



TRABAJO PRÁCTICO

UNIDAD 2

TRABAJO PRACTICO DE GIT Y GITHUB

Actividades

1. Contestar las siguiente preguntas:

- **¿Qué es GitHub?**
GitHub es una plataforma en línea que permite almacenar y gestionar proyectos que usan **Git**. Es ideal para el trabajo colaborativo, ya que permite compartir código, gestionar tareas, revisar y aprobar cambios en proyectos de software.
- **¿Cómo crear un repositorio en GitHub?**
 - i. Iniciar sesión en GitHub.
 - ii. Hacer clic en "New" o "Create repository".
 - iii. Asignar un nombre, elegir visibilidad (público o privado), opcionalmente agregar descripción y seleccionar "Initialize with a README" si se desea.
 - iv. Hacer clic en "Create repository".
- **¿Cómo crear una rama en Git?** → `git checkout -b nombre-de-la-rama`
- **¿Cómo cambiar a una rama en Git?** → `git checkout nombre-de-la-rama`
- **¿Cómo fusionar ramas en Git?** → `git checkout rama-principal`
→ `git merge nombre-de-la-rama`
- **¿Cómo crear un commit en Git?** → `git add archivo.js`
→ `git commit -m "Mensaje Descriptivo"`
- **¿Cómo enviar un commit a GitHub?** → `git push`
- **¿Qué es un repositorio remoto?**
Es una versión del repositorio alojada en la nube (por ejemplo, en GitHub), usada para compartir el proyecto con otros colaboradores
- **¿Cómo agregar un repositorio remoto a Git?**
→ `git remote add origin https://github.com/usuario/repositorio.git`
- **¿Cómo empujar cambios a un repositorio remoto?**
→ `git push origin nombre-de-la-rama`
- **¿Cómo tirar de cambios de un repositorio remoto?**
→ `git pull origin nombre-de-la-rama`
- **¿Qué es un fork de repositorio?**
Un fork es una copia completa de un repositorio de GitHub que se crea dentro de tu cuenta. Te permite: probar cambios sin afectar el proyecto original, proponer mejoras enviando los cambios mediante un pull request y, trabajar

con repositorios a los que no tenés permisos directos.

- **¿Cómo crear un fork de un repositorio?**

1. Ingresá a un repositorio público de GitHub (por ejemplo, <https://github.com/otro-usuario/proyecto>).

2. Hacé clic en el botón Fork (esquina superior derecha).



3. Elegí tu cuenta para crear una copia del proyecto en tu propio GitHub.

Luego → **Clonar tu fork en tu PC** (Una vez que tengas tu fork)

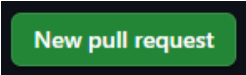
1. Copiá la URL de tu fork (por ejemplo: <https://github.com/tu-usuario/proyecto.git>).

2. En la terminal, ejecutá: `git clone https://github.com/tu-usuario/proyecto.git`

3. Entrá al directorio → `cd proyecto`. Ahora podés trabajar como si fuera tu propio repositorio.

- **¿Cómo enviar una solicitud de extracción (pull request) a un repositorio?**

Desde la página de tu fork en GitHub, ir a la pestaña "Pull Requests" y hacer clic en "New pull request".

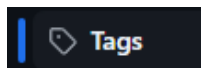
A green rectangular button with the text "New pull request" in white.

- **¿Cómo aceptar una solicitud de extracción?**

Desde el repositorio original, revisar el pull request y hacer clic en "Merge pull request".

- **¿Qué es una etiqueta en Git?**

Es una marca que se utiliza para señalar versiones importantes del proyecto, como versiones de lanzamiento.



- **¿Cómo crear una etiqueta en Git?** → `git tag -a v1.0 -m "Primera versión"`

- **¿Cómo enviar una etiqueta a GitHub?** → `git push origin v1.0`

- **¿Qué es un historial de Git?**

Es el registro de todos los commits realizados en un repositorio.

- **¿Cómo ver el historial de Git?** → `git log`

- **¿Cómo buscar en el historial de Git?** → `git log --grep="palabra clave"`

- **¿Cómo borrar el historial de Git?**

No se puede borrar completamente, pero sí reiniciarlo con:

→ `rm -rf .git`
`git init`

- **¿Qué es un repositorio privado en GitHub?**

Un repositorio que solo puede ser visto por el propietario y los colaboradores autorizados.

- **¿Cómo crear un repositorio privado en GitHub?**

Al crearlo, seleccionar la opción "Private".

- **¿Cómo invitar a alguien a un repositorio privado en GitHub?**

Desde el repositorio, ir a "Settings" > "Collaborators" > Agregar colaborador por su nombre de usuario o correo.

- **¿Qué es un repositorio público en GitHub?**

Es un repositorio accesible para cualquier usuario, que puede clonarlo y verlo.

- **¿Cómo crear un repositorio público en GitHub?**

Al crearlo, seleccionar la opción "Public".

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 *  MaruValletto / **Repository name ***

✔ TUP_P1 is available.

Great repository names are short and memorable. Need inspiration? How about **ubiquitous-giggle** ?

Description (optional)

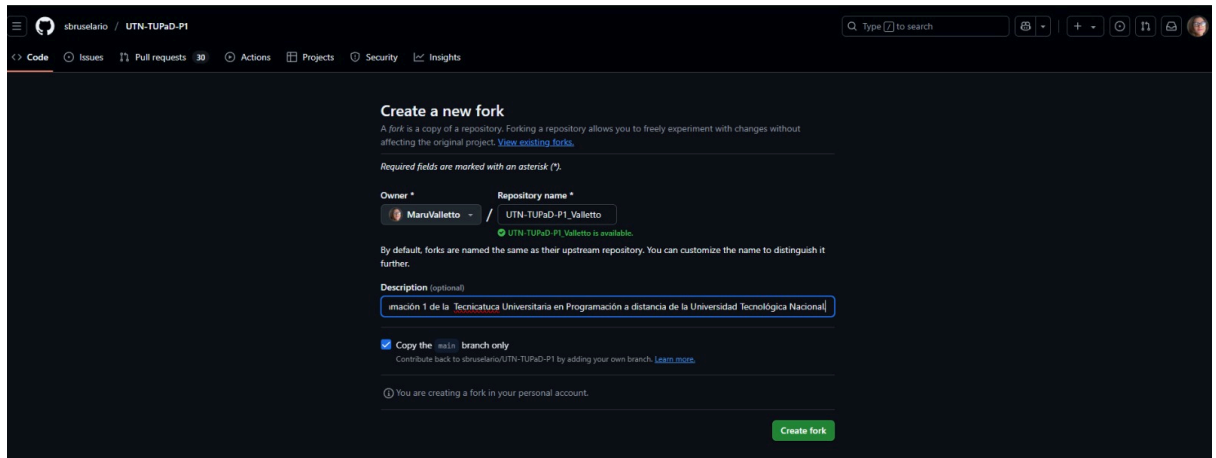
☒  **Public**
Anyone on the internet can see this repository. You choose who can commit.

☐  **Private**
You choose who can see and commit to this repository.

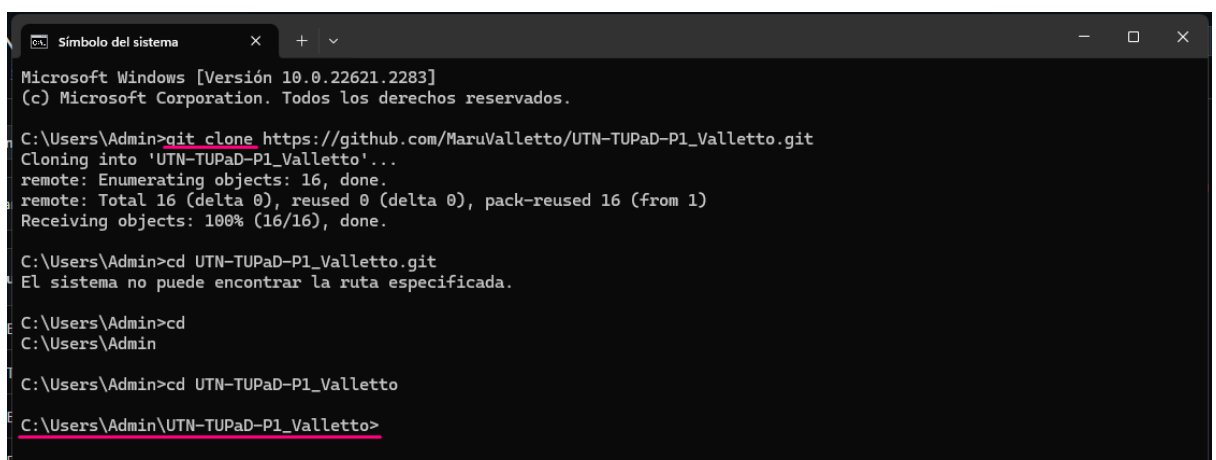
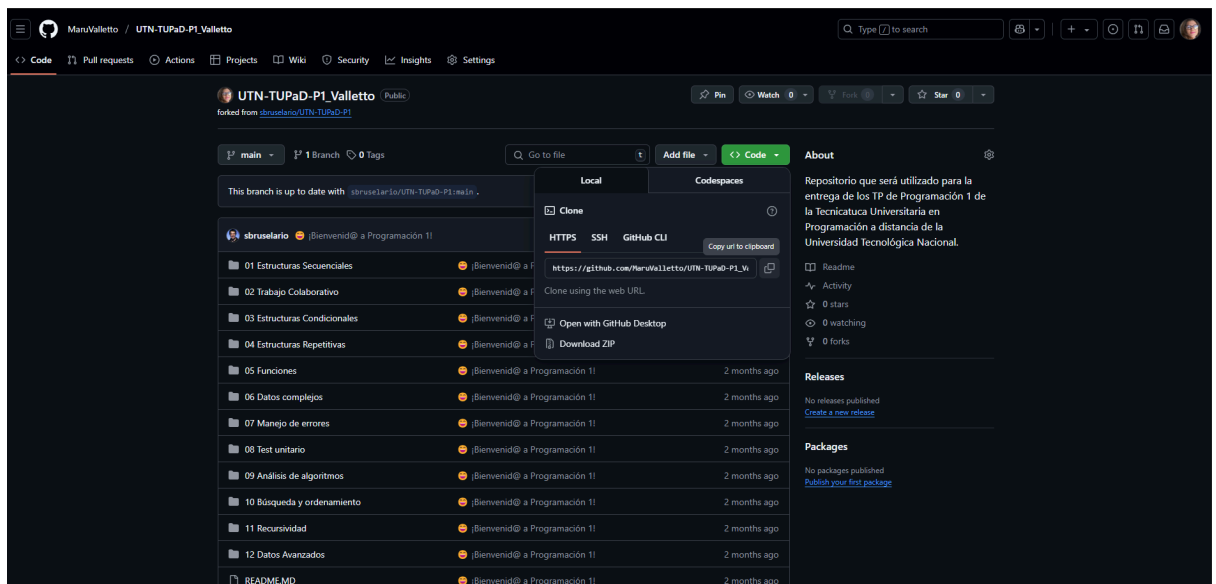
- **¿Cómo compartir un repositorio público en GitHub?**
Copiando la URL del repositorio y compartiéndola con otros.

2. Realizar la siguiente actividad:

- Creación del repositorio mediante Fork

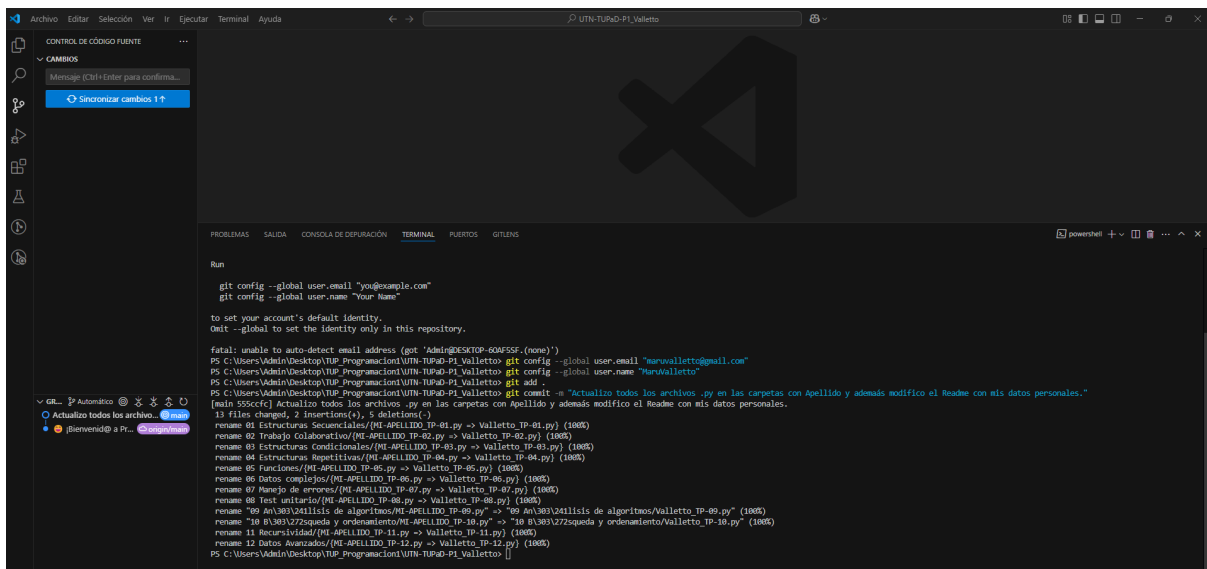


- Clonación del repositorio

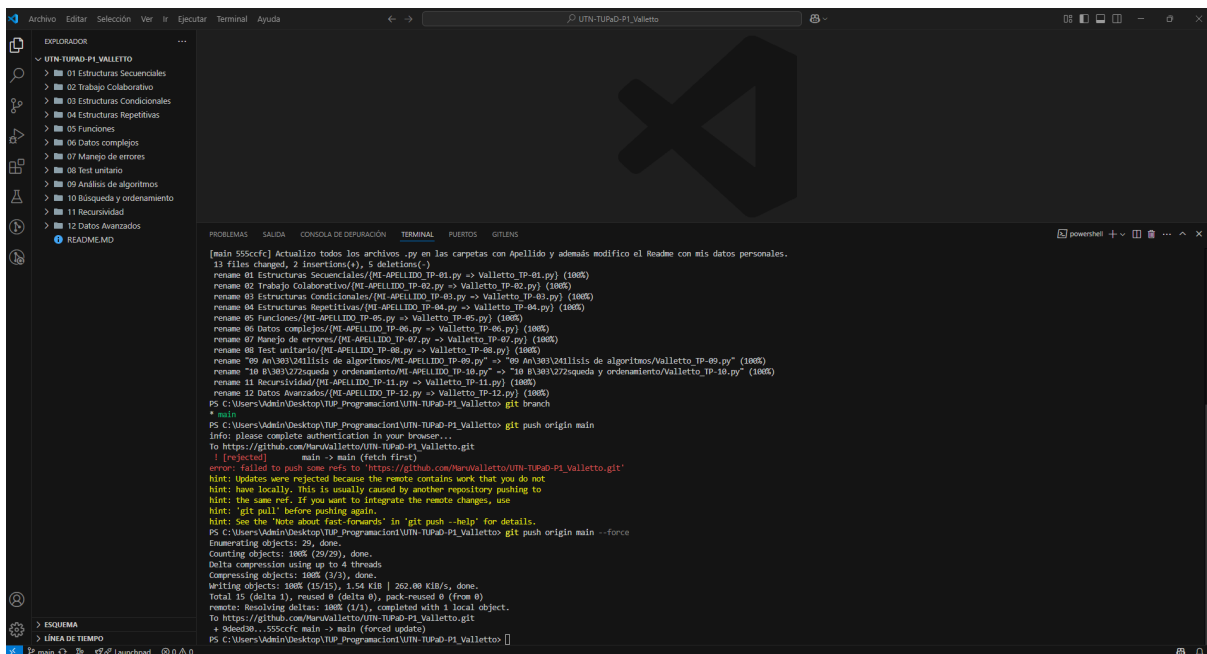


```
C:\Users\Admin\UTN-TUPaD-P1_Valletto>git ls-files
01 Estructuras Secuenciales/MI-APELLIDO_TP-01.py
02 Trabajo Colaborativo/MI-APELLIDO_TP-02.py
03 Estructuras Condicionales/MI-APELLIDO_TP-03.py
04 Estructuras Repetitivas/MI-APELLIDO_TP-04.py
05 Funciones/MI-APELLIDO_TP-05.py
06 Datos complejos/MI-APELLIDO_TP-06.py
07 Manejo de errores/MI-APELLIDO_TP-07.py
08 Test unitario/MI-APELLIDO_TP-08.py
"09 An\303\241lisis de algoritmos/MI-APELLIDO_TP-09.py"
"10 B\303\272squeda y ordenamiento/MI-APELLIDO_TP-10.py"
11 Recursividad/MI-APELLIDO_TP-11.py
12 Datos Avanzados/MI-APELLIDO_TP-12.py
README.MD

C:\Users\Admin\UTN-TUPaD-P1_Valletto>
```

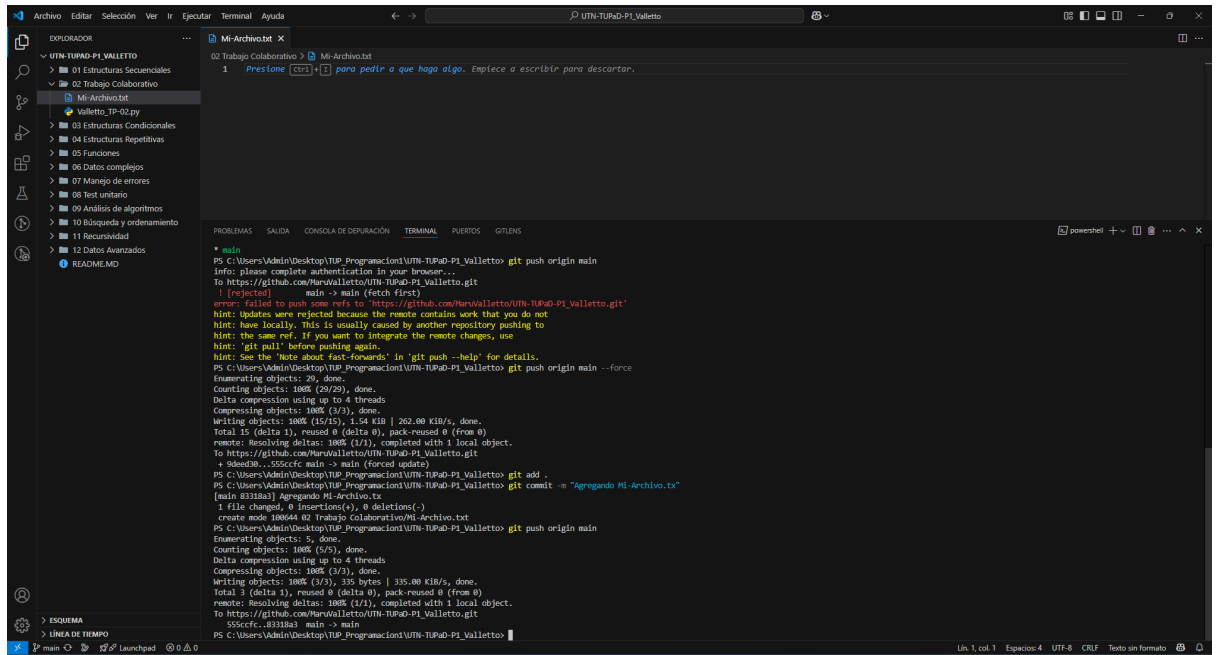


The screenshot shows the VS Code interface with the 'TERMINAL' tab active. The terminal displays the output of a 'git ls-files' command, listing 12 files and a README.MD file. The files are organized into a directory structure: 01 Estructuras Secuenciales, 02 Trabajo Colaborativo, 03 Estructuras Condicionales, 04 Estructuras Repetitivas, 05 Funciones, 06 Datos complejos, 07 Manejo de errores, 08 Test unitario, 09 Análisis de algoritmos, 10 Búsqueda y ordenamiento, 11 Recursividad, and 12 Datos Avanzados. The terminal also shows the output of a 'git config' command, indicating that the global user email is 'marvalletto@gmail.com' and the global user name is 'MarValletto'.



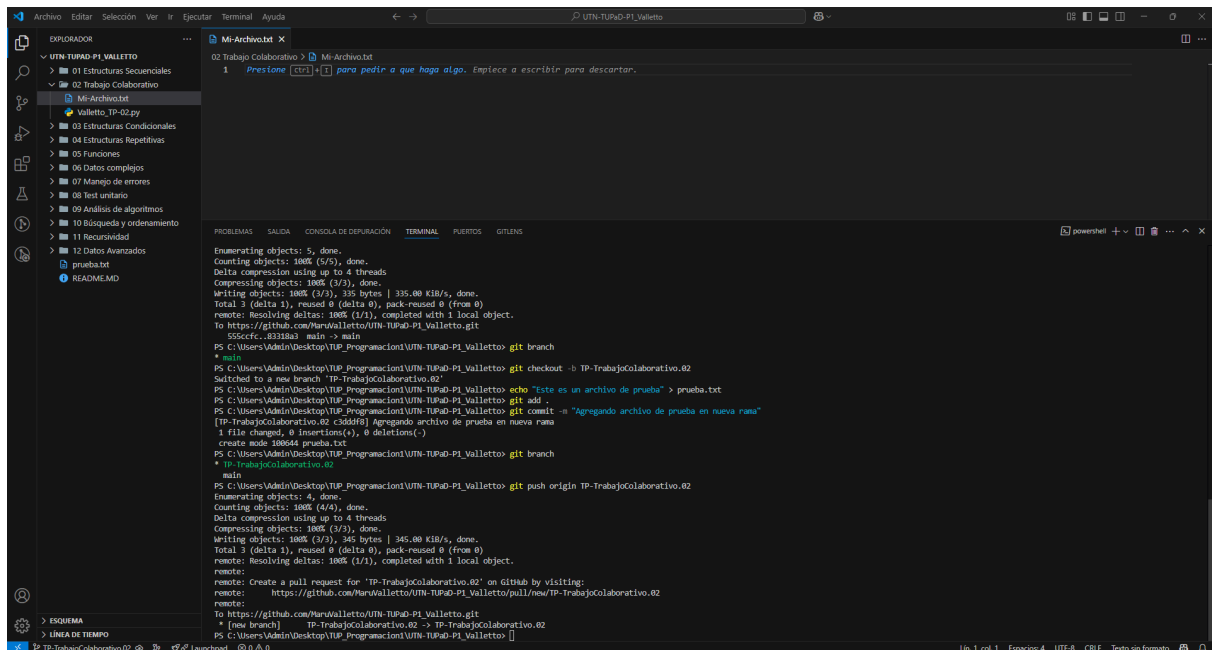
The screenshot shows the VS Code interface with the 'TERMINAL' tab active. The terminal displays the output of a 'git branch' command, showing the current branch is 'main'. It then shows the output of a 'git push origin main' command, which fails with an error: 'error: failed to push some refs to 'https://github.com/MarValletto/UTN-TUPaD-P1_Valletto.git''. The error message indicates that updates were rejected because the remote contains work that you do not have locally. The terminal also shows the output of a 'git pull' command, which successfully updates the local repository to the latest state of the remote.

- Agregando un Archivo



```
PS C:\Users\Admin\Desktop\TP_Programacion\UTN-TUPAD-P1_Valletto> git push origin main
info: please complete authentication in your browser...
To https://github.com/MarValletto/UTN-TUPAD-P1_Valletto.git
! [rejected]    main -> main (fetch first)
error: failed to push some refs to 'https://github.com/MarValletto/UTN-TUPAD-P1_Valletto.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.
PS C:\Users\Admin\Desktop\TP_Programacion\UTN-TUPAD-P1_Valletto> git push origin main --force
Enumerating objects: 29, done.
Counting objects: 100% (29/29), done.
Delta compression using up to 4 threads
Compressing objects: 100% (1/3), done.
Writing objects: 100% (15/15), 1.54 KiB | 262.00 KiB/s, done.
Total 15 (delta 1), reused 0 (delta 0), pack-reused 0 (from 0)
remote: Resolving deltas: 100% (1/3), completed with 1 local object.
To https://github.com/MarValletto/UTN-TUPAD-P1_Valletto.git
+ 90e5d6...555cfcf main -> main (forced update)
PS C:\Users\Admin\Desktop\TP_Programacion\UTN-TUPAD-P1_Valletto> git add .
[main 8233ba2] Agregando MI-Archivo.txt
1 file changed, 0 insertions(+), 0 deletions(-)
create mode 100644 02 Trabajo Colaborativo/MI-Archivo.txt
PS C:\Users\Admin\Desktop\TP_Programacion\UTN-TUPAD-P1_Valletto> git commit -m "Agregando MI-Archivo.txt"
Enumerating objects: 5, done.
Counting objects: 100% (5/5), done.
Delta compression using up to 4 threads
Compressing objects: 100% (1/3), done.
Writing objects: 100% (3/3), 335 bytes | 335.00 KiB/s, done.
Total 3 (delta 1), reused 0 (delta 0), pack-reused 0 (from 0)
remote: Resolving deltas: 100% (1/3), completed with 1 local object.
To https://github.com/MarValletto/UTN-TUPAD-P1_Valletto.git
555cfcf..4333ba2 main -> main
PS C:\Users\Admin\Desktop\TP_Programacion\UTN-TUPAD-P1_Valletto>
```

- Creando Branchs



```
Enumerating objects: 5, done.
Counting objects: 100% (5/5), done.
Delta compression using up to 4 threads
Compressing objects: 100% (1/3), done.
Writing objects: 100% (3/3), 335 bytes | 335.00 KiB/s, done.
Total 3 (delta 1), reused 0 (delta 0), pack-reused 0 (from 0)
remote: Resolving deltas: 100% (1/3), completed with 1 local object.
To https://github.com/MarValletto/UTN-TUPAD-P1_Valletto.git
555cfcf..4333ba2 main -> main
PS C:\Users\Admin\Desktop\TP_Programacion\UTN-TUPAD-P1_Valletto> git branch
* main
PS C:\Users\Admin\Desktop\TP_Programacion\UTN-TUPAD-P1_Valletto> git checkout -b TP-TrabajoColaborativo.02
Switched to a new branch 'TP-TrabajoColaborativo.02'
PS C:\Users\Admin\Desktop\TP_Programacion\UTN-TUPAD-P1_Valletto> echo "Este es un archivo de prueba" > prueba.txt
PS C:\Users\Admin\Desktop\TP_Programacion\UTN-TUPAD-P1_Valletto> git add .
PS C:\Users\Admin\Desktop\TP_Programacion\UTN-TUPAD-P1_Valletto> git commit -m "Agregando archivo de prueba en nueva rama"
[TP-TrabajoColaborativo.02 c3d6df8] Agregando archivo de prueba en nueva rama
1 file changed, 0 insertions(+), 0 deletions(-)
create mode 100644 prueba.txt
PS C:\Users\Admin\Desktop\TP_Programacion\UTN-TUPAD-P1_Valletto> git branch
* TP-TrabajoColaborativo.02
* main
PS C:\Users\Admin\Desktop\TP_Programacion\UTN-TUPAD-P1_Valletto> git push origin TP-TrabajoColaborativo.02
Enumerating objects: 4, done.
Counting objects: 100% (4/4), done.
Delta compression using up to 4 threads
Compressing objects: 100% (3/3), done.
Writing objects: 100% (3/3), 345 bytes | 345.00 KiB/s, done.
Total 3 (delta 1), reused 0 (delta 0), pack-reused 0 (from 0)
remote: Resolving deltas: 100% (1/3), completed with 1 local object.
remote: Create a pull request for 'TP-TrabajoColaborativo.02' on GitHub by visiting:
remote:   https://github.com/MarValletto/UTN-TUPAD-P1_Valletto/pull/new/TP-TrabajoColaborativo.02
To https://github.com/MarValletto/UTN-TUPAD-P1_Valletto.git
* [new branch] TP-TrabajoColaborativo.02 -> TP-TrabajoColaborativo.02
PS C:\Users\Admin\Desktop\TP_Programacion\UTN-TUPAD-P1_Valletto>
```


The screenshot shows a GitHub repository page for 'UTN-TUPaD-P1_Valletto'. The repository is a fork of 'abueladon/UTN-TUPaD-P1'. The main branch is 'main', which is 2 commits ahead of the upstream 'main'. The repository has 0 branches and 0 tags. The commit history shows a series of commits by 'ManuValletto' updating various files in the repository. The commit messages are: 'Agregando Mi-Archivo.txt', 'Actualizo todos los archivos .py en las carpetas con Apellido', 'Agregando Mi-Archivo.txt', 'Actualizo todos los archivos .py en las carpetas con Apellido', 'Actualizo todos los archivos .py en las carpetas con Apellido', 'Actualizo todos los archivos .py en las carpetas con Apellido', 'Actualizo todos los archivos .py en las carpetas con Apellido', 'Actualizo todos los archivos .py en las carpetas con Apellido', 'Actualizo todos los archivos .py en las carpetas con Apellido', 'Actualizo todos los archivos .py en las carpetas con Apellido', and 'Actualizo todos los archivos .py en las carpetas con Apellido'. The repository has 0 stars, 0 forks, and 0 releases. The 'About' section describes the repository as a tool for the delivery of the TP of Programming 1 of the National Technological University. The 'Releases' section shows no releases published. The 'Packages' section shows no packages published.

ManuValletto / UTN-TUPaD-P1_Valletto

UTN-TUPaD-P1_Valletto Public

forked from abueladon/UTN-TUPaD-P1

TP-TrabajoColaborativo.02 had recent pushes 1 minute ago [Compare & pull request](#)

main 2 Branches 0 Tags

Go to file Add file <> Code

This branch is 2 commits ahead of abueladon/UTN-TUPaD-P1:main. [Contribute](#) [Sync fork](#)

ManuValletto Agregando Mi-Archivo.txt 83316d · 13 minutes ago 1 Commits

01 Estructuras Secuenciales	Actualizo todos los archivos .py en las carpetas con Apellido	32 minutes ago
02 Trabajo Colaborativo	Agregando Mi-Archivo.txt	13 minutes ago
03 Estructuras Condicionales	Actualizo todos los archivos .py en las carpetas con Apellido	32 minutes ago
04 Estructuras Repetitivas	Actualizo todos los archivos .py en las carpetas con Apellido	32 minutes ago
05 Funciones	Actualizo todos los archivos .py en las carpetas con Apellido	32 minutes ago
06 Datos complejos	Actualizo todos los archivos .py en las carpetas con Apellido	32 minutes ago
07 Manejo de errores	Actualizo todos los archivos .py en las carpetas con Apellido	32 minutes ago
08 Test unitario	Actualizo todos los archivos .py en las carpetas con Apellido	32 minutes ago
09 Análisis de algoritmos	Actualizo todos los archivos .py en las carpetas con Apellido	32 minutes ago
10 Búsqueda y ordenamiento	Actualizo todos los archivos .py en las carpetas con Apellido	32 minutes ago
11 Recursividad	Actualizo todos los archivos .py en las carpetas con Apellido	32 minutes ago

About

Repositorio que será utilizado para la entrega de los TP de Programación 1 de la Tecnicatura Universitaria en Programación a distancia de la Universidad Tecnológica Nacional.

Readme Activity 0 stars 0 watching 0 forks

Releases

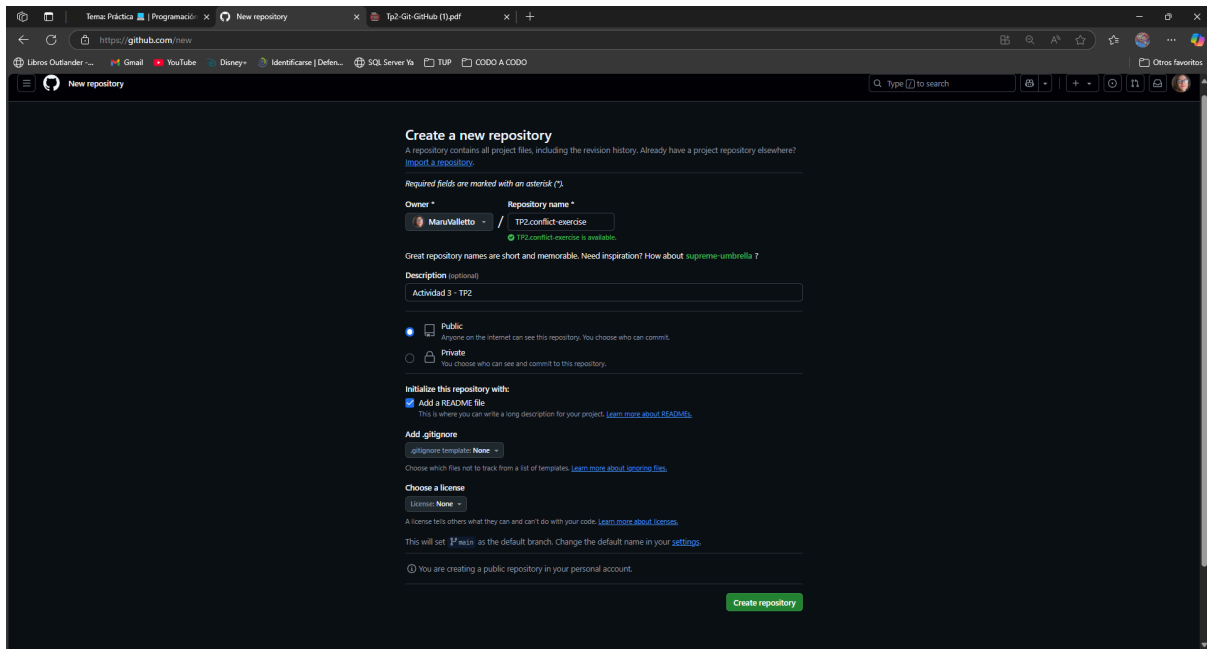
No releases published [Create a new release](#)

Packages

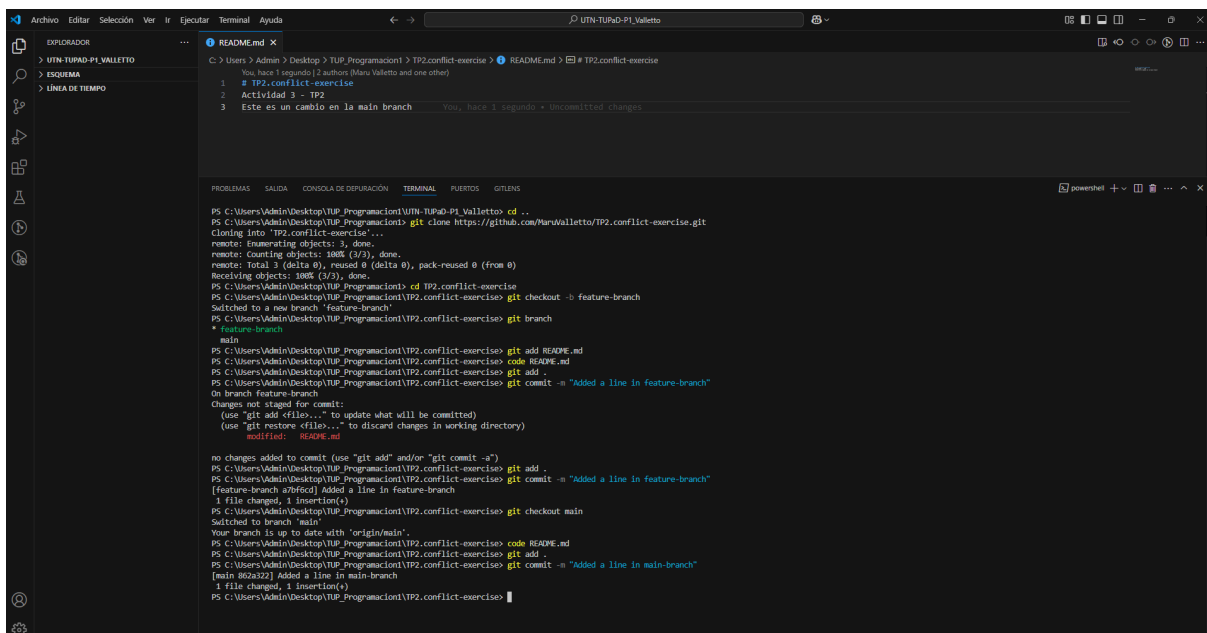
No packages published [Publish your first package](#)

3) Realizar la siguiente actividad:

- Paso 1: Crear un repositorio en GitHub



- Paso 2: Clonar el repositorio a tu máquina local
- Paso 3: Crear una nueva rama y editar un archivo
- Paso 4: Volver a la rama principal y editar el mismo archivo



- Paso 5: Hacer un merge y generar un conflicto
- Paso 6: Resolver el conflicto
- Paso 7: Subir los cambios a GitHub

```
PS C:\Users\Admin\Desktop\TP2.conflict-exercise> git merge feature-branch
Auto-merging README.md
CONFLICT (content): Merge conflict in README.md
Automatic merge failed; fix conflicts and then commit the result.
PS C:\Users\Admin\Desktop\TP2.conflict-exercise> git add .
PS C:\Users\Admin\Desktop\TP2.conflict-exercise> git commit -m "Resolved merge conflict"
[main b78b2a1] Resolved merge conflict
1 file changed, 1 insertion(+), 1 deletion(-)
PS C:\Users\Admin\Desktop\TP2.conflict-exercise> git push origin main
Enumerating objects: 9, done.
Counting objects: 100% (9/9), done.
Delta compression using up to 4 threads
Compressing objects: 100% (5/5), done.
Writing objects: 100% (2/2), 659 bytes | 659.00 KiB/s, done.
Total 7 (delta 3), reused 0 (delta 0), pack-reused 0 (from 0)
remote: Resolving deltas: 100% (3/3), done.
To https://github.com/MaruValletto/TP2.conflict-exercise.git
42ba6e0..578b2a1 main -> main
PS C:\Users\Admin\Desktop\TP2.conflict-exercise> git push origin feature-branch
Total 0 (delta 0), reused 0 (delta 0), pack-reused 0 (from 0)
remote:
remote: Create a pull request for 'feature-branch' on GitHub by visiting:
remote:   https://github.com/MaruValletto/TP2.conflict-exercise/pull/new/feature-branch
remote:
To https://github.com/MaruValletto/TP2.conflict-exercise.git
* [new branch]   feature-branch -> feature-branch
PS C:\Users\Admin\Desktop\TP2.conflict-exercise> |
```

- Paso 8: Verificar en GitHub

