

INTRODUCCIÓN GIT

David Alfonso Barbosa Gómez, Juan José Díaz Gómez

December 2024

PARTE I (Trabajo Individual)

Parte David Barbosa

1. Crea un repositorio localmente.

Solución

```
C:\Users\david.barbosa-g\Documents\LAB1CVDS>notepad README.md
```

2. Agrega un archivo de ejemplo al repositorio, el README.md puede ser una gran opción.

Solución

```
C:\Users\david.barbosa-g\Documents\LAB1CVDS>notepad README.md
```

3. Averigua para qué sirve y como se usan estos comandos git add y git commit -m "mensaje"
4. Abre una cuenta de github, si ya la tienes, enlazala con el correo institucional.
5. Crea un repositorio en blanco (vacío) e GitHub.

Solución

Create a new repository

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

Required fields are marked with an asterisk (*).

Owner *

DavidBarbosag

Repository name *

LAB1P1

LAB1P1 is available.

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

Description (optional)

Parte individual lab 1 CVDS

☐ Public

Anyone on the internet can see this repository. You choose who can commit.

☒ Private

You choose who can see and commit to this repository.

Initialize this repository with:

☐ Add a README file

This is where you can write a long description for your project. [Learn more about READMEs.](#)

Add .gitignore

.gitignore template: None

Choose which files not to track from a list of templates. [Learn more about ignoring files.](#)

Choose a license

License: None

A license tells others what they can and can't do with your code. [Learn more about licenses.](#)

ⓘ You are creating a private repository in your personal account.

Create repository

6. Configura el repositorio local con el repositorio remoto.


Solución

```
C:\Users\david.barbosa-g\Documents\PartelCVDS>git init
Initialized empty Git repository in C:/Users/david.barbosa-g/Documents/PartelCVDS/.git/
C:\Users\david.barbosa-g\Documents\PartelCVDS>git add .
C:\Users\david.barbosa-g\Documents\PartelCVDS>git commit -m "First Commit"
On branch master
Initial commit
nothing to commit (create/copy files and use "git add" to track)
C:\Users\david.barbosa-g\Documents\PartelCVDS>git remote add https://github.com/DavidBarbosag/LAB1P1.git
usage: git remote add [<options>] <name> <url>


    -f, --[no-]fetch          fetch the remote branches
    --[no-]tags              import all tags and associated objects when fetching
                             or do not fetch any tag at all (--no-tags)
    -t, --[no-]track <branch> branch(es) to track
    -m, --[no-]master <branch> master branch
    --[no-]mirror[=(push|fetch)] set up remote as a mirror to push to or fetch from
```

7. Sube los cambios, teniendo en cuenta lo que averiguaste en el punto 3.

Solución

 **unknown** Respuesta del punto 3 de la parte 1 del laboratorio ab9c320 · 7 minutes ago 

 README.md Respuesta del punto 3 de la parte... 7 minutes ago

 README  

Respuestas

Averigua para qué sirve y como se usan estos comandos git add y git commit -m "mensaje"

Solución

El comando git add se utiliza para agregar los archivos modificados al área de preparación (staging area) de Git.

Modo de uso:

Para agregar uno o varios archivos git add

Para agregar todos los archivos git add .

8. Configura el correo en git local de manera correcta.

Solución

En el momento que se hace el *push*, git automaticamente manda la solicitud de ingreso a la cuenta de github.

9. Vuelve a subir los cambios y observa que todo esté bien en el repositorio remoto (en GitHub).

 **unknown** Respuesta del punto 3 de la parte 1 del laboratorio ab9c320 · 7 minutes ago

 README.md Respuesta del punto 3 de la parte... 7 minutes ago

 README  

Respuestas

Averigua para qué sirve y como se usan estos comandos git add y git commit -m "mensaje"

Solución

El comando git add se utiliza para agregar los archivos modificados al área de preparación (staging area) de Git.

Modo de uso:

Para agregar uno o varios archivos git add

Para agregar todos los archivos git add .

Parte Juan Diaz

1. Crea un repositorio localmente.

Solución

```
juan.dgomez@sistemas57 MINGW64 ~/Documents/LAB1_LOCAL
$ git init
Initialized empty Git repository in C:/Users/juan.dgomez/Documents/LAB1_LOCAL/.git/
```

2. Agrega un archivo de ejemplo al repositorio, el README.md puede ser una gran opción.

Solución

```
juan.dgomez@sistemas57 MINGW64 ~/Documents/LAB1_LOCAL (master)
$ echo "# LAB1" >> README.md

juan.dgomez@sistemas57 MINGW64 ~/Documents/LAB1_LOCAL (master)
$ git add README.md
warning: in the working copy of 'README.md', LF will be replaced by CRLF the next time Git touches it
```

3. Abre una cuenta de github, si ya la tienes, enlazala con el correo institucional.

Solución

Emails

juanjodigo@gmail.com – Primary

This email will be used for account-related notifications and can also be used for password resets.

- Not visible in emails
This email will not be used as the 'from' address for web-based Git operations, e.g., edits and merges. We will instead use 81777362+Juan-Jose-D@users.noreply.github.com.
- Receives notifications
This email address is the default used for GitHub notifications, i.e., replies to issues, pull requests, etc.

juan.dgomez@mail.escuelaing.edu.co

- Not visible in emails
This email will not be used as the 'from' address for web-based Git operations, e.g., edits and merges. We will instead use 81777362+Juan-Jose-D@users.noreply.github.com.

Add email address

4. Crea un repositorio en blanco (vacío) e GitHub.

Solución

Create a new repository

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

Required fields are marked with an asterisk (*).

Repository template

No template ▾

Start your repository with a template repository's contents.

Owner *

 Juan-Jose-D ▾

Repository name *

LAB1

✓ LAB1 is available.

Great repository names are short and memorable. Need inspiration? How about [super-duper-spoon](#) ?

Description (optional)

Laboratorio 1 Parte individual, Juan José Díaz



Public

Anyone on the internet can see this repository. You choose who can commit.



Private

You choose who can see and commit to this repository.

Initialize this repository with:

☐ Add a README file

This is where you can write a long description for your project. [Learn more about READMEs.](#)

Add .gitignore

.gitignore template: None ▾

Choose which files not to track from a list of templates. [Learn more about ignoring files.](#)

Choose a license

License: None ▾

A license tells others what they can and can't do with your code. [Learn more about licenses.](#)

 You are creating a private repository in your personal account.

Create repository

5. Sube los cambios, teniendo en cuenta lo que averiguaste en el punto 3.

Solución

```
juan.dgomez@sistemas57 MINGW64 ~/Documents/LAB1_LOCAL (master)
$ git commit -m "Primer Commit"
[master (root-commit) 5d4ecc3] Primer Commit
Committer: unknown <juan.dgomez@labinformis.escuelaing.edu.co>
Your name and email address were configured automatically based
on your username and hostname. Please check that they are accurate.
You can suppress this message by setting them explicitly. Run the
following command and follow the instructions in your editor to edit
your configuration file:
```

```
git config --global --edit
```

After doing this, you may fix the identity used for this commit with:

```
git commit --amend --reset-author
```

```
1 file changed, 1 insertion(+)
create mode 100644 README.md
```

```
juan.dgomez@sistemas57 MINGW64 ~/Documents/LAB1_LOCAL (master)
$ git branch -M main
```

```
juan.dgomez@sistemas57 MINGW64 ~/Documents/LAB1_LOCAL (main)
$ git remote add origin https://github.com/Juan-Jose-D/LAB1.git
error: remote origin already exists.
```

```
juan.dgomez@sistemas57 MINGW64 ~/Documents/LAB1_LOCAL (main)
$ git push -u origin main
info: please complete authentication in your browser...
```

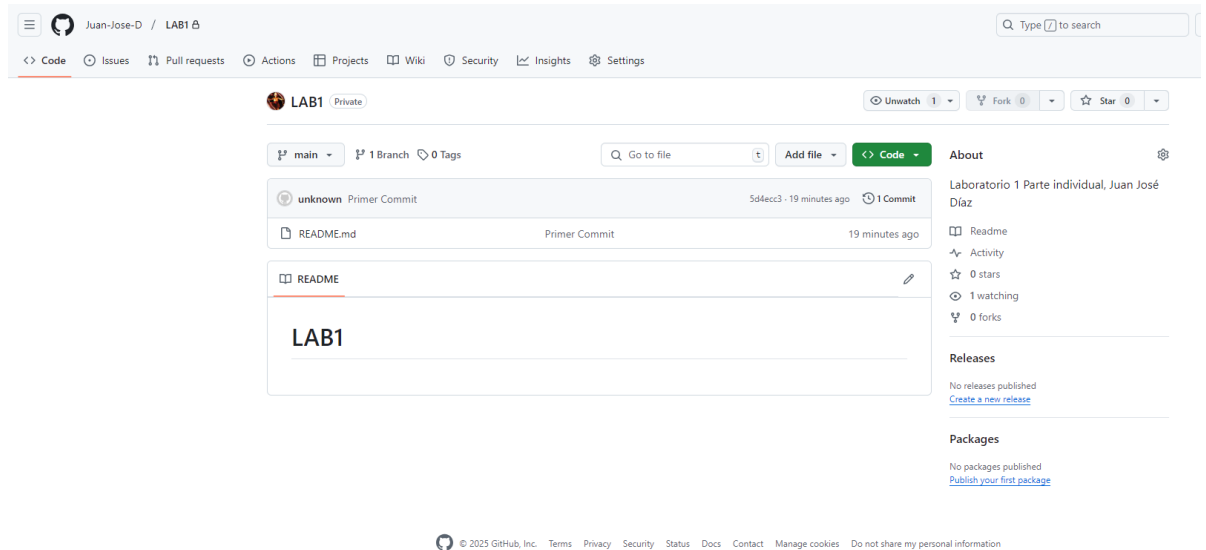
```
juan.dgomez@sistemas57 MINGW64 ~/Documents/LAB1_LOCAL (main)
$ git push -u origin main
info: please complete authentication in your browser...
Enumerating objects: 3, done.
Counting objects: 100% (3/3), done.
Writing objects: 100% (3/3), 235 bytes | 235.00 KiB/s, done.
Total 3 (delta 0), reused 0 (delta 0), pack-reused 0 (from 0)
To https://github.com/Juan-Jose-D/LAB1.git
 * [new branch]      main -> main
branch 'main' set up to track 'origin/main'.
```

```
juan.dgomez@sistemas57 MINGW64 ~/Documents/LAB1_LOCAL (main)
$ git config --global user.email "juanjodigo@gmail.com"
```

```
juan.dgomez@sistemas57 MINGW64 ~/Documents/LAB1_LOCAL (main)
$ git config --global user.email
juanjodigo@gmail.com
```

```
juan.dgomez@sistemas57 MINGW64 ~/Documents/LAB1_LOCAL (main)
$
```

6. Vuelve a subir los cambios y observa que todo esté bien en el repositorio remoto (en GitHub).



PARTE II (Trabajo en parejas)

1. Se escogen los roles para trabajar en equipo, una persona debe escoger ser "Owner" o Propietario del repositorio y la otra "Collaborator" o Colaborador en el repositorio.

Solución:

Owner: David Alfonso Barbosa Gómez Collaborator: Juan José Díaz Gómez

2. El owner agrega al colaborador con permisos de escritura en el repositorio que creó en la parte 1

Invitar colaboradores a un repositorio personal

3. El owner le comparte la url via Teams al colaborador
4. El colaborador acepta la invitación al repositorio.
5. Owner y Colaborador editan el archivo README.md al mismo tiempo e intentan subir los cambios al mismo tiempo.

6. ¿Que sucedió?

Solución:


```


juan.dgomez@sistemas57 MINGW64 ~/Documents/LAB1 (main)
$ git push -u origin main
To https://github.com/DavidBarbosag/LAB1CVDS.git
 ! [rejected]        main -> main (fetch first)
error: failed to push some refs to 'https://github.com/DavidBarbosag/LAB1CVDS.git'
hint: Updates were rejected because the remote contains work that you do not
hint: have locally. This is usually caused by another repository pushing to
hint: the same ref. If you want to integrate the remote changes, use
hint: 'git pull' before pushing again.
hint: See the 'Note about fast-forwards' in 'git push --help' for details.

juan.dgomez@sistemas57 MINGW64 ~/Documents/LAB1 (main)
$ git pull origin main
remote: Enumerating objects: 12, done.
remote: Counting objects: 100% (11/11), done.
remote: Compressing objects: 100% (6/6), done.
remote: Total 8 (delta 0), reused 8 (delta 0), pack-reused 0 (from 0)
Unpacking objects: 100% (8/8), 1.13 KiB | 64.00 KiB/s, done.
From https://github.com/DavidBarbosag/LAB1CVDS
 * branch            main            -> FETCH_HEAD
    d7aee7b..54b3dea  main            -> origin/main
Auto-merging README.md
CONFLICT (content): Merge conflict in README.md
Automatic merge failed; fix conflicts and then commit the result.

```

7. La persona que perdió la competencia de subir los cambios, tiene que resolver los conflictos, cuando haces pull de los cambios, los archivos tienen los símbolos <<<=== y >>> (son normales en la resolución de conflictos), estos conflictos debes resolverlos manualmente. Como resolver Conflictos GitHub.

Commit **8b81001**


 **unknown** committed 2 minutes ago



intento de conflicto

 **main**


Q Filter files... 

 README.md

 **1 file changed** +3 -1 lines changed

```
▼ README.md  
@@ -19,4 +19,6 @@ git add .
19 19
20 20 ## Solución
21 21
22 - se genero un error de conflictos por los cambios realizados simultaneamente
22 + se genero un error de conflictos por los cambios realizados simultaneamente
23 +
24 + jeje saludos
```

Comments 0

 Lock conversation

 Comment

 Subscribe You're not receiving notifications from this thread.

El conflicto se resuelve haciendo un pull del repositorio para luego realizar los cambios al mismo.

8. Volver a repetir un cambio sobre el README.md ambas personas al tiempo para volver a tener conflictos.

Solución:

```
① README.md X
C: > Users > juan.dgomez > Documents > LAB1 > ① README.md > # Respuestas > ## Solución
Accept Current Change | Accept Incoming Change | Accept Both Changes | Compare Changes
1 <<<<<< HEAD (Current Change)
2 # LAB1_CVDS
3
4 ## Estudiantes
5
6 ## Respuestas
7
8 =====
9
10 # Respuestas
11 Averigua para qué sirve y como se usan estos comandos git add y git commit -m "mensaje"
12
13 ## Solución
14
15 El comando git add se utiliza para agregar los archivos modificados al área de preparación (staging area) de Git.
16
17 Modo de uso:
18
19 Para agregar uno o varios archivos
20 git add <Nombre de archivo>
21
22 Para agregar todos los archivos
23 git add .
24 >>>>>> 54b3dea61aaf1ab8d5be8406876099c80155a360 (Incoming Change)
25
```

9. Resuelvan el conflicto con IntelliJ si es posible, Resolver conflictos en IntelliJ.

Solución:

Simplemente se edita en IntelliJ los símbolos y lo que se desea implementar en el repositorio.

PARTE III (Trabajo de a parejas)

1. ¿Hay una mejor forma de trabajar con git para no tener conflictos?
2. ¿Qué es y como funciona el Pull Request?
3. Creen una rama cada uno y suban sus cambios

Solución:

| Default | | | | | |
|----------------------|---------------|--------------|--------|---------|--------------|
| Branch | Updated | Check status | Behind | Ahead | Pull request |
| main | 9 minutes ago | | | Default | ... |

| Your branches | | | | | |
|--------------------------------------|--------------|--------------|--------|-------|--------------|
| Branch | Updated | Check status | Behind | Ahead | Pull request |
| feature/davidBarbosa | 1 minute ago | | 0 | 1 | ... |

| Active branches | | | | | |
|--------------------------------------|----------------|--------------|--------|-------|--------------|
| Branch | Updated | Check status | Behind | Ahead | Pull request |
| feature/davidBarbosa | 1 minute ago | | 0 | 1 | ... |
| Juan_Jose | 11 minutes ago | | 1 | 0 | ... |

[feature/davidBarbosa](#) had recent pushes 3 minutes ago
 [Compare & pull request](#)

4. Tanto owner como colaborador hacen un cambio en el README.md y hacen un Pull Request (PR) a la rama main/master

Parte tres #1

Open

Juan-Jose-D wants to merge 3 commits into [main](#) from [Juan_Jose](#)

Conversation **0**
 Commits **3**
 Checks **0**
 Files changed **1**
+30 -7

Juan-Jose-D commented now Collaborator

Se añadieron las respuestas de la parte tres al readme

unknown added 3 commits 34 minutes ago

- Se añadieron integrantes al readme f706b8e
- Solución preguntas tercera parte c679fcb
- Solución preguntas tercera parte 0861739

This branch has conflicts that must be resolved
 Use the [web editor](#) or the [command line](#) to resolve conflicts.
Conflicting files
 README.md
 [Resolve conflicts](#)

Merge pull request ▼
 You can also [open this in GitHub Desktop](#) or view [command line instructions](#).
 [Try the new merge experience](#)

Reviewers
 No reviews
 Still in progress? [Convert to draft](#)

Assignees
 No one—[assign yourself](#)

Labels
 None yet

Projects
 None yet

Milestone
 No milestone

Development
 Successfully merging this pull request may close these issues.
 None yet

Notifications Customize
[DavidBarbosa](#)

Parte tres #1

Merged Juan-Jose-D merged 4 commits into `main` from `Juan_Jose` now

Conversation 0 Commits 3 Checks 0 Files changed 1

Juan-Jose-D commented 2 minutes ago Collaborator ...

Se añadieron las respuestas de la parte tres al readme

unknown added 3 commits 36 minutes ago

- Se añadieron integrantes al readme f706b8e
- Solucion preguntas tercera parte c679fcb
- Solucion preguntas tercera parte 0861739
- Merge branch 'main' into Juan_Jose Verified b838cb1

Juan-Jose-D merged commit `b8c5b82` into `main` now Revert

Pull request successfully merged and closed Delete branch

You're all set—the `Juan_Jose` branch can be safely deleted.

[Try the new merge experience](#)

5. Teniendo en cuenta la recomendación, mezclen los cambios a la rama main a través de PR con el check/review/approval del otro compañero (Cuando se hace merge se deberían borrar las ramas en github)

After pull requests are merged, you can have head branches deleted automatically.

☒ **Automatically delete head branches** ✓
Deleted branches will still be able to be restored.

Classic branch protections have not been configured
Define branch rules to disable force pushing, prevent branches from being deleted, or require pull requests before merging. Learn more about [repository rules](#) and [protected branches](#).

[Add branch ruleset](#)

[Add classic branch protection rule](#)

Protect matching branches

☒ **Require a pull request before merging**

When enabled, all commits must be made to a non-protected branch and submitted via a pull request before they can be merged into a branch that matches this rule.

☒ **Require approvals**

When enabled, pull requests targeting a matching branch require a number of approvals and no changes requested before they can be merged.

Required number of approvals before merging: 1 ▼

☐ **Dismiss stale pull request approvals when new commits are pushed**

New reviewable commits pushed to a matching branch will dismiss pull request review approvals.

☐ **Require review from Code Owners**

Require an approved review in pull requests including files with a designated code owner.

☐ **Require approval of the most recent reviewable push**

Whether the most recent reviewable push must be approved by someone other than the person who pushed it.

main

Currently applies to 1 branch

Edit

Delete