**Publicis Sapient Week1 – Assignment**

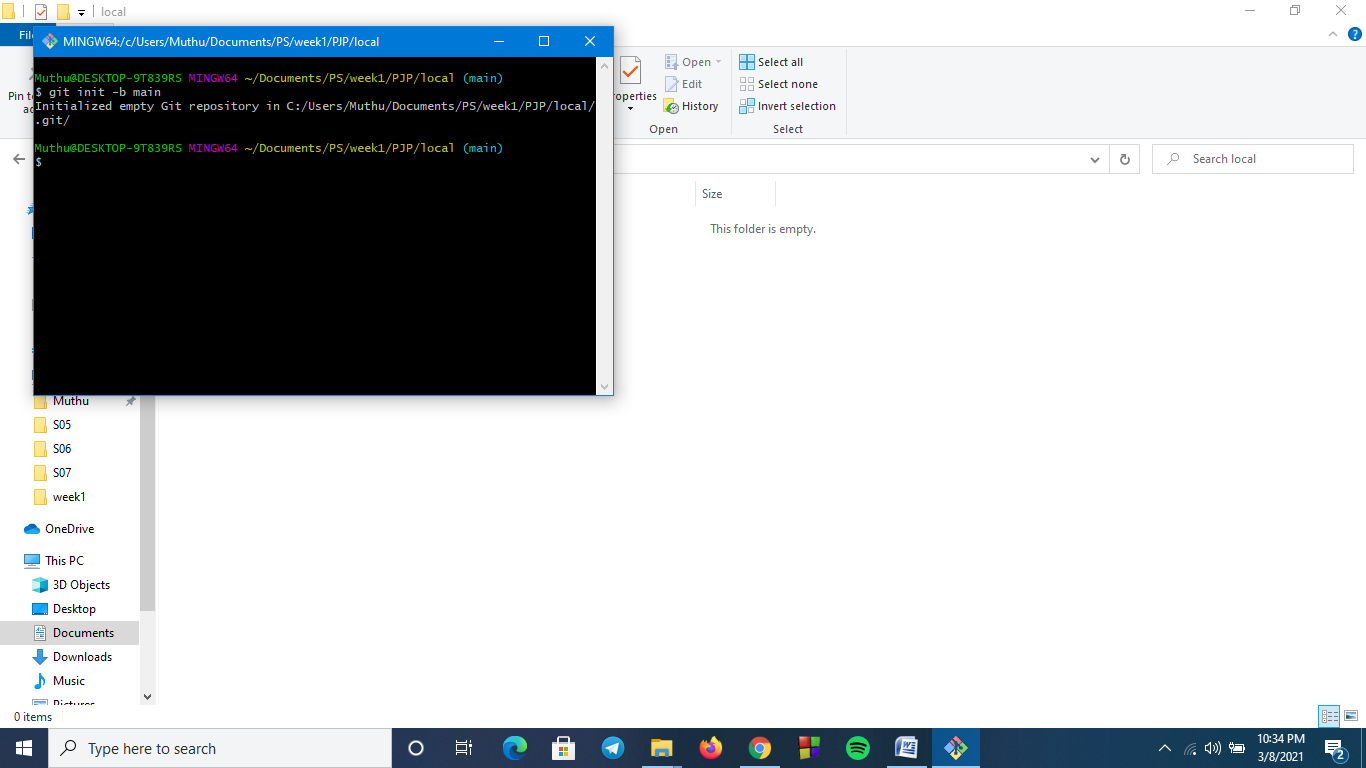
Mutharasan E

Anna University,CEG

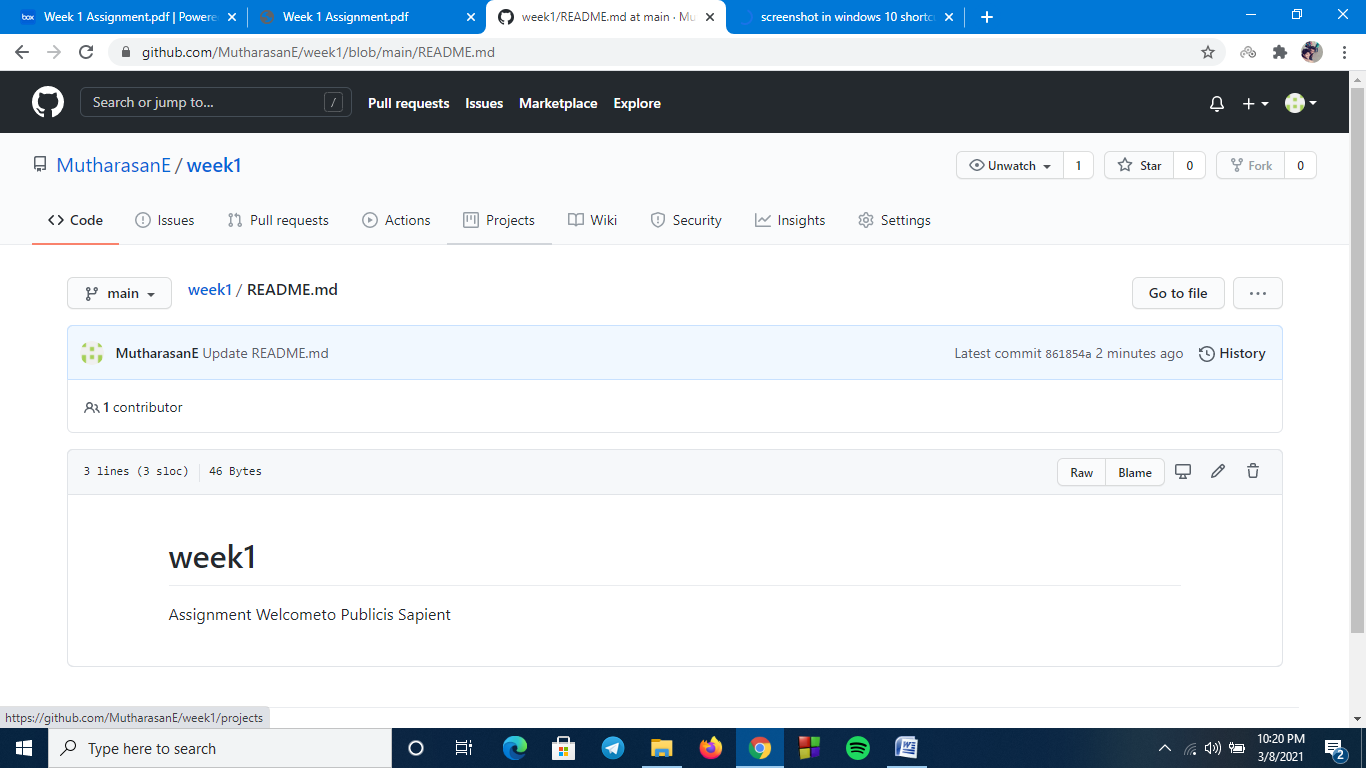
1.a.

i.Local repository:

$git init -b main



ii.Remote Repository:



iii.local branches:

$git branch dev

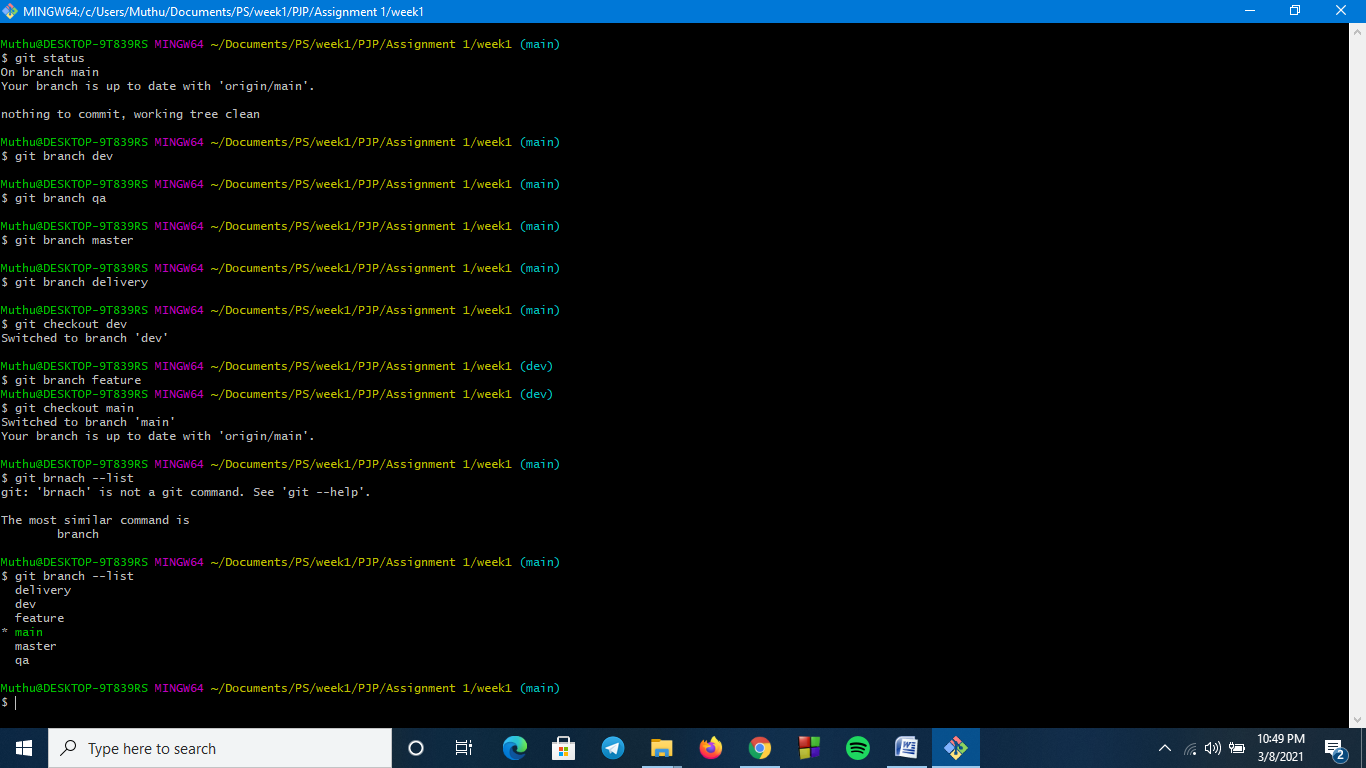
$git branch feature

$git branch qa

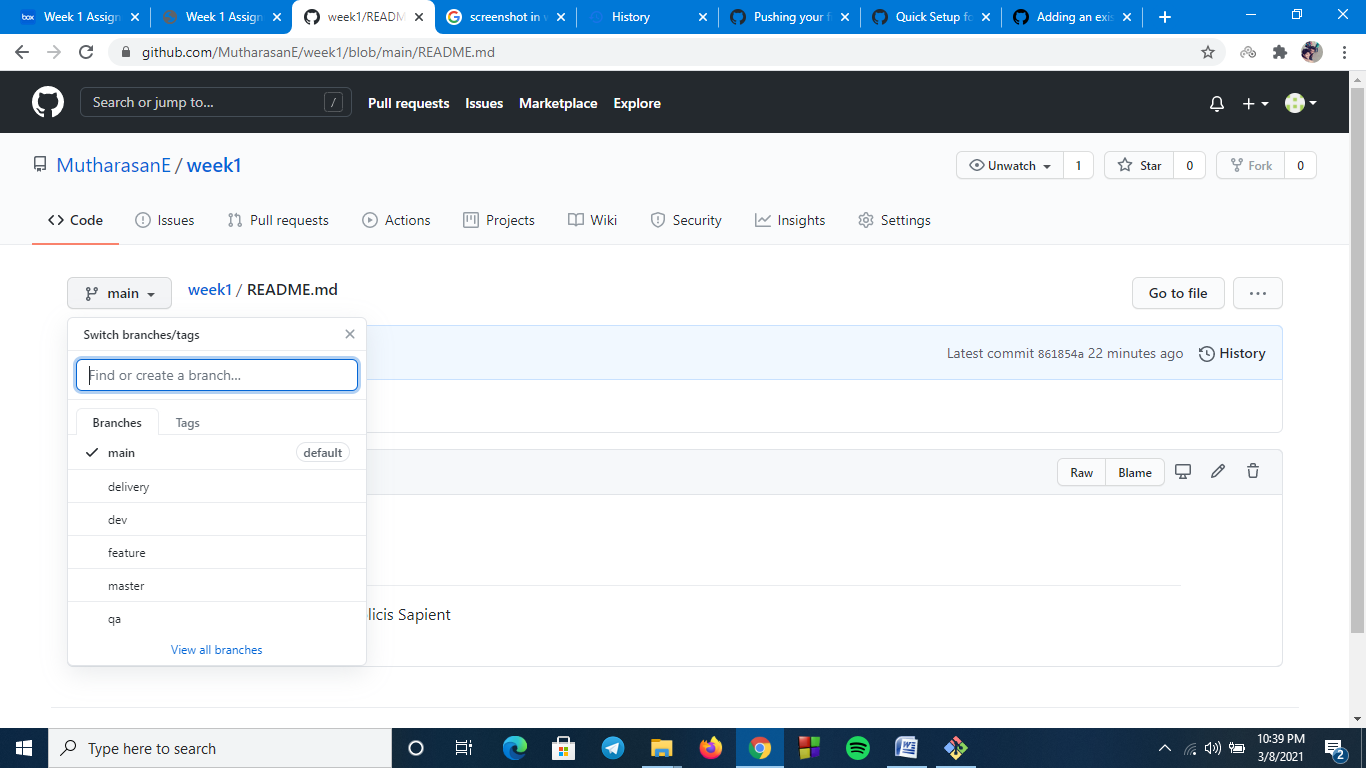
$git branch master

$git branch delivery

$git branch –list



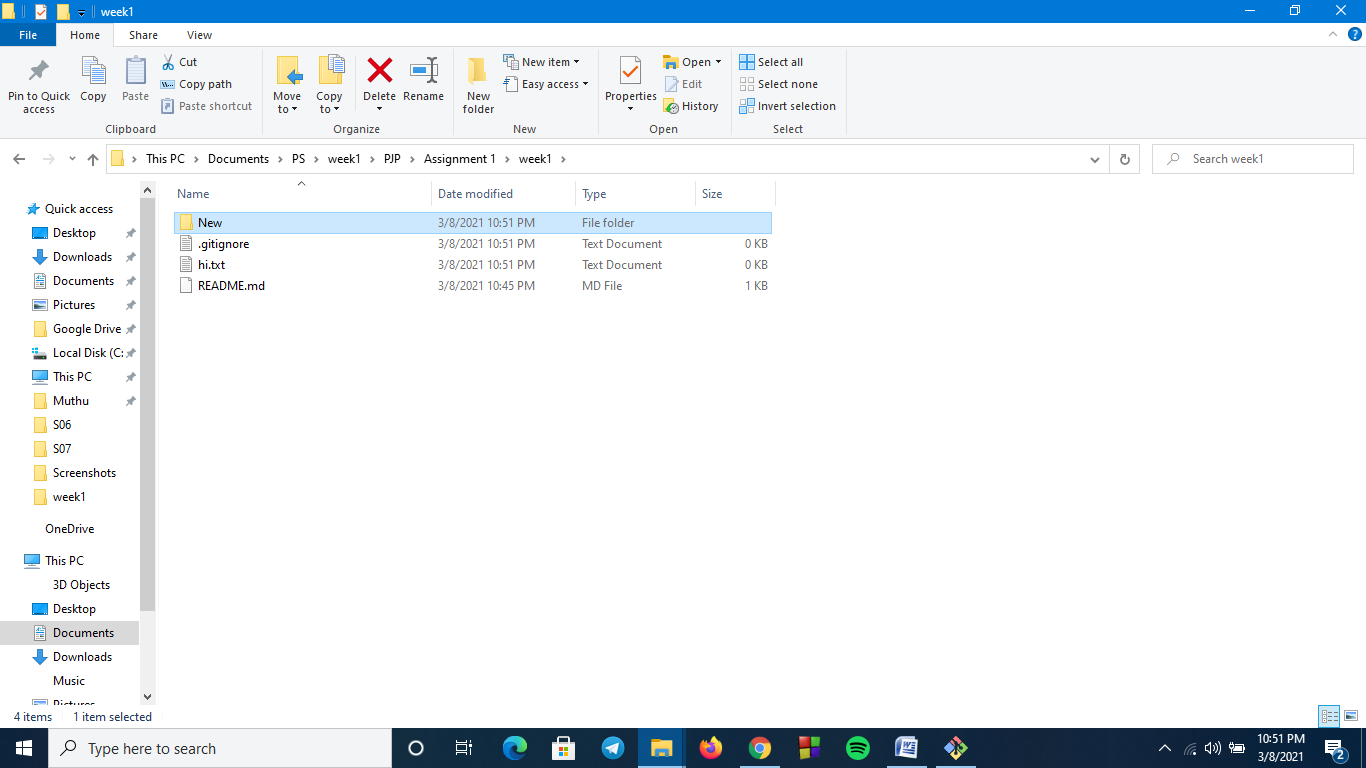
iv.remote branches:



v.add files add folders:

Folder-new

Files - .gitignore,hi.txt



vi.Check-in ,Stage, Commit,Push into feature:

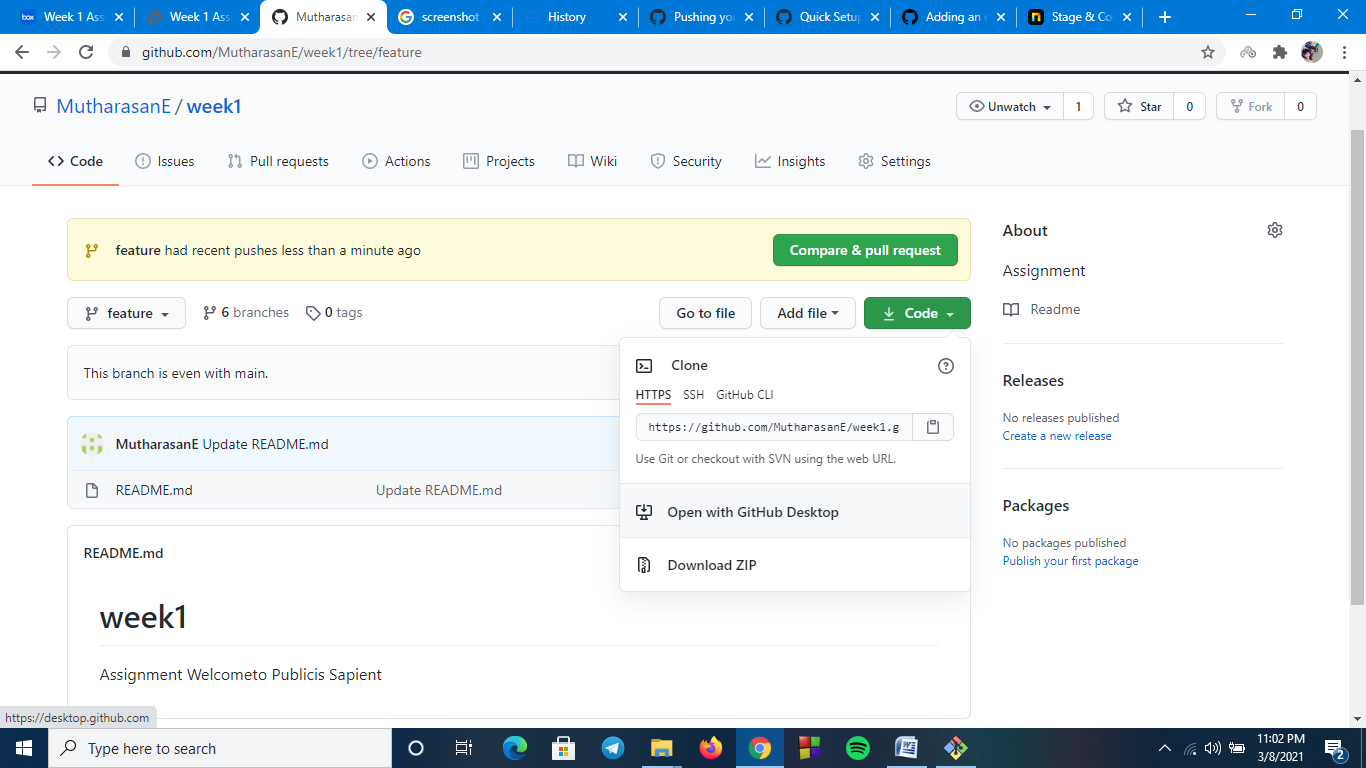
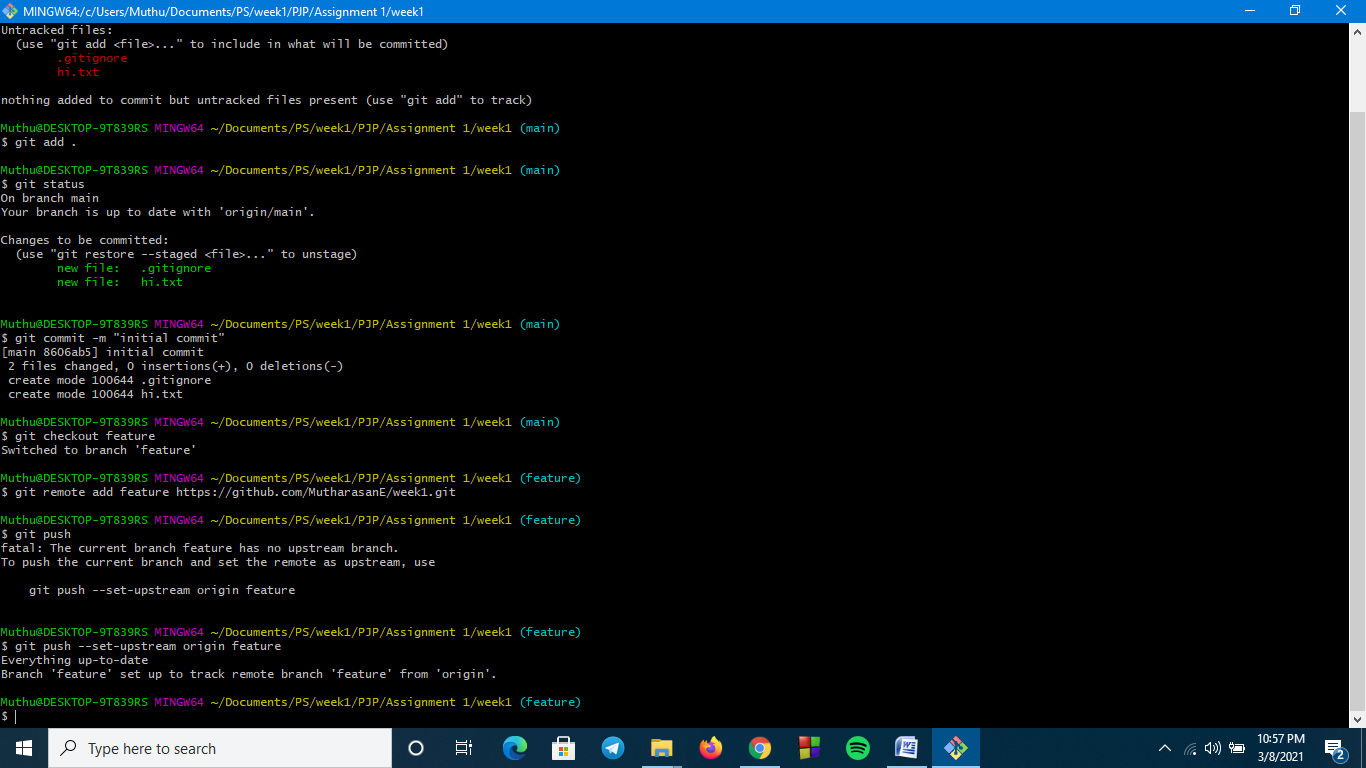
$git status

$git add .

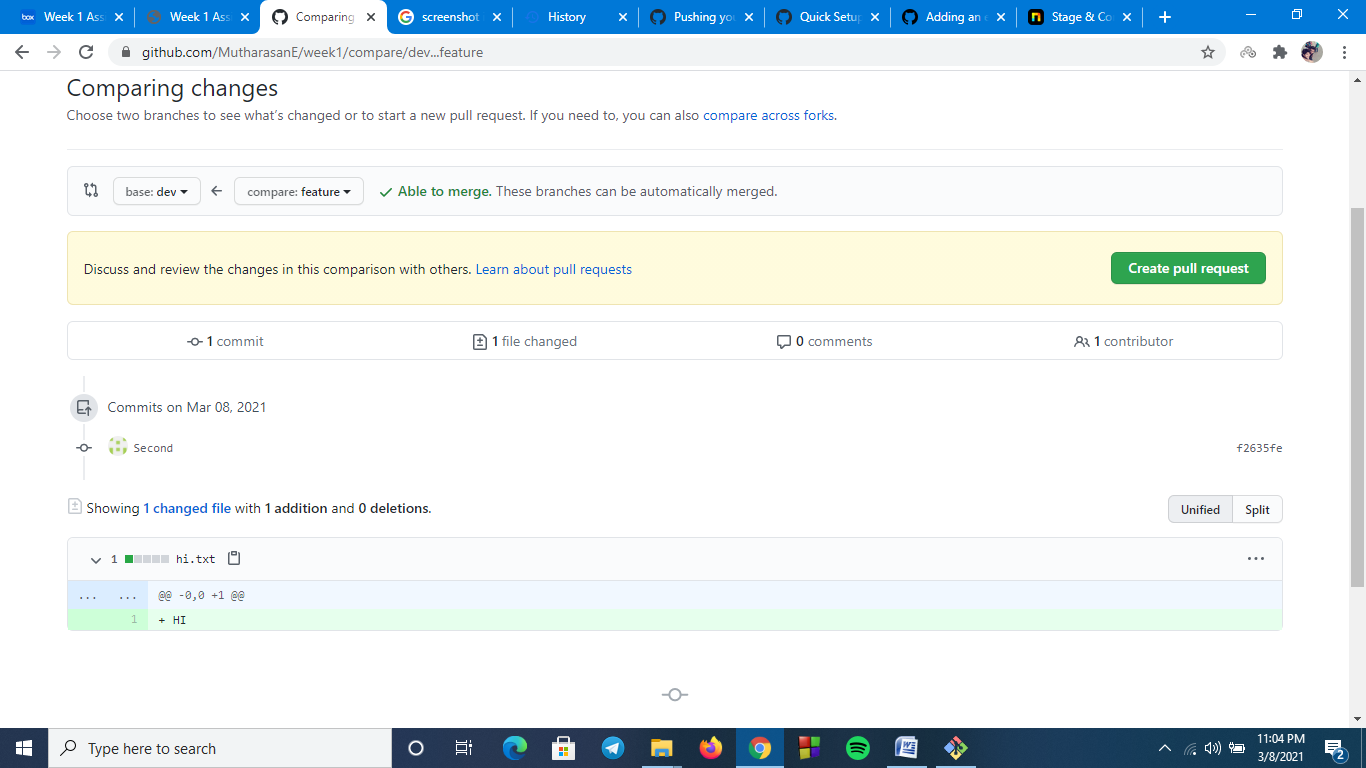
$git commit –m “Initial commit”

$git remote add feature <https://github.com/MutharasanE/week1.git>

$git push



vii.Promote code from feature to dev:



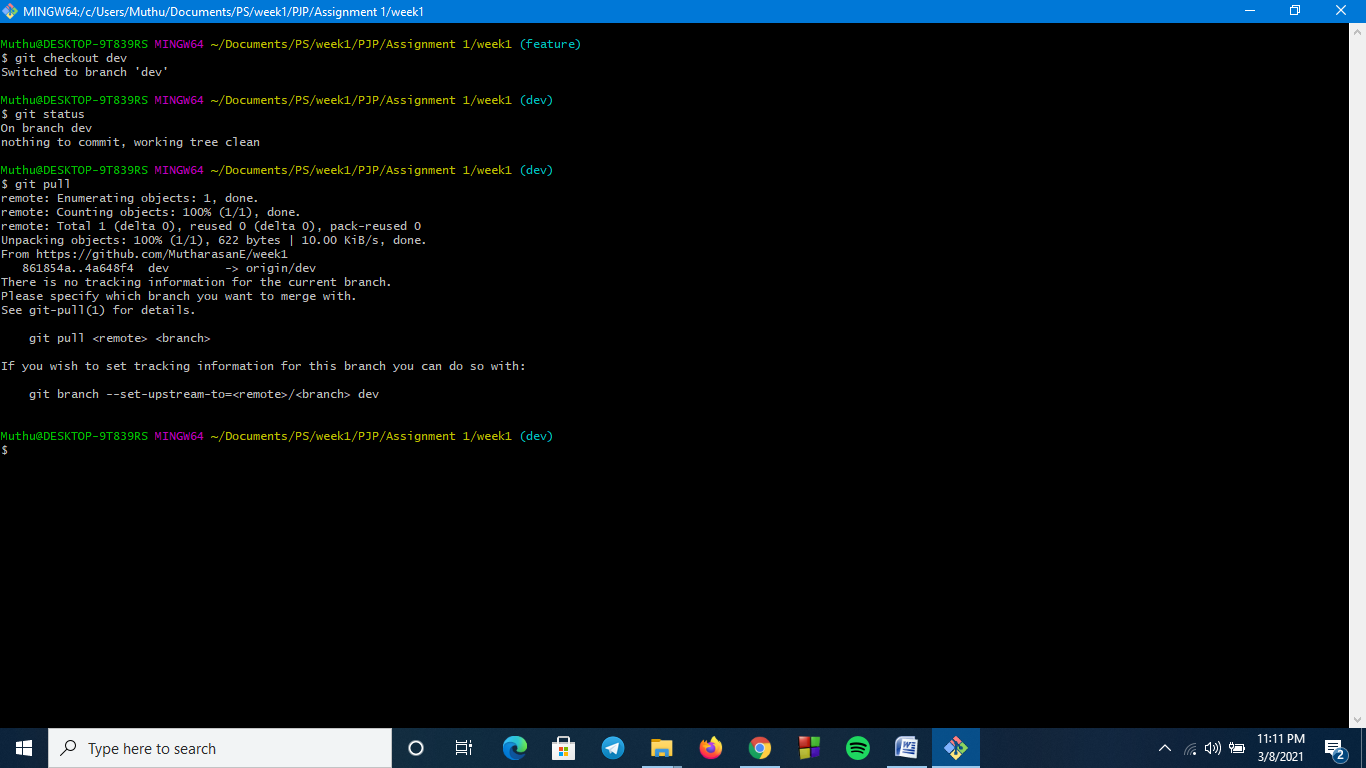
viii.Latest code from remote to local:

In remote repository , We created a pull request from feature to dev but it is not updated in local repository.

$git checkout dev

$git pull

Now pull updated the local dev branch from remote repository.



ix.Checkout Vs Pull:

Checkout:Check for the update in a branch but does not merge the updates.

Pull:Check for the updates and merges in local branch.

x.Handle Merge Conflict:

I created merge conflict by adding same line in two branches in

Branch QA and branch delivery

* In branch qa I created a text document “new.txt” and added first two lines as hi,hello.
* Then I created a commit in this branch
* Then I checkedout branch delivery and changed first two lines as muthu,anbu,
* I created a commit in delivery branch also.
* I merged delivery branch in main branch
* After that when I try to merge qa branch to main “It shows merge conflict error”.

Command:

$git checkout qa

Change new.txt

$git add .

$git commit –m “greet”

$git checkout delivery

Change new.txt

$git add .

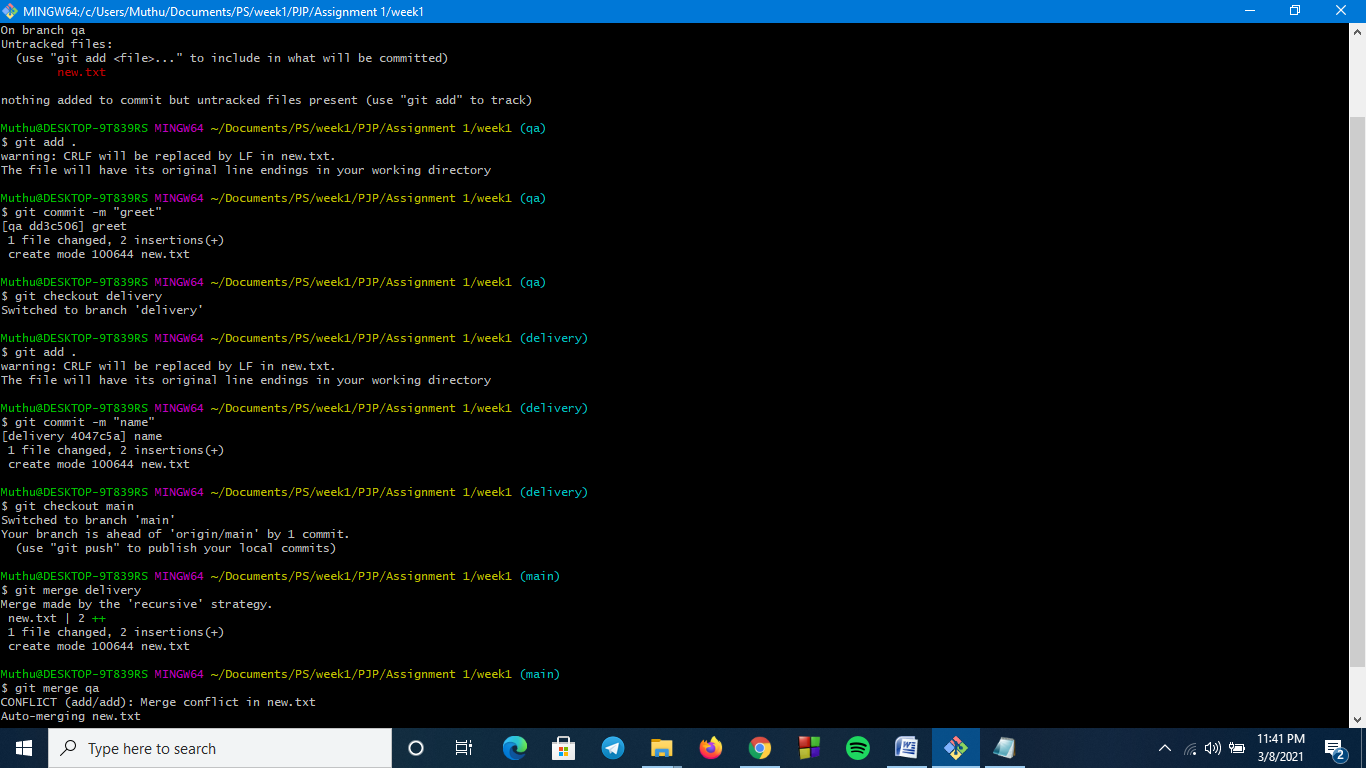
$git commit –m “name”

$git checkout main

$ git merge delivery –m “merge”

$git merge qa –m “merge1”

Merge Conflict occurs here

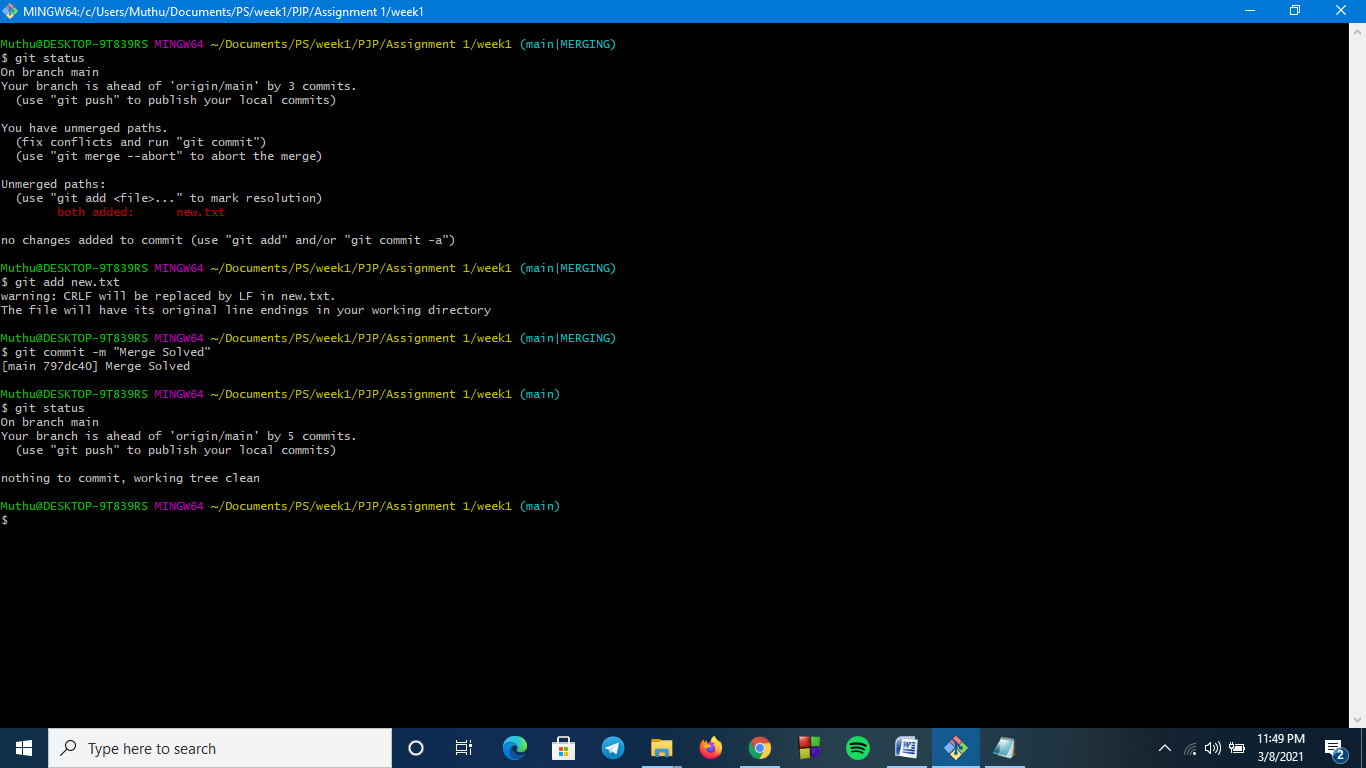


Handling merge Conflicts:

* I added both lines in next.txt muthu,anbu,hi,hello.
* Then added the new.txt
* And marked the merge conflict as resolved.

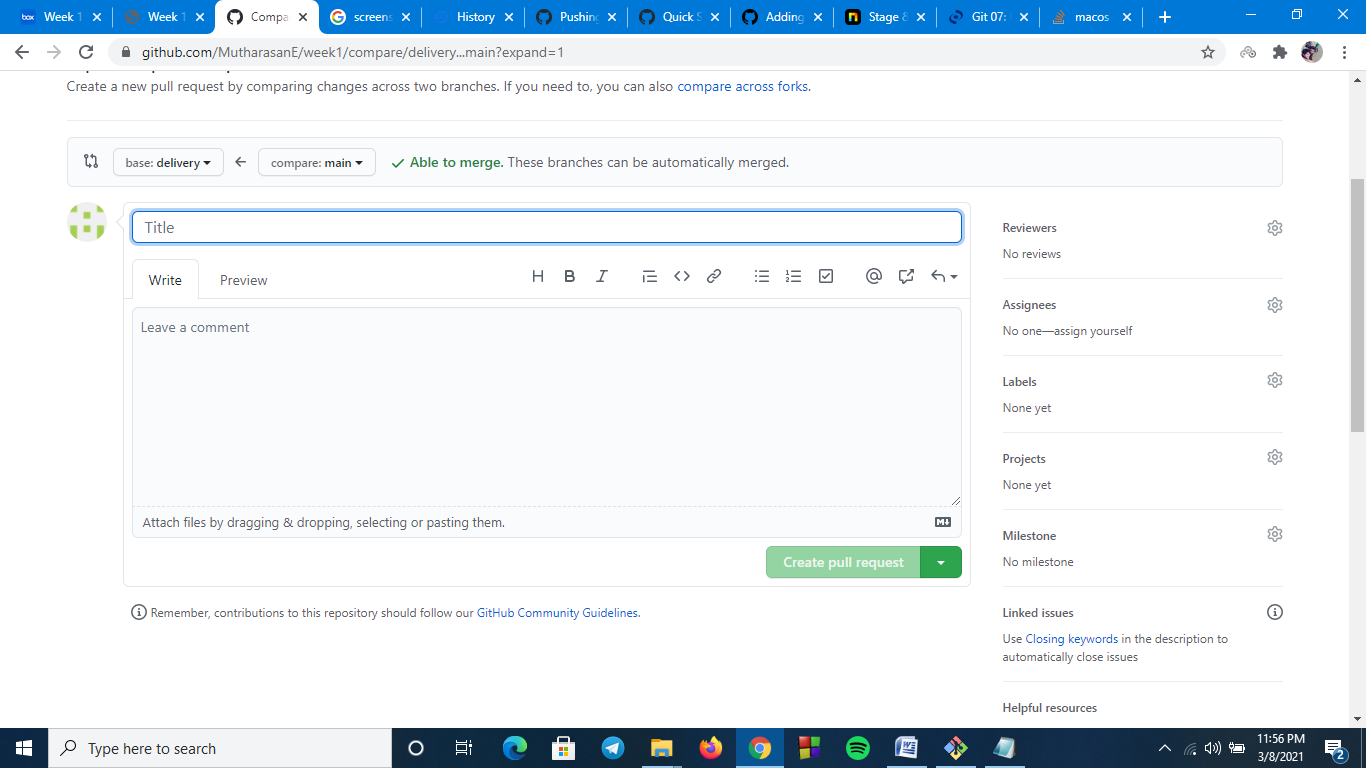
$ git add new.txt

$ git commit -m “Merge Resoved”

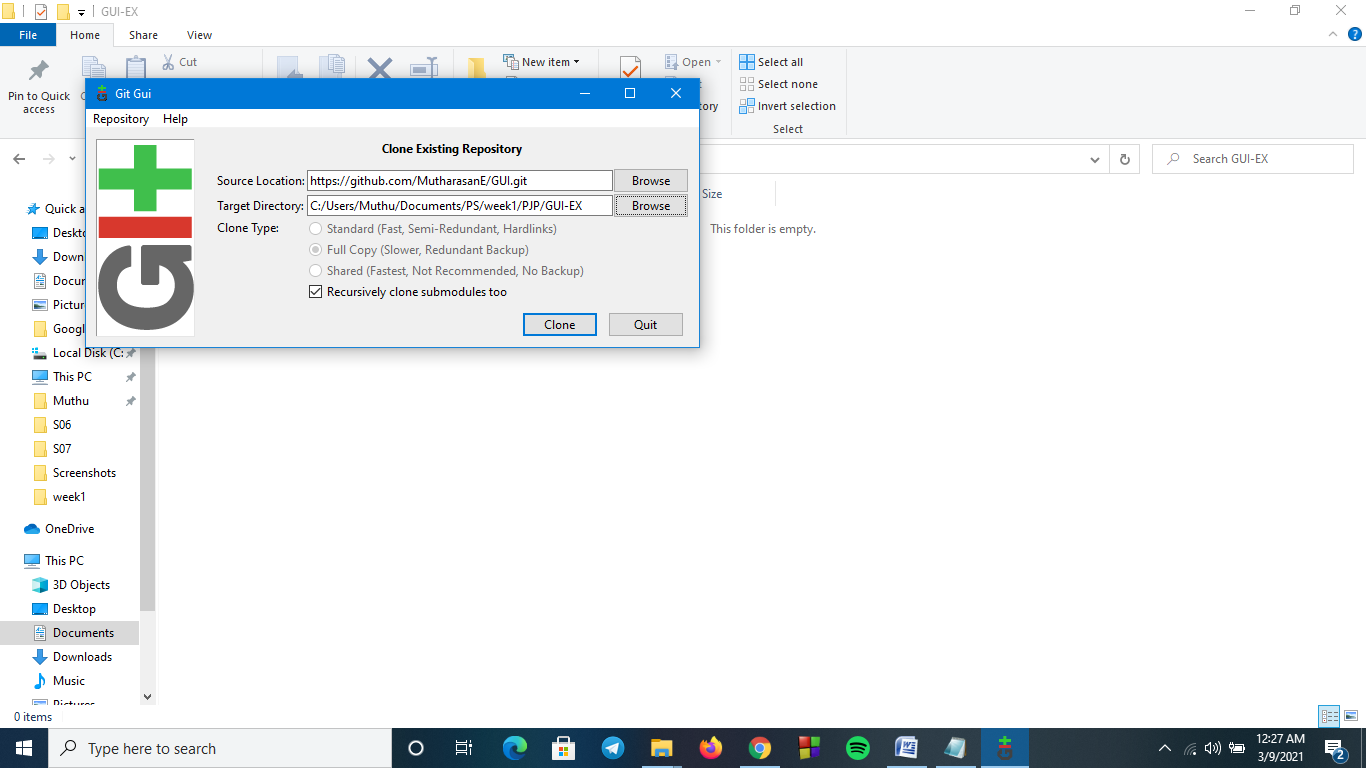


xi.Ensure the code is in sync .with latest changes across all branches:

I created “pull request” for all the branches from feature branch.

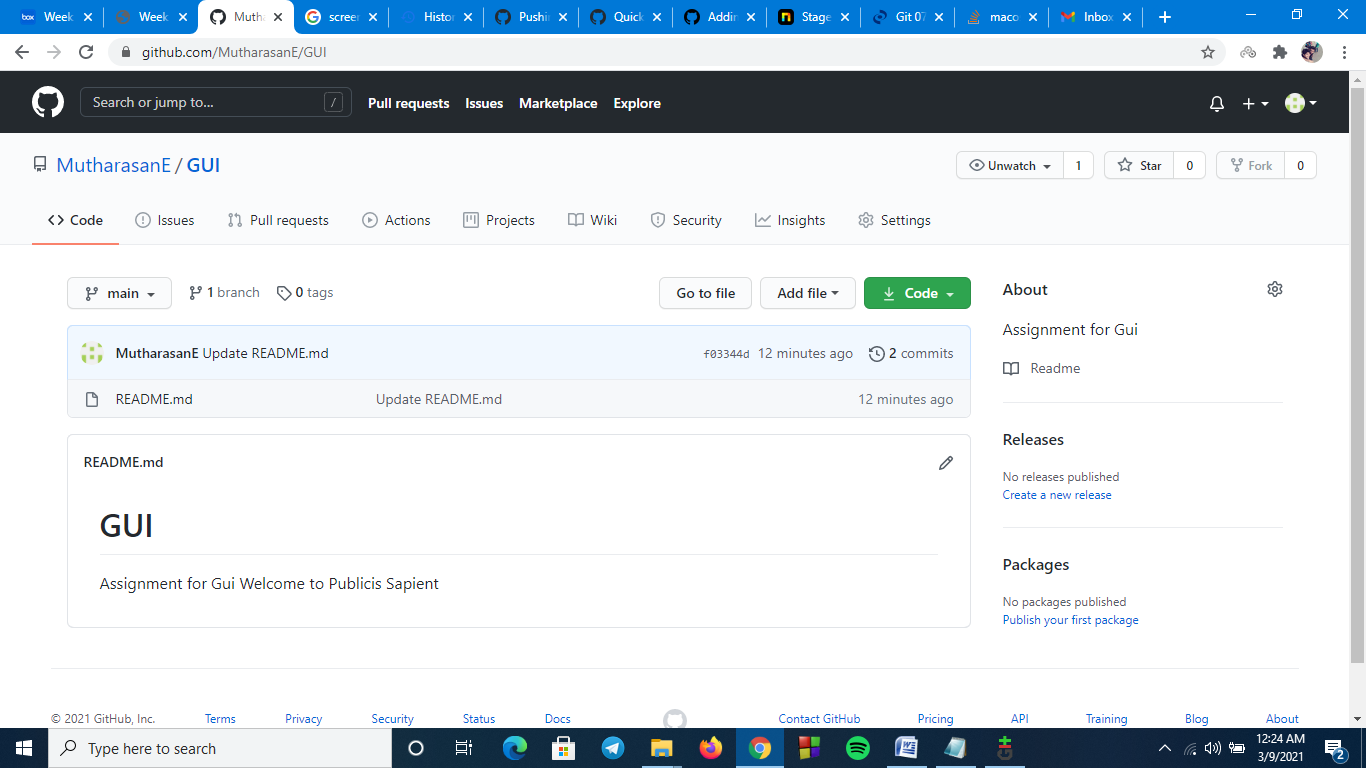


1.b i.Local repository:

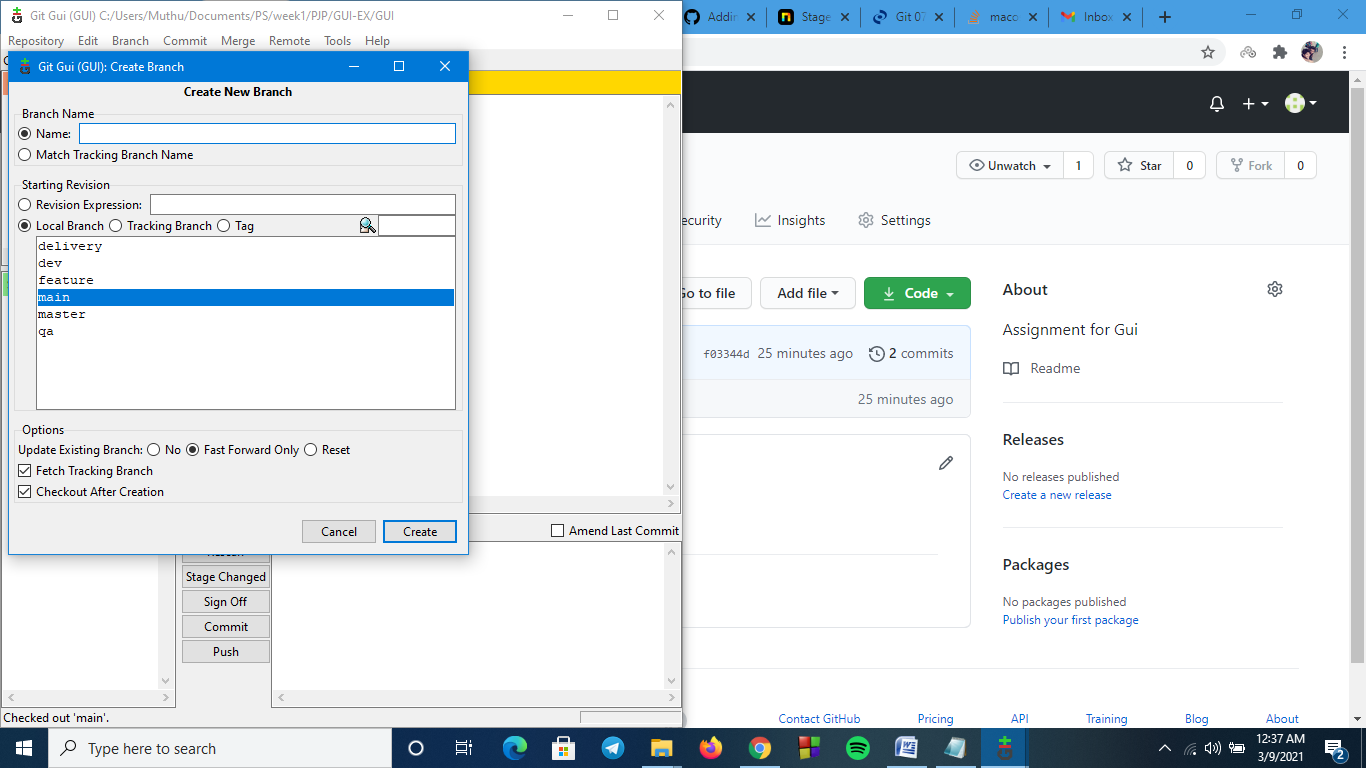


ii.Remote Repository:

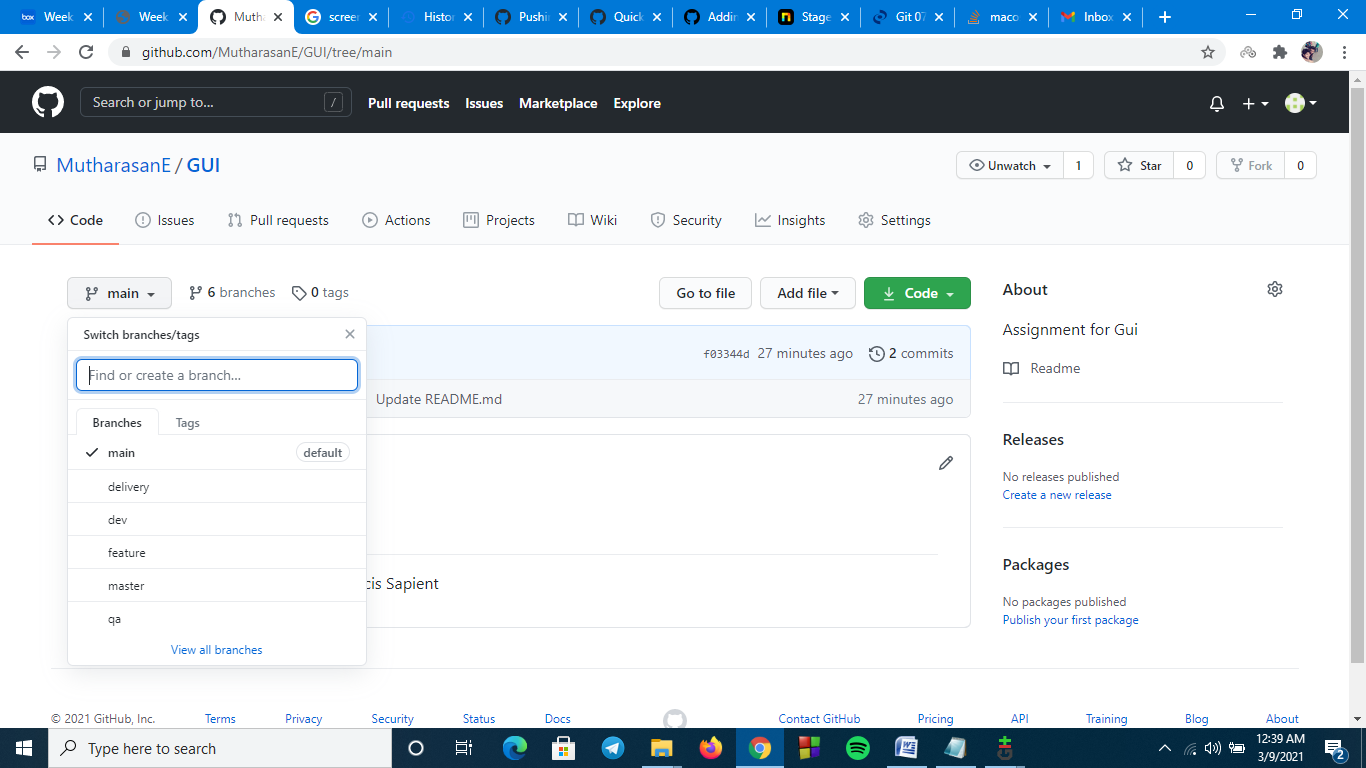
Created a new remote repository named GUI



iii.local branches:

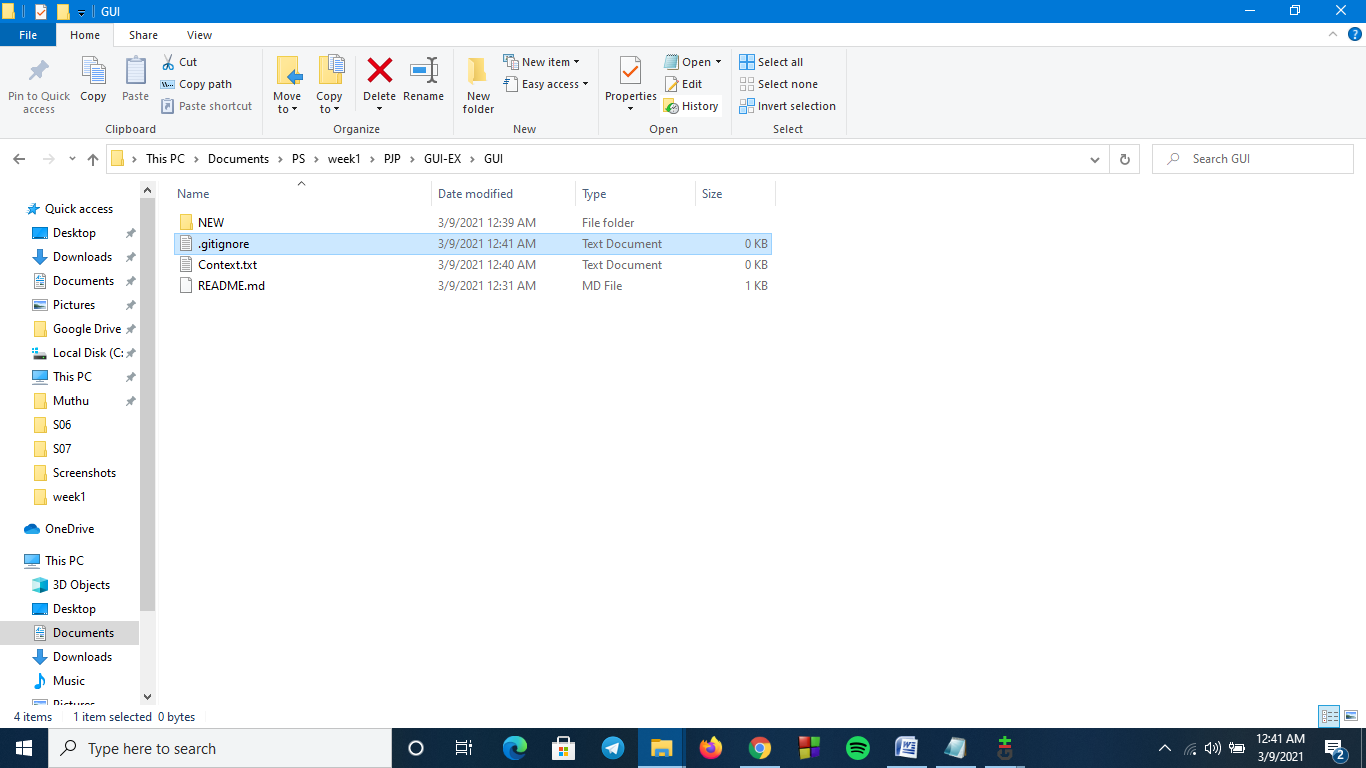


iv.remote branches:

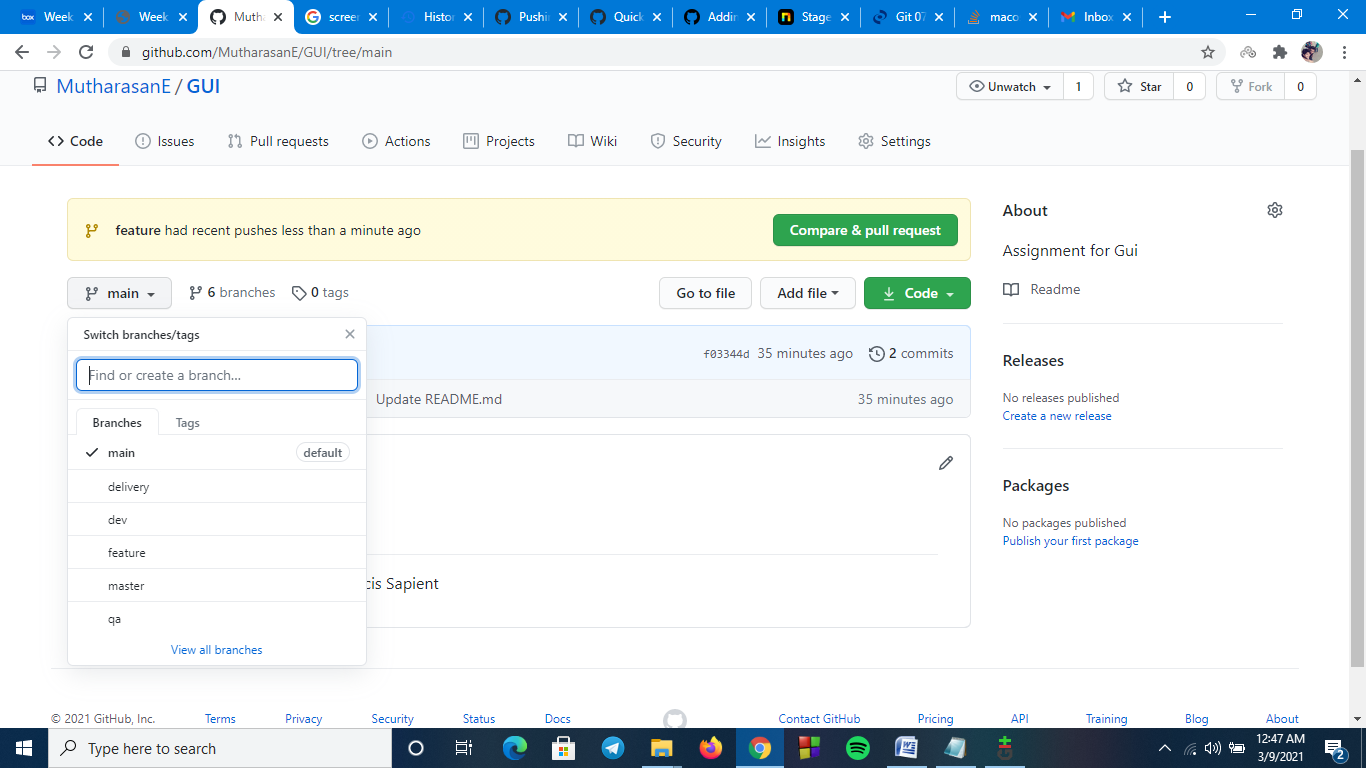
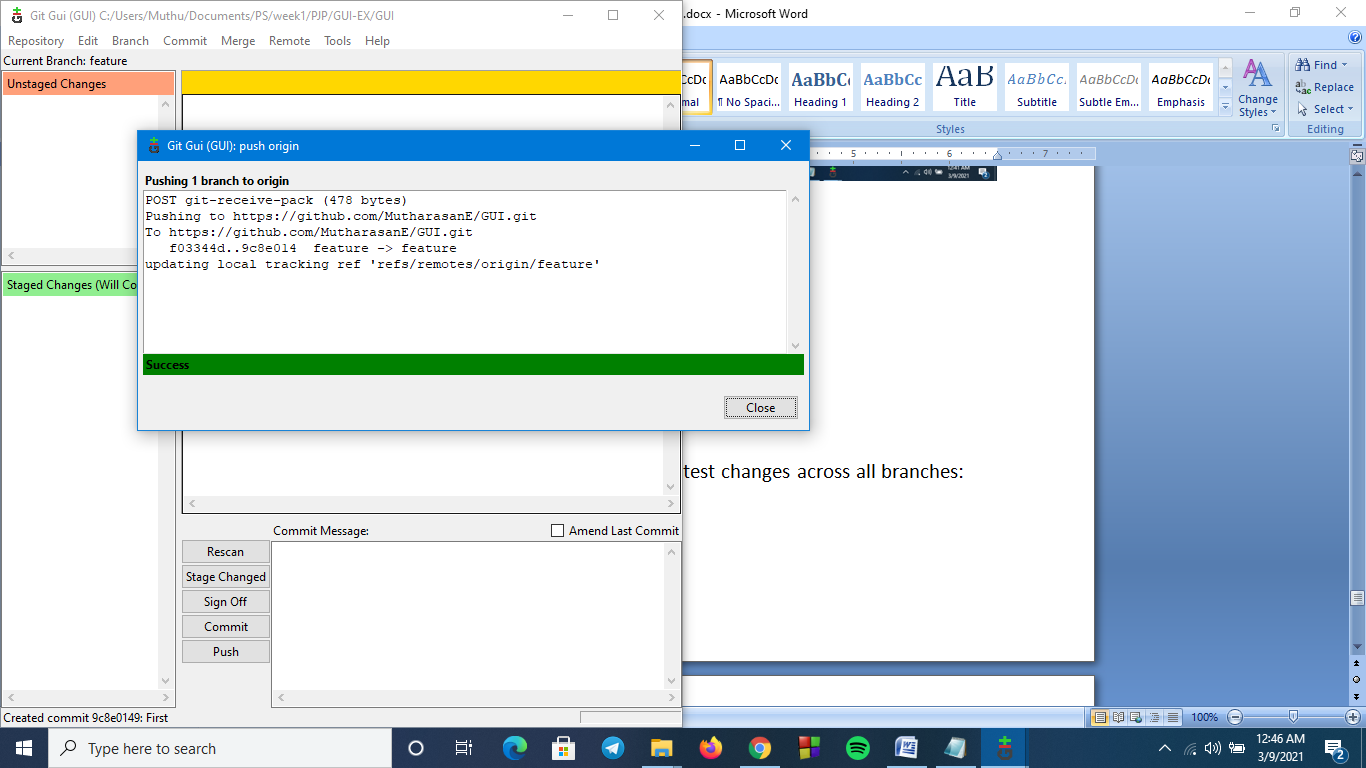


v.add files add folders:

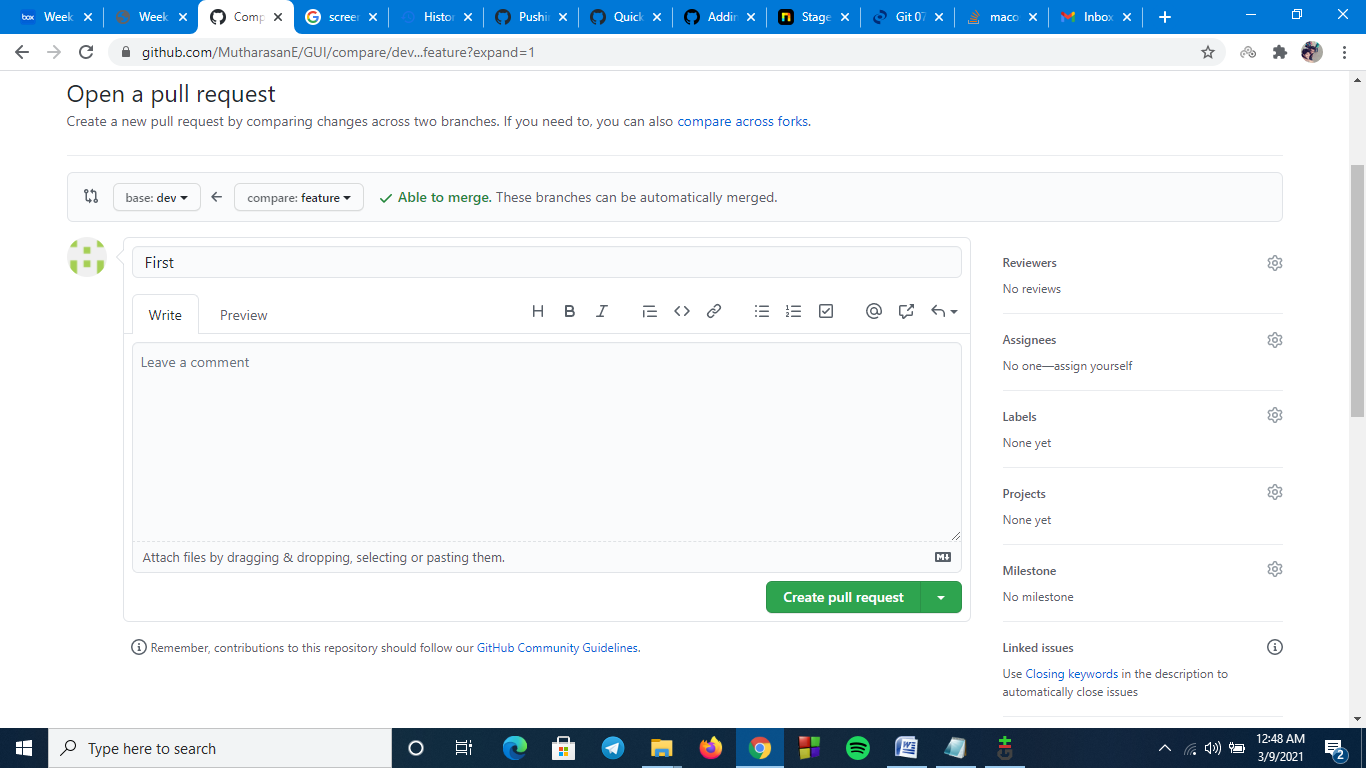
Folder - NEW Files - .gitignore,Context.txt



vi.Check-in ,Stage, Commit,Push into feature:

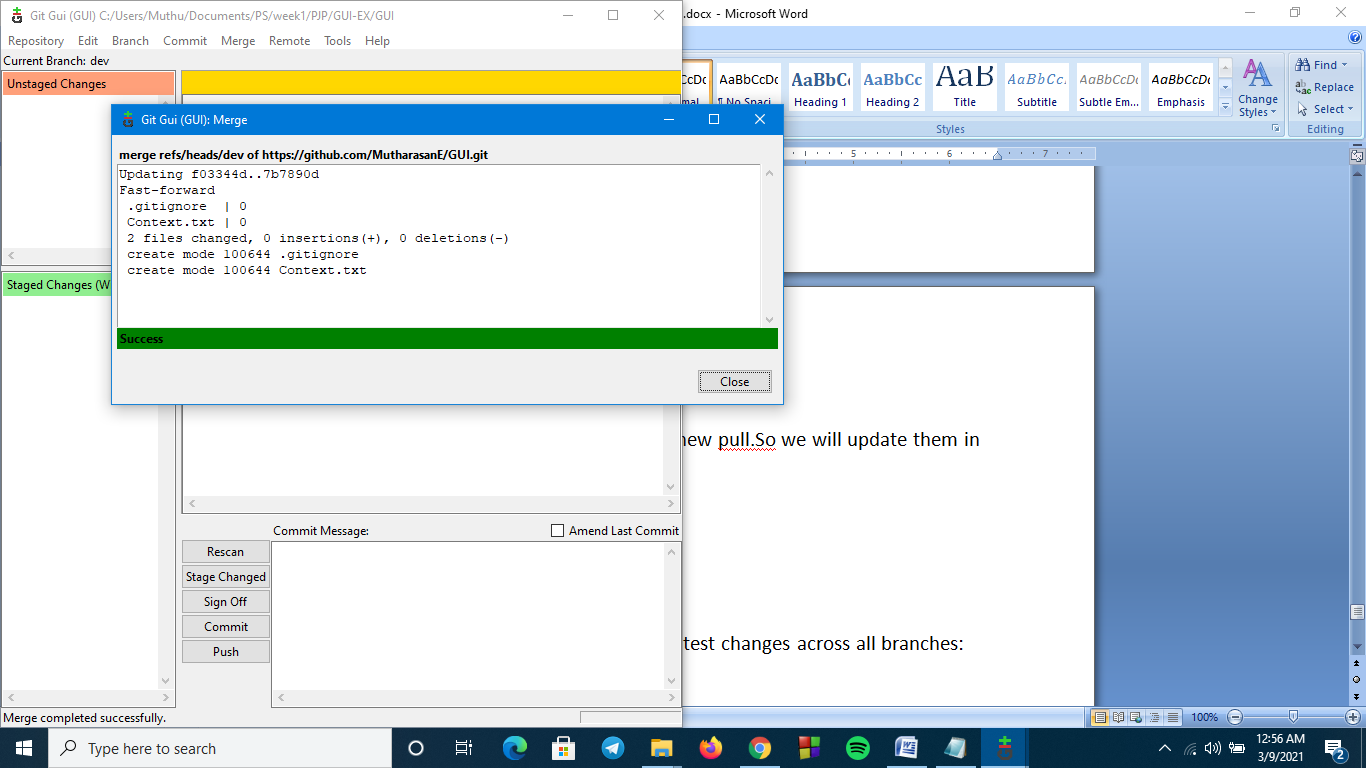


vii.Promote code from feature to dev:



viii.Latest code from remote to local:

* dev in remote repository has new pull.So we will update them in local repository using GUI.
* We have to create a fetch and merge in local dev branch.



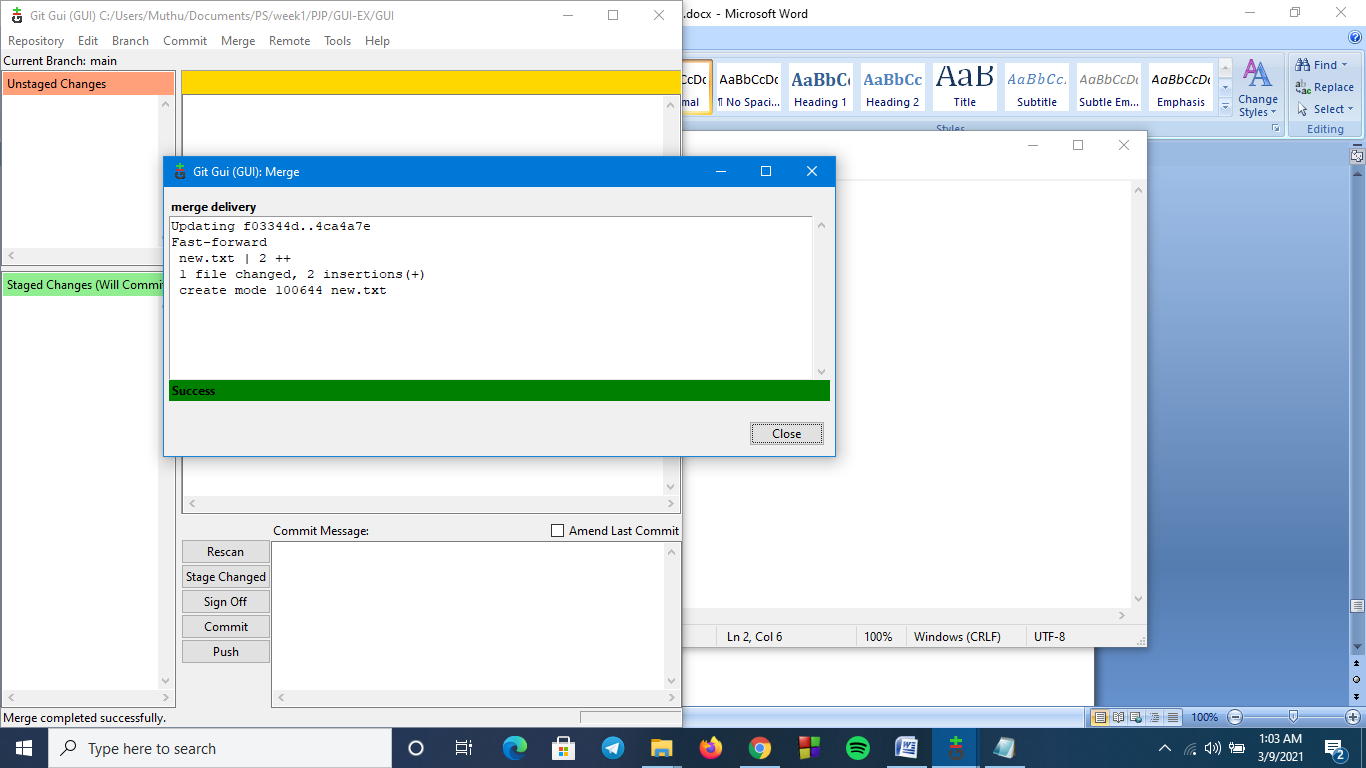
ix.Checkout Vs Pull:

It was defined earlier.

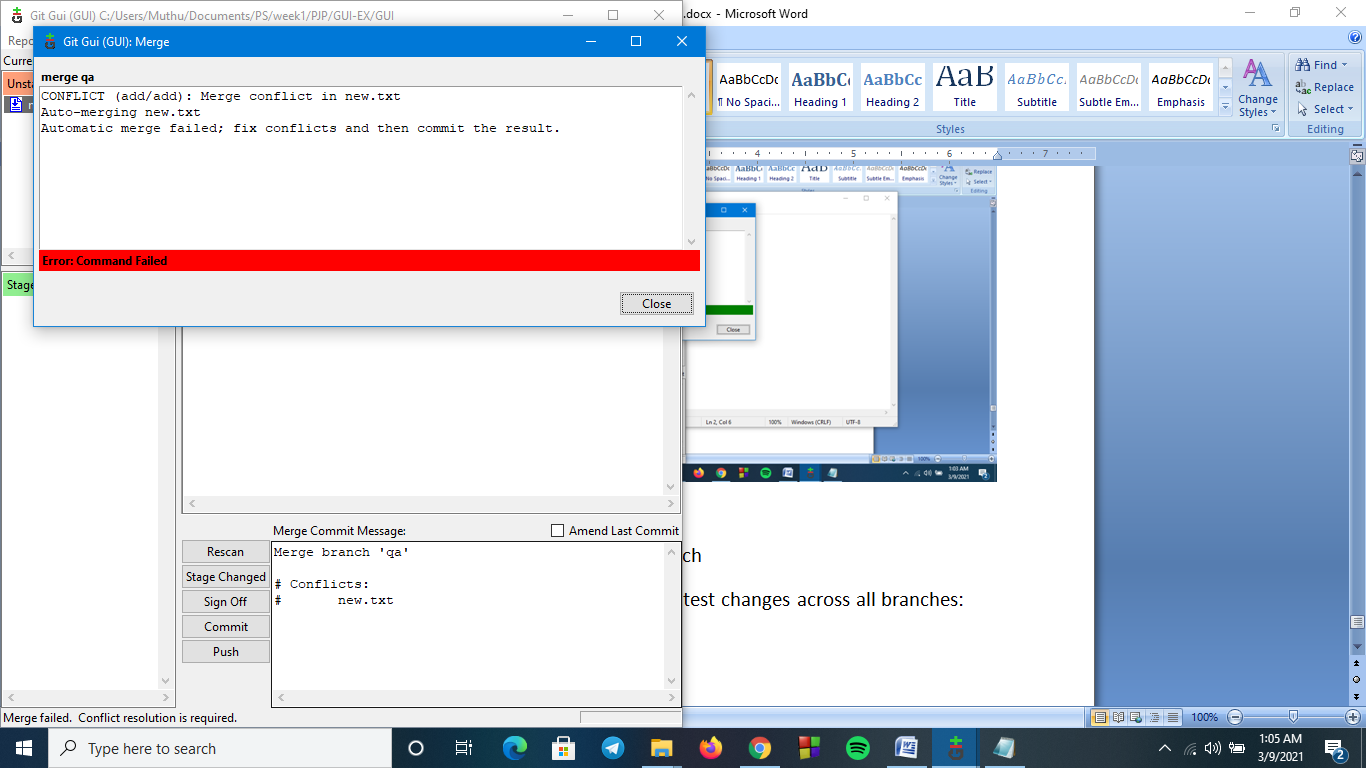
x.Handle Merge Conflict:

The Merge conflict was created by the same Merge Conflict method used above but here with the help of GUI Tool rather than CLI.

First merge to main with branch delivery



Second merge to main with qa branch – Merge Conflict occurs here

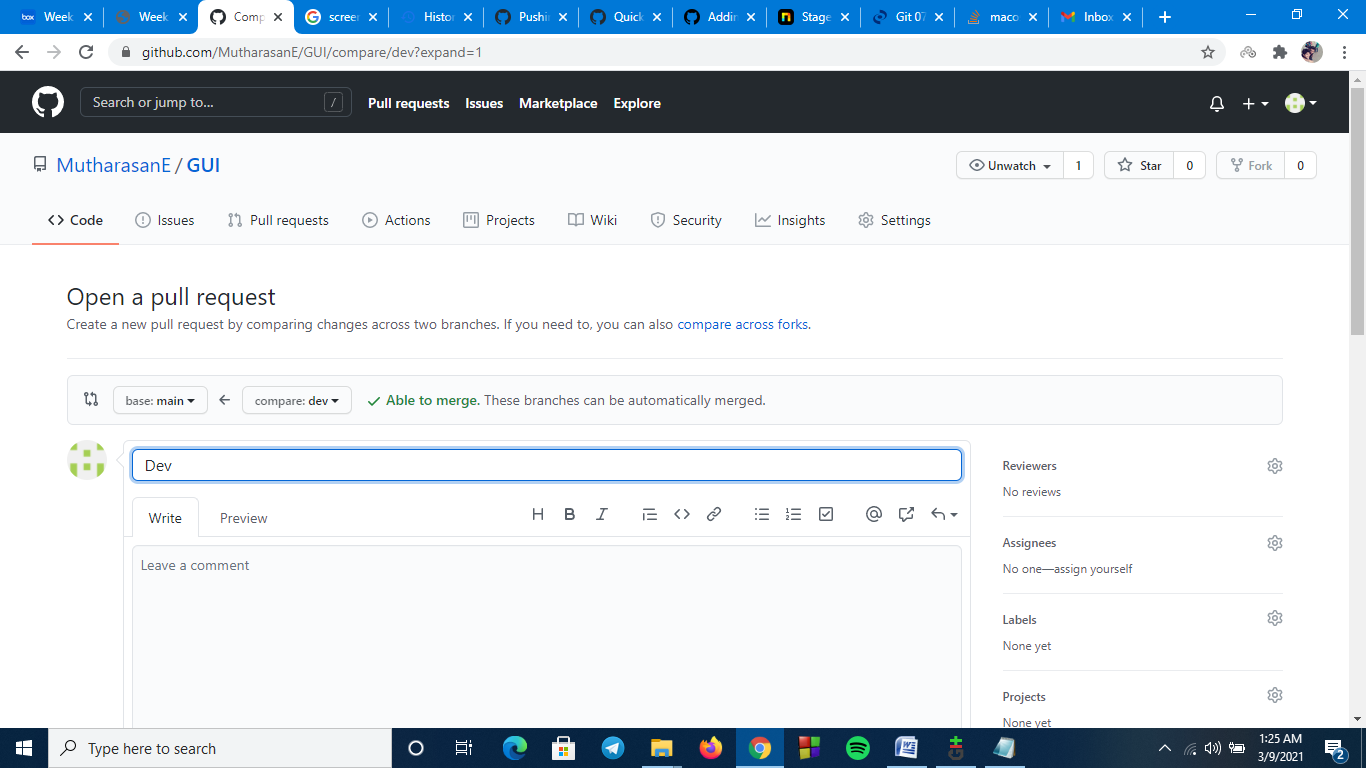


Solution:

Using Ctrl+T will resove this issue or commit->stage to commit.

xi.Ensure the code is in sync .with latest changes across all branches:

All File are updated to main.



1.c.

i.Reset / Revert (soft reset & hard reset):

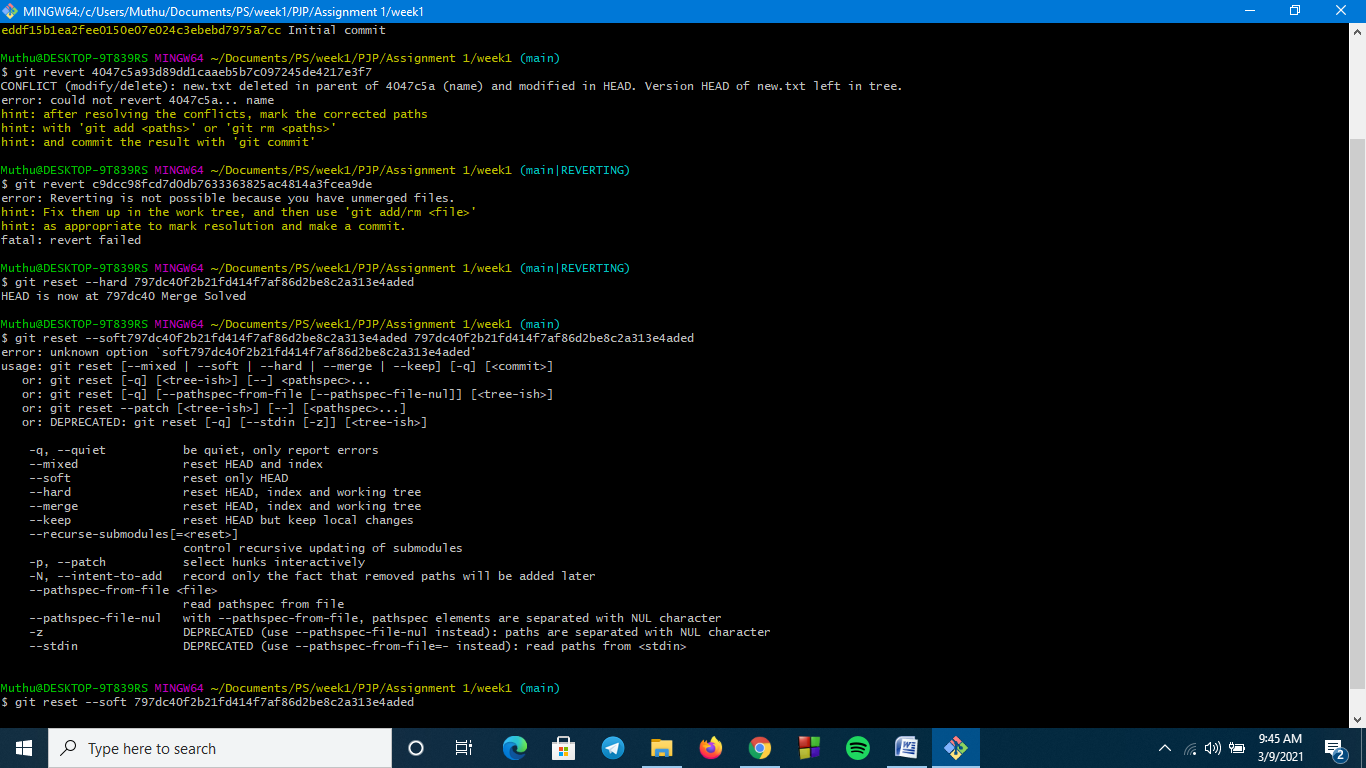
$ git log –stat

$ git log --pretty=oneline

$git revert<commit id>

$git reset –hard<commit id>

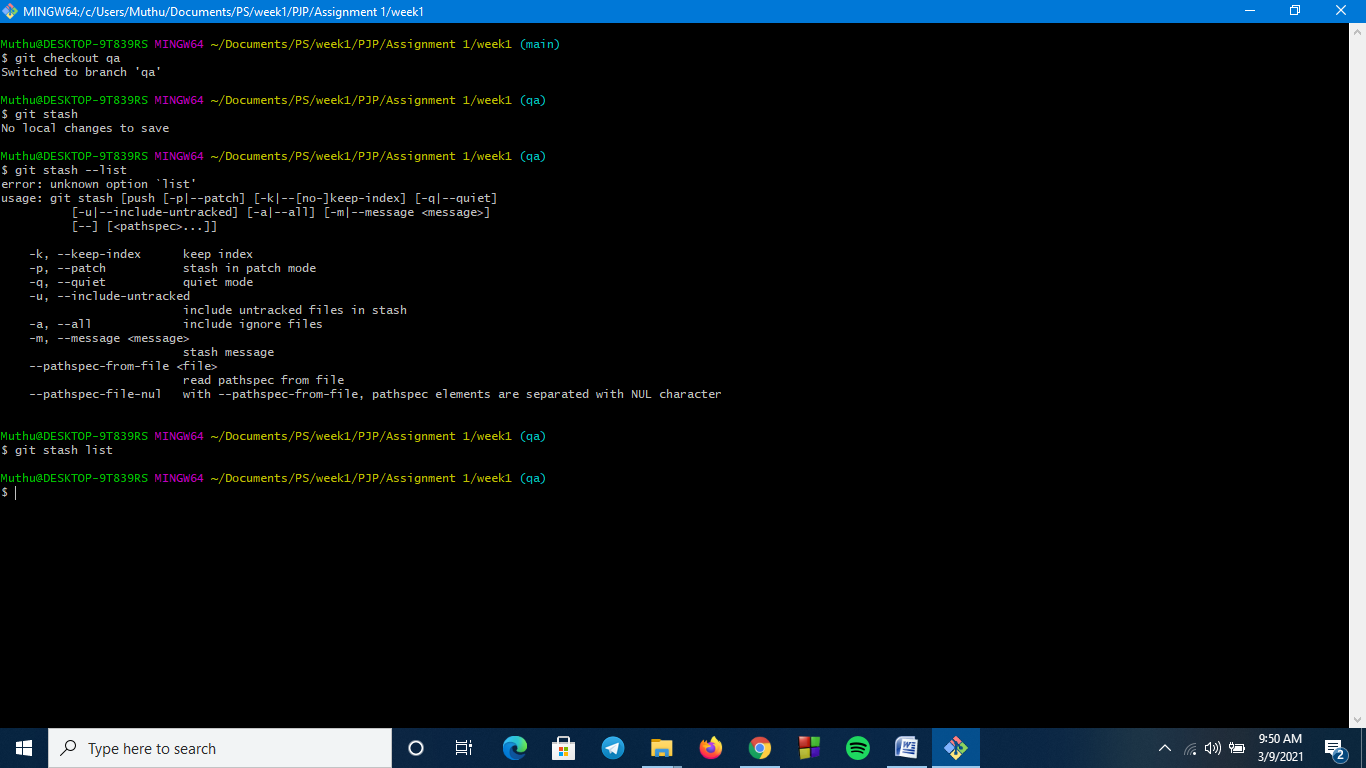
$ git reset –soft<commit id>



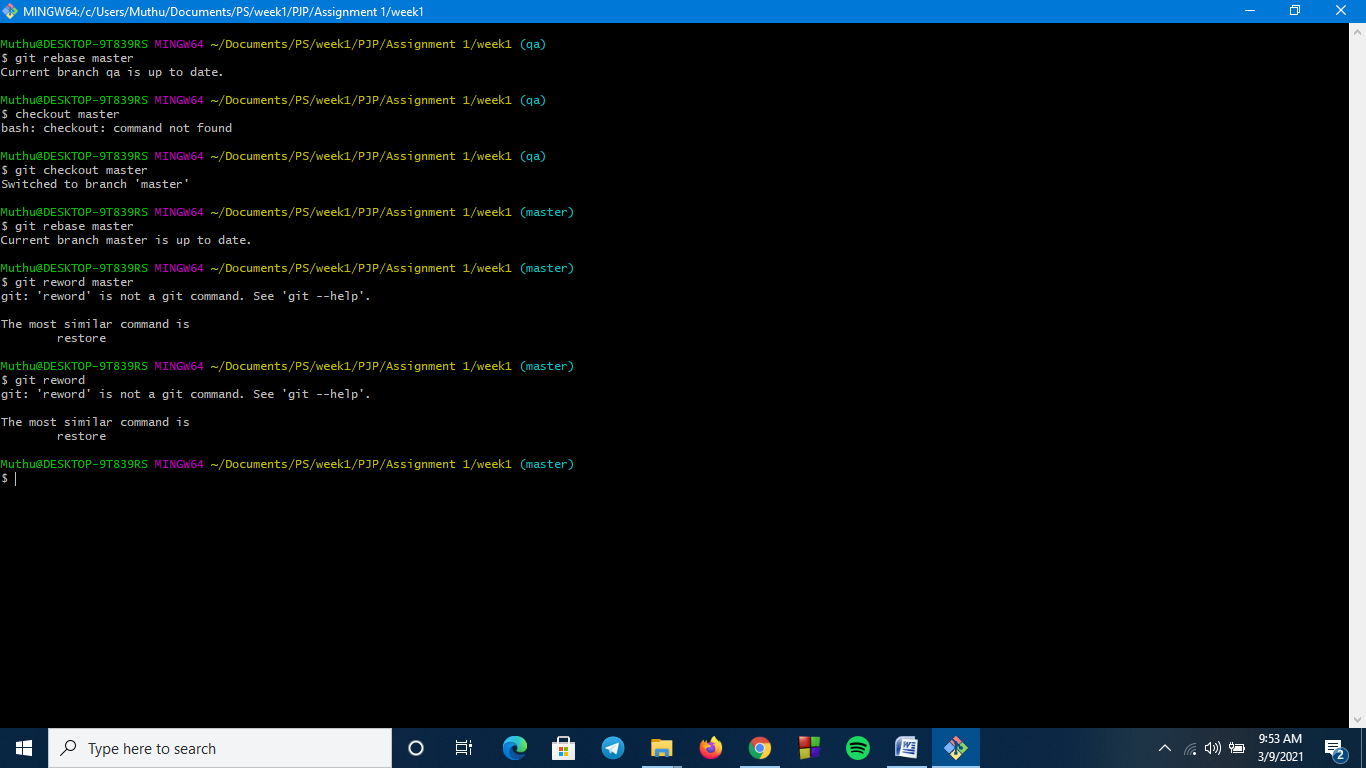
ii.Stash local changes during merge conflicts:

$git stash

$git stash list



iii.Rebasing – reword,edit,squash,fixup,exec,drop:



iv.git log,status & reflog:

