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

2.3)

On prépare “**burger.txt**” pour le commit avec “**git add burger.txt**” puis on vérifie que les modifications ont été placées dans l’index avec “**git status**”, puis on observe les différences entre l’index et le master avec “**git diff --cached**”.

```
Maxime QUESSADA@DESKTOP-NAHOR16 MINGW64 ~/OneDrive
)
$ git add burger.txt

Maxime QUESSADA@DESKTOP-NAHOR16 MINGW64 ~/OneDrive
)
$ git status
On branch master

No commits yet

Changes to be committed:
  (use "git rm --cached <file>..." to unstage)
    new file:   burger.txt
```

```
Maxime QUESSADA@DESKTOP-NAHOR16 MINGW64 ~
)
$ git diff --cached
diff --git a/burger.txt b/burger.txt
new file mode 100644
index 0000000..d106d82
--- /dev/null
+++ b/burger.txt
@@ -0,0 +1,5 @@
+steak
+salade
+tomate
+cornichon
+fromage
\ No newline at end of file
```

2.4)

On commit les modifications avec “**git commit -m “message de commit”**”.

```
Maxime QUESSADA@DESKTOP-NAHOR16 MINGW64 ~
)
$ git commit -m "burger"
[master (root-commit) dd47ac4] burger
1 file changed, 5 insertions(+)
create mode 100644 burger.txt
```

2.5)

On vérifie que les modifications ont été correctement commit avec “**git status**”.

```
Maxime QUESSADA@DESKTOP-NAHOR16 MINGW64 ~/  
)  
$ git status  
On branch master  
nothing to commit, working tree clean
```

2.6)

On affiche l'historique des commit avec “**git log**”. On voit alors que le dernier commit a l'id suivant : “**dd47ac49101d77103c8569c69f8320d7bb0f5570**”.

```
Maxime QUESSADA@DESKTOP-NAHOR16 MINGW64 ~/OneDrive - MEWO/Bureau/TP  
)  
$ git log  
commit dd47ac49101d77103c8569c69f8320d7bb0f5570 (HEAD -> master)  
Author: Maxime Quessada <maxime.quessada@mewo-campus.fr>  
Date: Wed Feb 21 09:57:11 2024 +0100  
  
burger
```

2.7)

On crée et modifie au moins 5 sandwiches et on commit chaque modification séparément pour avoir 5 commits.

```
MINGW64:/c/Users/Maxime QUESSADA/OneDrive - MEWO/Bureau/TP Cybersecurit /TP Git/sandwich
Maxime QUESSADA@DESKTOP-NAHOR16 MINGW64 ~/OneDrive - MEWO/Bureau/TP Cybersecurit /TP Git/sandwich (master)
$ git diff --cached
diff --git a/hot_dog.txt b/hot_dog.txt
new file mode 100644
index 0000000..1ece8c6
--- /dev/null
+++ b/hot_dog.txt
@@ -0,0 +1,3 @@
+saucisse
+ketchup
+moutarde
\ No newline at end of file

Maxime QUESSADA@DESKTOP-NAHOR16 MINGW64 ~/OneDrive - MEWO/Bureau/TP Cybersecurit /TP Git/sandwich (master)
$ git commit
Aborting commit due to empty commit message.

Maxime QUESSADA@DESKTOP-NAHOR16 MINGW64 ~/OneDrive - MEWO/Bureau/TP Cybersecurit /TP Git/sandwich (master)
$ git commit -m "hotdog"
[master 5cf5f91] hotdog
1 file changed, 3 insertions(+)
create mode 100644 hot_dog.txt
```

MINGW64:/c/Users/Maxime QUESSADA/OneDrive - MEWO/Bureau/TP Cybersecurit /TP Git/sandwich

```
Maxime QUESSADA@DESKTOP-NAHOR16 MINGW64 ~/OneDrive - MEWO/Bureau/TP Cybersecurit /TP Git/sandwich (master)
$ git log
commit 7fc793fd9b7ad5fffb281494cee567629c6dbc17 (HEAD -> master)
Author: Maxime Quessada <maxime.quessada@mewo-campus.fr>
Date: Wed Feb 21 10:32:42 2024 +0100

    tacos

commit 8c1f0c630724871066360556431b098477cd450b
Author: Maxime Quessada <maxime.quessada@mewo-campus.fr>
Date: Wed Feb 21 10:31:28 2024 +0100

    kebab

commit 56644c972abd621ca88db23d6b983bfd15979143
Author: Maxime Quessada <maxime.quessada@mewo-campus.fr>
Date: Wed Feb 21 10:30:56 2024 +0100

    poulet_mayo

commit 4b235165e81ff4ce6b4faaac049e773b9e000481
Author: Maxime Quessada <maxime.quessada@mewo-campus.fr>
Date: Wed Feb 21 10:30:30 2024 +0100

    jambon_beurre

commit 5cf5f91d0f3d220d397b8a0a647bdd625e20f0fb
Author: Maxime Quessada <maxime.quessada@mewo-campus.fr>
Date: Wed Feb 21 10:29:04 2024 +0100

    hotdog

commit dd47ac49101d77103c8569c69f8320d7bb0f5570
Author: Maxime Quessada <maxime.quessada@mewo-campus.fr>
Date: Wed Feb 21 09:57:11 2024 +0100

    burger
```

2.8)

On vérifie l'historique des commits avec “**git log**” et on confirme que tout a été commit avec “**git status**”. On utilise également des logiciels comme gitg et gitk afin de voir un historique des commits de façon graphique/visuelle.

```
Maxime QUESSADA@DESKTOP-NAHOR16 MINGW64
$ git status
On branch master
nothing to commit, working tree clean
```

```
MINGW64:/c/Users/Maxime QUESSADA/OneDrive - MEWO/Bureau/TP Cybersecurit /TP Git/sandwich
Maxime QUESSADA@DESKTOP-NAHOR16 MINGW64 ~/OneDrive - MEWO/Bureau/TP Cybersecurit /TP Git/sandwich (master)
$ git log --graph --pretty=short
* commit 7fc793fd9b7ad5fffb281494cee567629c6dbc17 (HEAD -> master)
  Author: Maxime Quessada <maxime.quessada@mewo-campus.fr>

  tacos

* commit 8c1f0c630724871066360556431b098477cd450b
  Author: Maxime Quessada <maxime.quessada@mewo-campus.fr>

  kebab

* commit 56644c972abd621ca88db23d6b983bfd15979143
  Author: Maxime Quessada <maxime.quessada@mewo-campus.fr>

  poulet_mayo

* commit 4b235165e81ff4ce6b4faaac049e773b9e000481
  Author: Maxime Quessada <maxime.quessada@mewo-campus.fr>

  jambon_beurre

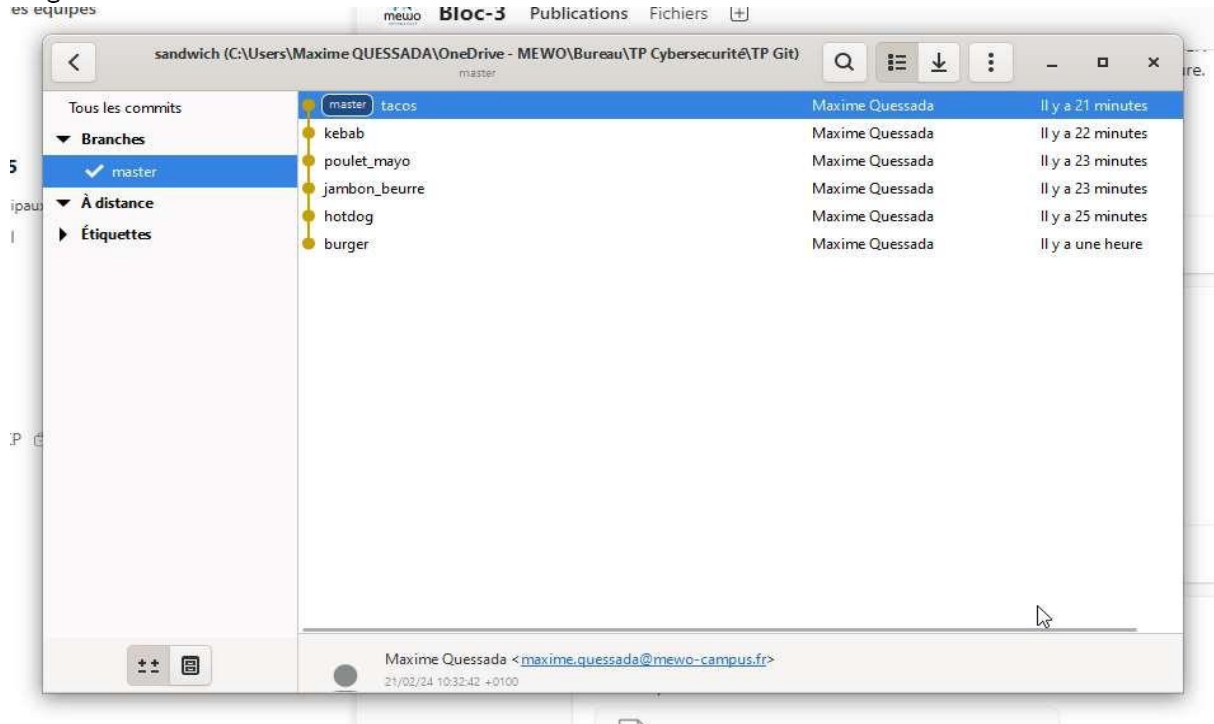
* commit 5cf5f91d0f3d220d397b8a0a647bdd625e20f0fb
  Author: Maxime Quessada <maxime.quessada@mewo-campus.fr>

  hotdog

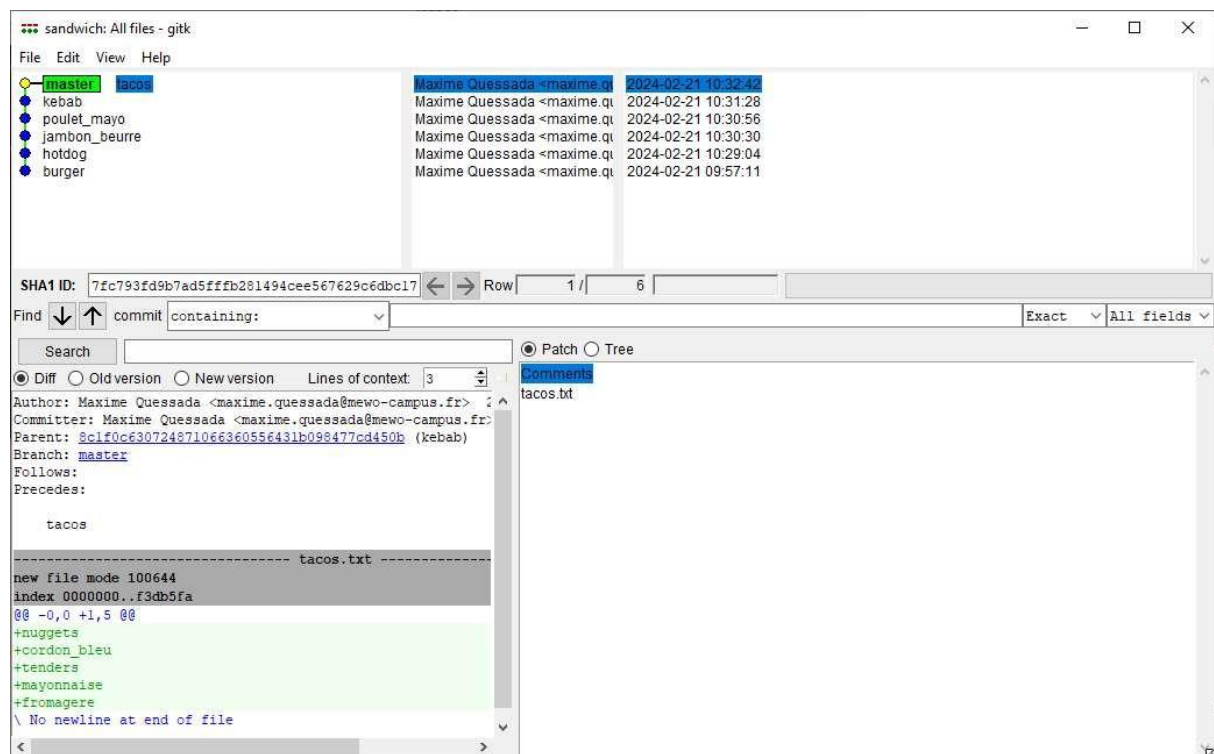
* commit dd47ac49101d77103c8569c69f8320d7bb0f5570
  Author: Maxime Quessada <maxime.quessada@mewo-campus.fr>

  burger
```

Gitg:



Gitk:



2.9)

On ajoute burger au commit avec “**git add burger.txt**”, puis on l’enlève grâce à “**git reset burger.txt**”. On vérifie entre chaque étape que tout s’est bien déroulé avec “**git status**”.

```
Maxime QUESSADA@DESKTOP-NAHOR16 MINGW64 ~/OneDrive - MEWO/Bureau/TP Cybersecurit  /TP Git/sandwich (master)
$ git add burger.txt

Maxime QUESSADA@DESKTOP-NAHOR16 MINGW64 ~/OneDrive - MEWO/Bureau/TP Cybersecurit  /TP Git/sandwich (master)
$ git status
On branch master
Changes to be committed:
  (use "git restore --staged <file>..." to unstage)
        modified:   burger.txt

Maxime QUESSADA@DESKTOP-NAHOR16 MINGW64 ~/OneDrive - MEWO/Bureau/TP Cybersecurit  /TP Git/sandwich (master)
$ git reset burger.txt
Unstaged changes after reset:
M       burger.txt

Maxime QUESSADA@DESKTOP-NAHOR16 MINGW64 ~/OneDrive - MEWO/Bureau/TP Cybersecurit  /TP Git/sandwich (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:   burger.txt

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

2.10)

On utilise la commande “**git checkout burger.txt**” qui enl  ve les modifications de “**burger.txt**” de l’index. Il n’appara  t alors plus dans les modifications    commit, ou    ajouter.

```
Maxime QUESSADA@DESKTOP-NAHOR16 MINGW64   /TP Git/sandwich (master)
$ git checkout burger.txt
Updated 1 path from the index

Maxime QUESSADA@DESKTOP-NAHOR16 MINGW64   /TP Git/sandwich (master)
$ git status
On branch master
nothing to commit, working tree clean
```

2.11)

On utilise “**git checkout ID_COMMIT**” afin de déplacer le **HEAD** (index actuel) vers un commit précédent. Toutes les modifications passées ce commit sont annulées. On vérifie ensuite le status avec “**git status**”.

```
Maxime QUESSADA@DESKTOP-NAHOR16 MINGW64 ~/OneDrive - MEWO/Bureau/TP Cybersecurité/TP Git/sandwich (master)
$ git checkout 5cf5f91d0f3d220d397b8a0a647bdd625e20f0fb
Note: switching to '5cf5f91d0f3d220d397b8a0a647bdd625e20f0fb'.

You are in 'detached HEAD' state. You can look around, make experimental
changes and commit them, and you can discard any commits you make in this
state without impacting any branches by switching back to a branch.

If you want to create a new branch to retain commits you create, you may
do so (now or later) by using -c with the switch command. Example:

    git switch -c <new-branch-name>

Or undo this operation with:

    git switch -

Turn off this advice by setting config variable advice.detachedHead to false

HEAD is now at 5cf5f91 hotdog
```

```
Maxime QUESSADA@DESKTOP-NAHOR16 MINGW64
$ git status
HEAD detached at 5cf5f91
nothing to commit, working tree clean
```

2.12)

Enfin, on revient à la dernière commit avec “**git checkout master**” et on récupère alors les modifications de tout le dépôt Git.

```
Maxime QUESSADA@DESKTOP-NAHOR16 MINGW64 ~/OneDrive
$ git checkout master
Previous HEAD position was 5cf5f91 hotdog
Switched to branch 'master'
```

```
Maxime QUESSADA@DESKTOP-NAHOR16 MINGW64
$ git status
On branch master
nothing to commit, working tree clean
```