

ASSIGNMENT 1: Git and GitHub

Contents

I.	Demonstration of Knowledge	2
1.	Creation of the README.md file	2
2.	Checking the status #1	2
3.	Creating a Branch #1	3
4.	Opening a Pull request #1	3
5.	Merging and deploying to main branch #1	4
6.	Create Branch #2.....	5
7.	Adding files to a branch.....	6
8.	Checking the status and pushing them to Github.....	6
9.	Create another Pull request	7
10.	Merge, deploy, comment and delete branch	8
II.	Create .gitignore	8
III.	Create requirements.txt.....	12

By: Alejandra Cuadros Rivas

I. Demonstration of Knowledge

1. Creation of the README.md file

```
acuad@APCR-ASUS MINGW64 ~/Documents
$ ls
Default.rdp      'Python Scripts'/
Demo_Ale/          R/
'My Data Sources'/
'My Music'@        README.md
'RapidMiner'/
'My Pictures'@    'SQL Server Management Studio'/
'My Videos'@       'Sara Rivas'/
Outlook1.pst      'Sound Recordings'/

acuad@APCR-ASUS MINGW64 ~/Documents
$ mkdir CFG-Degree

acuad@APCR-ASUS MINGW64 ~/Documents
$ ls
CFG-Degree/        Outlook1.pst
Default.rdp        'Python Scripts'/
Demo_Ale/          R/
'My Data Sources'/
'My Music'@        README.md
'RapidMiner'/
'My Pictures'@    'SQL Server Management Studio'/
'My Videos'@       'Sara Rivas'/

acuad@APCR-ASUS MINGW64 ~/Documents
$ cd CFG-Degree/
acuad@APCR-ASUS MINGW64 ~/Documents/CFG-Degree
$ #Let's create the README file
```

```
acuad@APCR-ASUS MINGW64 ~/Documents/CFG-Degree
$ echo "# CFG-Assignments" >> README.md
git init
git add README.md
git commit -m "first commit"
git branch -M main
git remote add origin https://github.com/acuadrosr18/CFG-Assignments.git
git push -u origin main
Initialized empty Git repository in C:/Users/acuad/Documents/CFG-Degree/.git/
[master (root-commit) 1f85ad1] first commit
 1 file changed, 1 insertion(+)
 create mode 100644 README.md
Enumerating objects: 3, done.
Counting objects: 100% (3/3), done.
Writing objects: 100% (3/3), 229 bytes | 114.00 KiB/s, done.
Total 3 (delta 0), reused 0 (delta 0), pack-reused 0 (from 0)
To https://github.com/acuadrosr18/CFG-Assignments.git
 * [new branch]  main -> main
branch 'main' set up to track 'origin/main'.
```

2. Checking the status #1

```
acuad@APCR-ASUS MINGW64 ~/Documents/CFG-Degree (main)
$ git status
On branch main
Your branch is up to date with 'origin/main'.

nothing to commit, working tree clean
```

By: Alejandra Cuadros Rivas

3. Creating a Branch #1

We create a branch first on GitHub to add the edition of the GitHub 😊 I have no pictures of that, but I did it and then I opened the first pull request. Its name was Readme-edition

4. Opening a Pull request #1

By: Alejandra Cuadros Rivas

The screenshot shows a GitHub pull request interface for a repository named "Update README.md #1". The pull request was created by "acuadros18" and merged into the "main" branch from the "Readme-edition" branch. The merge was performed automatically because there were no conflicts. A green "Merge pull request" button is visible. Below the merge button, there is a "Rulesets" section with a note about requiring approval from specific reviewers before merging. There is also a message indicating that the branch has no conflicts with the base branch. At the bottom, there is a "Add a comment" section with a text input field and a "Comment" button.

5. Merging and deploying to main branch #1

We merge them, deployed it to main branch and then we add a comment and delete the branch.

The screenshot shows a GitHub pull request interface for a repository named "Update README.md #1". The pull request was merged by "acuadros18" into the "main" branch. The merge was successful and closed the pull request. A purple "Delete branch" button is visible. Below the merge message, there is a "Pull request successfully merged and closed" message with a note that the "Readme-edition" branch can be safely deleted. At the bottom, there is a "Add a comment" section with a text input field and a "Comment" button.

By: Alejandra Cuadros Rivas

The screenshot shows a GitHub pull request interface. At the top, it says 'Update README.md #1' with a 'Merged' button. Below that, there's a summary of the merge: 'acuadrosr18 merged 1 commit into main from Readme-edition now'. The main area shows a conversation thread. The first message is from 'acuadrosr18' saying 'First update with detailed information on the Readme.' Below it is a file named 'Update README.md'. A second message from 'acuadrosr18' says 'Incredible update!'. At the bottom, it shows that 'acuadrosr18' deleted the 'Readme-edition' branch.

6. Create Branch #2

```
acuad@APCR-ASUS MINGW64 ~/Documents/CFG-Degree (main)
$ git branch feature/adding-files

acuad@APCR-ASUS MINGW64 ~/Documents/CFG-Degree (main)
$ git branch
  feature/adding-files
* main

acuad@APCR-ASUS MINGW64 ~/Documents/CFG-Degree (main)
$ git switch feature/adding-files
Switched to branch 'feature/adding-files'

acuad@APCR-ASUS MINGW64 ~/Documents/CFG-Degree (feature/adding-files)
$ git branch
* feature/adding-files
  main
```

By: Alejandra Cuadros Rivas

7. Adding files to a branch

I added manually from the local repository and also added some through Git Bash:

```
acuad@APCR-ASUS MINGW64 ~/Documents/CFG-Degree (feature/adding-files)
$ touch presentation.pptx

acuad@APCR-ASUS MINGW64 ~/Documents/CFG-Degree (feature/adding-files)
$ ls
README.md  presentation.pptx

acuad@APCR-ASUS MINGW64 ~/Documents/CFG-Degree (feature/adding-files)
$ nano document.pdf

acuad@APCR-ASUS MINGW64 ~/Documents/CFG-Degree (feature/adding-files)
$ touch pleasegivemethefullmark.txt

acuad@APCR-ASUS MINGW64 ~/Documents/CFG-Degree (feature/adding-files)
$ ls
README.md  document.pdf  pleasegivemethefullmark.txt  presentation.pptx
```

8. Checking the status and pushing them to Github

```
acuad@APCR-ASUS MINGW64 ~/Documents/CFG-Degree (feature/adding-files)
$ git status
On branch feature/adding-files
Untracked files:
  (use "git add <file>..." to include in what will be committed)
    Claps.mp3
    document.pdf
    pleasegivemethefullmark.txt
    presentation.pptx

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

acuad@APCR-ASUS MINGW64 ~/Documents/CFG-Degree (feature/adding-files)
$ git add .

acuad@APCR-ASUS MINGW64 ~/Documents/CFG-Degree (feature/adding-files)
$ git commit -m "Addition of four documents vitals for the project"
[feature/adding-files 5fb2c81] Addition of four documents vitals for the project
 4 files changed, 2 insertions(+)
 create mode 100644 Claps.mp3
 create mode 100644 document.pdf
 create mode 100644 pleasegivemethefullmark.txt
 create mode 100644 presentation.pptx
```

```
acuad@APCR-ASUS MINGW64 ~/Documents/CFG-Degree (feature/adding-files)
$ git push --set-upstream origin feature/adding-files
Enumerating objects: 8, done.
Counting objects: 100% (8/8), done.
Delta compression using up to 8 threads
Compressing objects: 100% (5/5), done.
Writing objects: 100% (8/8), 1.04 MiB | 1.30 MiB/s, done.
Total 8 (delta 0), reused 0 (delta 0), pack-reused 0 (from 0)
remote:
remote: Create a pull request for 'feature/adding-files' on GitHub by visiting:
remote:   https://github.com/acuadrosr18/CFG-Assignments/pull/new/feature/adding-files
remote:
To https://github.com/acuadrosr18/CFG-Assignments.git
 * [new branch]      feature/adding-files -> feature/adding-files
branch 'feature/adding-files' set up to track 'origin/feature/adding-files'.
```

By: Alejandra Cuadros Rivas

```
acuad@APCR-ASUS MINGW64 ~/Documents/CFG-Degree (Feature/adding-files)
$ git status
On branch feature/adding-files
Your branch is up to date with 'origin/feature/adding-files'.

nothing to commit, working tree clean
```

9. Create another Pull request

Addition of four documents vitals for the project #2

[! Open](#) acuadrosr18 wants to merge 1 commit into [main](#) from [feature/adding-files](#)

Conversation 0 Commits 1 Checks 0 Files changed 4

acuadrosr18 commented now

we add the next documents:

1. Claps.mp3
2. document.pdf
3. pleasegivemethefullmark.txt
4. presentation.pptx

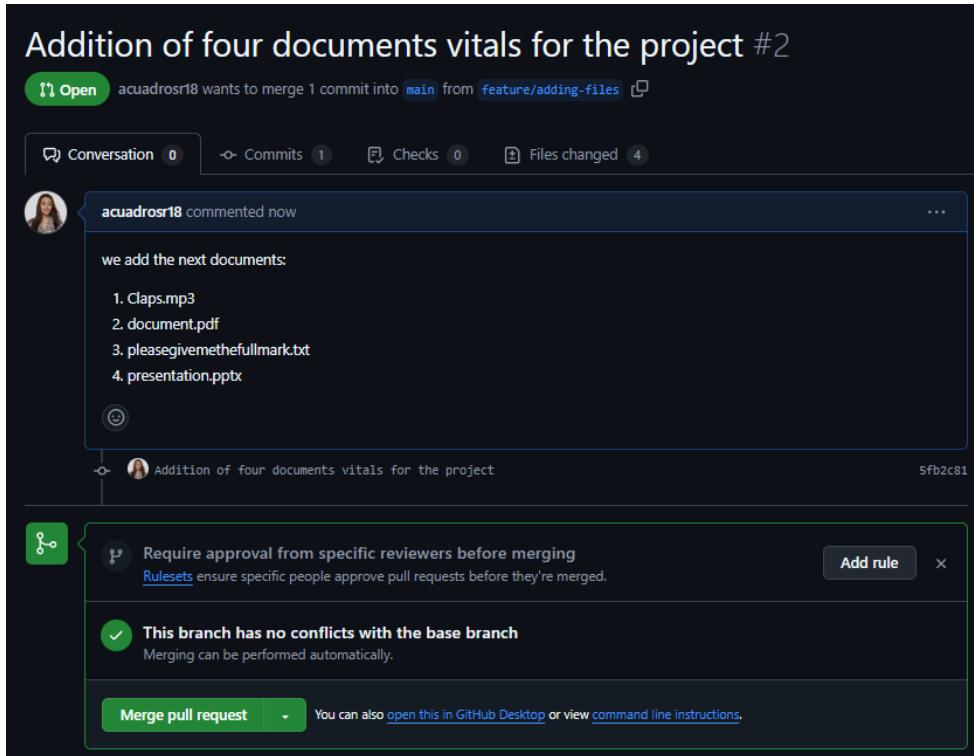
...

acuadrosr18 Addition of four documents vitals for the project 5fb2c81

Require approval from specific reviewers before merging [Rulesets](#) ensure specific people approve pull requests before they're merged. [Add rule](#) [x](#)

This branch has no conflicts with the base branch
Merging can be performed automatically.

[Merge pull request](#) You can also [open this in GitHub Desktop](#) or view [command line instructions](#).



By: Alejandra Cuadros Rivas

10. Merge, deploy, comment and delete branch

The screenshot shows a GitHub pull request interface. At the top, a purple banner indicates the pull request has been merged. Below this, the main interface shows a conversation with one message from the user 'acuadrosr18' stating: 'we add the next documents: 1. Claps.mp3 2. document.pdf 3. pleasegivemethefullmark.txt 4. presentation.pptx'. A commit message follows: 'Addition of four documents vitals for the project'. The pull request is then merged by the user 'acuadrosr18' into the 'main' branch. A second comment from 'acuadrosr18' says: 'Excellent documents, crucial for the development of the project.' A summary message at the bottom states: 'Pull request successfully merged and closed' and 'You're all set—the feature/adding-files branch can be safely deleted.' Finally, a message indicates the branch has been deleted.

II. Create .gitignore

The `.gitignore` file in a Git repository is used to specify the files and directories that Git **should ignore and not track**. This helps in excluding files that contain sensitive data, intermediate build artifacts, or other files that do not need to be shared or stored in version control. By using a `.gitignore` file, we keep the repository clean and avoid cluttering it with unnecessary files, which also improves the performance of Git operations.

We created 3 `*.xlsx` documents and add manually an `*.png` file, and when we pushed the branch Git ignore them during the push operation. They are not hidden in the file system, rather, they are simply not tracked by Git. We won't see them listed in Git status reports or in the repository on GitHub, but we can still find and access them normally on our local file system.

```
acuad@APCR-ASUS MINGW64 ~/Documents/CFG-Degree (main)
$ git branch adding-documents-and-create-.gitignore
```

By: Alejandra Cuadros Rivas

```
acuad@APCR-ASUS MINGW64 ~/Documents/CFG-Degree (adding-documents-and-create-.get
ignore)
$ nano Finanzas.xlsx

acuad@APCR-ASUS MINGW64 ~/Documents/CFG-Degree (adding-documents-and-create-.get
ignore)
$ nano Finance.xlsx

acuad@APCR-ASUS MINGW64 ~/Documents/CFG-Degree (adding-documents-and-create-.get
ignore)
$ nano Confidential.xlsx
```

The screenshot shows the nano 8.1 text editor with the following content:

```
GNU nano 8.1          .gitignore          Modified
# Ignore all the excel files
*.xlsx

# Ignore files
Screenshot.png

^G Help      ^A0 Write Out   ^F Where Is    ^K Cut        ^T Execute    ^C Location
^X Exit      ^R Read File   ^\ Replace     ^U Paste      ^J Justify    ^/ Go To Line
```

```
acuad@APCR-ASUS MINGW64 ~/Documents/CFG-Degree (adding-documents-and-create-.get
ignore)
acuad@APCR-ASUS MINGW64 ~/Documents/CFG-Degree (adding-documents-and-create-.getignore)
$ git status
On branch adding-documents-and-create-.getignore
Untracked files:
  (use "git add <file>..." to include in what will be committed)
    .gitignore

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

acuad@APCR-ASUS MINGW64 ~/Documents/CFG-Degree (adding-documents-and-create-.getignore)
$ git add .

acuad@APCR-ASUS MINGW64 ~/Documents/CFG-Degree (adding-documents-and-create-.getignore)
$ git status
On branch adding-documents-and-create-.getignore
Changes to be committed:
  (use "git restore --staged <file>..." to unstage)
    new file:   .gitignore
```

```
acuad@APCR-ASUS MINGW64 ~/Documents/CFG-Degree (adding-documents-and-create-.getignore)
$ git commit -m "Addition of the .gitignore file"
[adding-documents-and-create-.getignore 5dfd456] Addition of the .gitignore file
  1 file changed, 6 insertions(+)
  create mode 100644 .gitignore
```

By: Alejandra Cuadros Rivas

```
acuad@APCR-ASUS MINGW64 ~/Documents/CFG-Degree (adding-documents-and-create-.getignore)
$ git push --set-upstream origin adding-documents-and-create-.getignore
Enumerating objects: 6, done.
Counting objects: 100% (6/6), done.
Delta compression using up to 8 threads
Compressing objects: 100% (4/4), done.
Writing objects: 100% (6/6), 549 bytes | 183.00 KiB/s, done.
Total 6 (delta 0), reused 0 (delta 0), pack-reused 0 (from 0)
remote:
remote: Create a pull request for 'adding-documents-and-create-.getignore' on GitHub by visiting
:
remote:     https://github.com/acuadrosr18/CFG-Assignments/pull/new/adding-documents-and-create-
remote:     .getignore
remote:
To https://github.com/acuadrosr18/CFG-Assignments.git
 * [new branch]      adding-documents-and-create-.getignore -> adding-documents-and-create-.geti
gnore
branch 'adding-documents-and-create-.getignore' set up to track 'origin/adding-documents-and-cre
ate-.getignore'.

acuad@APCR-ASUS MINGW64 ~/Documents/CFG-Degree (adding-documents-and-create-.getignore)
$ git status
On branch adding-documents-and-create-.getignore
Your branch is up to date with 'origin/adding-documents-and-create-.getignore'.

nothing to commit, working tree clean
```

By: Alejandra Cuadros Rivas

Addition of the .gitignore file #3

Merged acuadrosr18 merged 1 commit into main from adding-documents-and-create-.gitignore now

Conversation 0 Commits 1 Checks 0 Files changed 1

acuadrosr18 commented now
Addition of the .gitignore file

acuadrosr18 Addition of the .gitignore file 5dfd456

acuadrosr18 merged commit 730c50c into main now Revert

acuadrosr18 commented now
Necessary! Thanks.

acuadrosr18 deleted the adding-documents-and-create-.gitignore branch now Restore branch

Code Issues Pull requests Actions Projects Security

0 stars 0 forks 1 watching 1 Branch 0 Tags Activity

Private repository

main Go to file + <> Code

acuadrosr18 Merge pull request #3 from acuadrosr18/adding-docu... 730c50c · 5 minutes ago

File	Description	Time
.gitignore	Addition of the .gitignore file	10 minutes ago
Claps.mp3	Addition of four documents vitals for the ...	38 minutes ago
README.md	Update README.md	1 hour ago
document.pdf	Addition of four documents vitals for the ...	38 minutes ago
pleasegivemethefullmark.txt	Addition of four documents vitals for the ...	38 minutes ago
presentation.pptx	Addition of four documents vitals for the ...	38 minutes ago

If I execute the push, they appear because files were already tracked by Git before I added them to .gitignore,

By: Alejandra Cuadros Rivas

File	Description	Time Ago
.gitignore	Addition of the .gitignore file	18 minutes ago
Claps.mp3	Addition of four documents vitals for the project	45 minutes ago
Confidential.xlsx	addition of confidential documents	6 minutes ago
Finance.xlsx	addition of confidential documents	6 minutes ago
Finanzas.xlsx	addition of confidential documents	6 minutes ago
README.md	Update README.md	2 hours ago
Screenshot.png	addition of confidential documents	6 minutes ago
document.pdf	Addition of four documents vitals for the project	45 minutes ago
pleasegivemethefullmark.txt	Addition of four documents vitals for the project	45 minutes ago
presentation.pptx	Addition of four documents vitals for the project	45 minutes ago

III. Create requirements.txt

The **requirements.txt** file in a Python project lists all the libraries and their specific versions that the project depends on.

This helps ensure that anyone who sets up the project uses the exact versions required, avoiding compatibility issues and maintaining consistent behaviour across different environments.

It simplifies project setup by allowing users to install all dependencies with a single command:

pip install -r requirements.txt.

```
acuad@APCR-ASUS MINGW64 ~/Documents/CFG-Degree (main)
$ git branch requirements

acuad@APCR-ASUS MINGW64 ~/Documents/CFG-Degree (main)
$ git switch requirements
Switched to branch 'requirements'

acuad@APCR-ASUS MINGW64 ~/Documents/CFG-Degree (requirements)
$ nano requirements.txt
```

By: Alejandra Cuadros Rivas

```
acuad@APCR-ASUS MINGW64 ~/Documents/CFG-Degree (requirements)
$ git status
On branch requirements
Untracked files:
  (use "git add <file>..." to include in what will be committed)
    requirements.txt

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

acuad@APCR-ASUS MINGW64 ~/Documents/CFG-Degree (requirements)
$ git add .

acuad@APCR-ASUS MINGW64 ~/Documents/CFG-Degree (requirements)
$ git commit -m "Adding requirements for the project"
[requirements 5d26541] Adding requirements for the project
 1 file changed, 1 insertion(+)
 create mode 100644 requirements.txt

acuad@APCR-ASUS MINGW64 ~/Documents/CFG-Degree (requirements)
$ git push
fatal: The current branch requirements has no upstream branch.
To push the current branch and set the remote as upstream, use

  git push --set-upstream origin requirements

To have this happen automatically for branches without a tracking
upstream, see 'push.autoSetupRemote' in 'git help config'.

acuad@APCR-ASUS MINGW64 ~/Documents/CFG-Degree (requirements)
$ ^C

acuad@APCR-ASUS MINGW64 ~/Documents/CFG-Degree (requirements)
$ git push --set-upstream origin requirements
Enumerating objects: 4, done.
Counting objects: 100% (4/4), done.
Delta compression using up to 8 threads
Compressing objects: 100% (2/2), done.
Writing objects: 100% (3/3), 307 bytes | 307.00 KiB/s, done.
Total 3 (delta 1), reused 0 (delta 0), pack-reused 0 (from 0)
remote: Resolving deltas: 100% (1/1), completed with 1 local object.
remote:
remote: Create a pull request for 'requirements' on GitHub by visiting:
remote:   https://github.com/acuadrosr18/CFG-Assignments/pull/new/requirements
remote:
remote: To https://github.com/acuadrosr18/CFG-Assignments.git
 * [new branch]      requirements -> requirements
branch 'requirements' set up to track 'origin/requirements'.
```