

# INNOW Technology Fest 2024

Colegio Científico del Atlántico

## ENTREGABLE #1

### CONFIGURACIÓN DEL GIT Y GITHUB



Trabajo dirigido por:

Justin Geovany Lacayo Picado

Martin Francisco Rojas Barrios

Grevanny Mahia Quintana Manning

Tutor:

Elkira Francis Hernández

16/6/2024

# Índice

<b>1</b>	<b>Cuestionario</b>	<b>2</b>
1.1	¿Qué es Git y para qué se utiliza? . . . . .	2
1.2	¿Cuál es la diferencia entre Git y GitHub? . . . . .	2
1.3	¿Qué es un branch? . . . . .	2
1.4	¿Qué comandos usamos para iniciar un repositorio de manera local? . . . . .	2
1.5	¿Qué comando usamos para incluir los cambios en el repositorio local? . . . . .	2
1.6	¿Cuál es la diferencia entre <code>git add</code> y <code>git commit</code> ? . . . . .	3
1.7	¿Cómo puedes ver el historial de commits en Git? . . . . .	3
1.8	¿Qué comando usamos para crear una rama? . . . . .	3
1.9	¿Qué comando se usa para cambiar a una rama diferente en Git? . . . . .	3
1.10	¿Qué comandos usamos para clonar un repositorio de GitHub? . . . . .	3
1.11	¿Qué comando usamos para enviar nuestros cambios al repositorio remoto? . . . . .	4
1.12	¿Qué debemos usar para unir nuestros cambios de dos ramas en el repositorio remoto? . . . . .	4
<b>2</b>	<b>Glosario de Comandos</b>	<b>4</b>
<b>3</b>	<b>Evidencia</b>	<b>6</b>
3.1	Primer método: A través del bash git . . . . .	6
3.1.1	Paso 1: Descargar Git y realizar la configuración inicial en su máquina local. . . . .	6
3.1.2	Paso 2: Clonar el repositorio remoto. . . . .	7
3.1.3	Paso 3: Correr el comando: <code>git remote -v</code> . . . . .	8
3.2	Segundo método: A través de Visual Studio Code . . . . .	9
3.2.1	Clonar el repositorio remoto. . . . .	9

## 1. Cuestionario

### 1.1. ¿Qué es Git y para qué se utiliza?

Git es un sistema de control de versiones distribuido utilizado para gestionar y hacer seguimiento de los cambios en el código fuente de proyectos de software. Permite a múltiples desarrolladores trabajar simultáneamente en un proyecto, mantener un historial de los cambios realizados y revertir a versiones anteriores si es necesario.

### 1.2. ¿Cuál es la diferencia entre Git y GitHub?

Git es una herramienta de control de versiones que se utiliza localmente en la máquina para gestionar el historial de tu código. GitHub, por otro lado, es una plataforma de alojamiento de repositorios Git basada en la nube que facilita la colaboración entre desarrolladores y proporciona una interfaz web para gestionar repositorios Git, seguimiento de issues, y más.

### 1.3. ¿Qué es un branch?

Un branch (rama) en Git es una versión paralela del repositorio. Permite a los desarrolladores trabajar en diferentes características, correcciones de errores o experimentos de forma aislada del branch principal (generalmente llamado main o master).

### 1.4. ¿Qué comandos usamos para iniciar un repositorio de manera local?

Para iniciar un repositorio de manera local, usamos el siguiente comando:

```
git init
```

### 1.5. ¿Qué comando usamos para incluir los cambios en el repositorio local?

Para incluir los cambios en el repositorio local, usamos el siguiente comando:

```
git add .
```

### 1.6. ¿Cuál es la diferencia entre `git add` y `git commit`?

- `git add` se utiliza para añadir los cambios (nuevos archivos, modificaciones, eliminaciones) al área de preparación (*staging area*).
- `git commit` se utiliza para guardar esos cambios en el historial del repositorio, haciendo una "instantánea" de los archivos tal como están en ese momento en el área de preparación.

### 1.7. ¿Cómo puedes ver el historial de commits en Git?

Para ver el historial de commits en Git, usamos el siguiente comando:

```
git log
```

### 1.8. ¿Qué comando usamos para crear una rama?

Para crear una rama, usamos el siguiente comando:

```
git branch nombre_de_la_rama
```

### 1.9. ¿Qué comando se usa para cambiar a una rama diferente en Git?

Para cambiar a una rama diferente en Git, usamos el siguiente comando:

```
git checkout nombre_de_la_rama
```

A partir de Git 2.23, se recomienda usar:

```
git switch nombre_de_la_rama
```

### 1.10. ¿Qué comandos usamos para clonar un repositorio de GitHub?

Para clonar un repositorio de GitHub, usamos el siguiente comando:

```
git clone URL_del_repositorio
```

### 1.11. ¿Qué comando usamos para enviar nuestros cambios al repositorio remoto?

Para enviar nuestros cambios al repositorio remoto, usamos el siguiente comando:

```
git push origin nombre_de_la_rama
```

### 1.12. ¿Qué debemos usar para unir nuestros cambios de dos ramas en el repositorio remoto?

Para unir los cambios de dos ramas, generalmente se usa una *Pull Request* (PR) en plataformas como GitHub. Los comandos específicos que se usan localmente son:

```
git checkout rama_destino
```

```
git merge rama_fuente
```

Pero en el contexto de GitHub, se suele hacer mediante la interfaz web creando una *Pull Request* desde la rama\_fuente hacia la rama\_destino.

## 2. Glosario de Comandos

- **git init:** Inicializa un nuevo repositorio Git en el directorio actual, creando un subdirectorio `.git` que contiene todos los metadatos del repositorio.
- **git add .:** Añade todos los cambios (nuevos archivos, modificaciones, eliminaciones) del directorio actual al área de preparación (*staging area*).
- **git commit:** Guarda los cambios añadidos al área de preparación en el historial del repositorio, creando una nueva "instantánea" de los archivos.
- **git log:** Muestra el historial de commits del repositorio, incluyendo los identificadores de commits, mensajes de commit, y metadatos como el autor y la fecha.
- **git branch nombre\_de\_la\_rama:** Crea una nueva rama con el nombre especificado.

- `git checkout nombre_de_la_rama`: Cambia a la rama especificada. A partir de Git 2.23, se recomienda usar `git switch nombre_de_la_rama`.
- `git switch nombre_de_la_rama`: Cambia a la rama especificada. Es una alternativa moderna a `git checkout` para cambiar de rama.
- `git clone URL_del_repositorio`: Clona un repositorio remoto desde la URL especificada a un nuevo directorio local.
- `git push origin nombre_de_la_rama`: Envía los commits de la rama local especificada al repositorio remoto.
- `git checkout rama_destino`: Cambia a la rama de destino antes de realizar un merge.
- `git merge rama_fuente`: Une los cambios de la rama fuente en la rama actual (que generalmente es la rama de destino).

### 3. Evidencia

#### 3.1. Primer método: A través del bash git

##### 3.1.1. Paso 1: Descargar Git y realizar la configuración inicial en su máquina local.

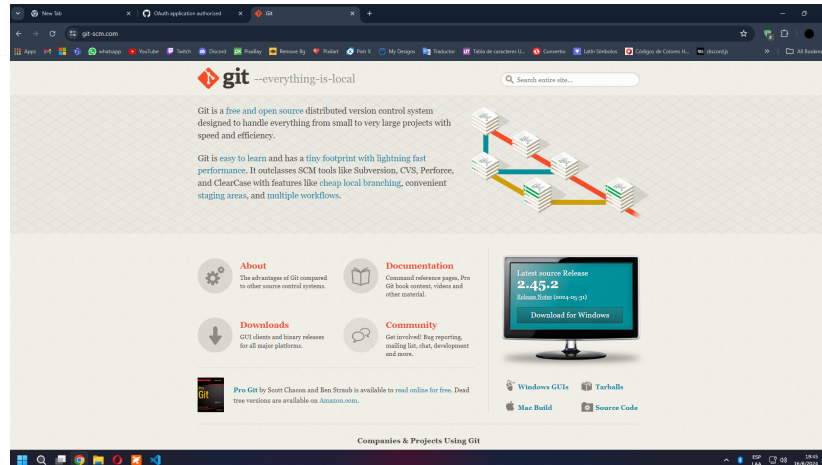


Figura 1: Primer paso

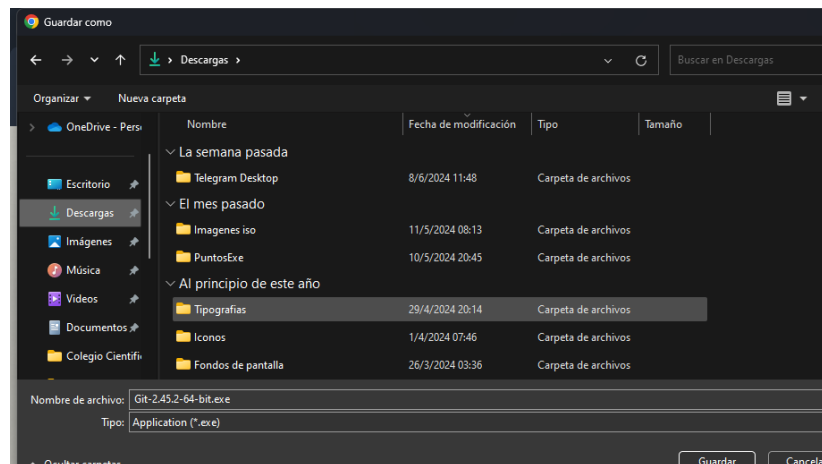


Figura 2: Segundo paso

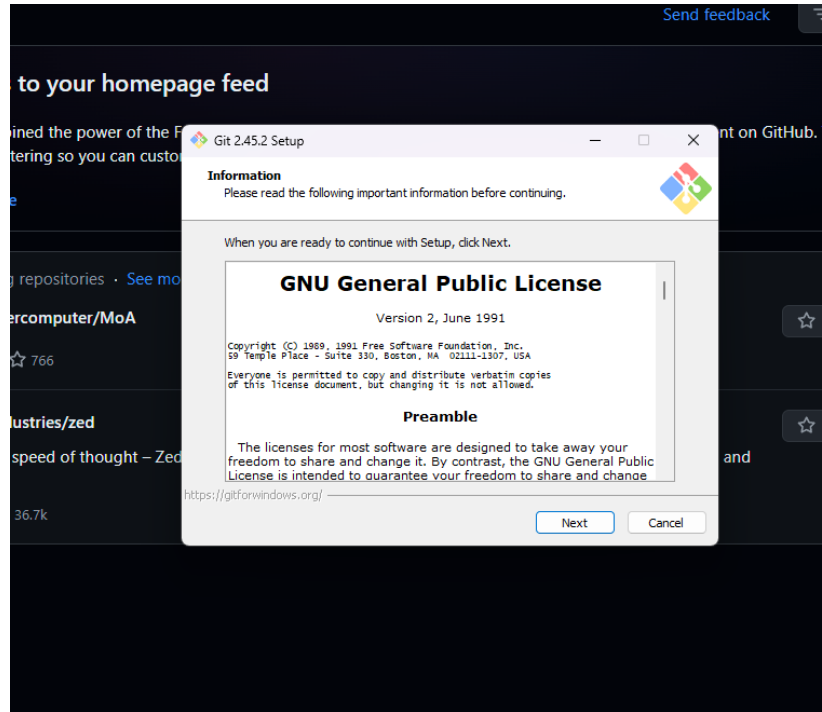


Figura 3: Tercer paso

### 3.1.2. Paso 2: Clonar el repositorio remoto.

```
Marfr@JARVIS MINGW64 ~/Downloads/INNOW
$ git clone https://github.com/innow-tech-fest-2024/repo-innow-colegio-cientifico-del-atlantico.git
Cloning into 'repo-innow-colegio-cientifico-del-atlantico'...
remote: Enumerating objects: 7, done.
remote: Counting objects: 100% (7/7), done.
remote: Compressing objects: 100% (5/5), done.
remote: Total 7 (delta 0), reused 7 (delta 0), pack-reused 0
Receiving objects: 100% (7/7), done.

Marfr@JARVIS MINGW64 ~/Downloads/INNOW
$ ^C

Marfr@JARVIS MINGW64 ~/Downloads/INNOW
$ cd repo-innow-colegio-cientifico-del-atlantico

Marfr@JARVIS MINGW64 ~/Downloads/INNOW/repo-innow-colegio-cientifico-del-atlantico (main)
$ |
```

Figura 4: Primer paso



```
Marfr@JARVIS MINGW64 ~/Downloads/INNOW
$ git clone https://github.com/innow-tech-fest-2024/repo-innow-colegio-cientifico-del-atlantico.git
Cloning into 'repo-innow-colegio-cientifico-del-atlantico'...
remote: Enumerating objects: 7, done.
remote: Counting objects: 100% (7/7), done.
remote: Compressing objects: 100% (5/5), done.
remote: Total 7 (delta 0), reused 7 (delta 0), pack-reused 0
Receiving objects: 100% (7/7), done.

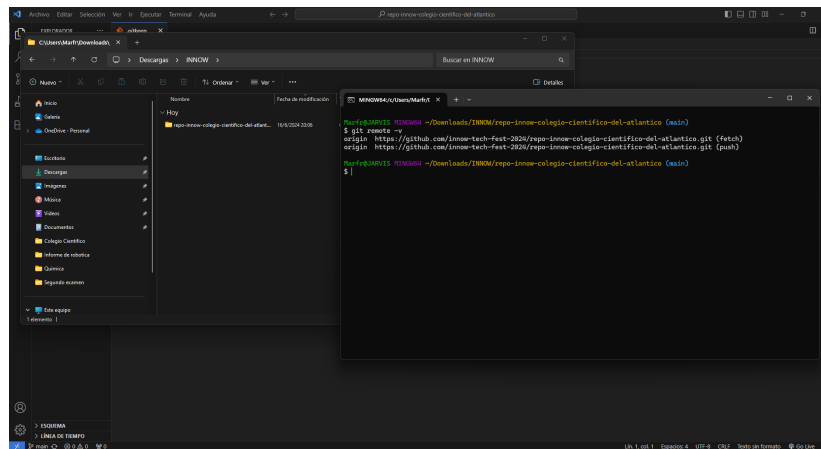
Marfr@JARVIS MINGW64 ~/Downloads/INNOW
$ ^C

Marfr@JARVIS MINGW64 ~/Downloads/INNOW
$ cd repo-innow-colegio-cientifico-del-atlantico

Marfr@JARVIS MINGW64 ~/Downloads/INNOW/repo-innow-colegio-cientifico-del-atlantico (main)
$ |
```

Figura 5: Segundo paso

### 3.1.3. Paso 3: Correr el comando: git remote -v



```
Marfr@JARVIS MINGW64 ~/Downloads/INNOW/repo-innow-colegio-cientifico-del-atlantico (main)
$ git remote -v
origin https://github.com/innow-tech-fest-2024/repo-innow-colegio-cientifico-del-atlantico.git (push)
origin https://github.com/innow-tech-fest-2024/repo-innow-colegio-cientifico-del-atlantico.git (push)

Marfr@JARVIS MINGW64 ~/Downloads/INNOW/repo-innow-colegio-cientifico-del-atlantico (main)
$ |
```

Figura 6: Único paso

## 3.2. Segundo método: A través de Visual Studio Code

### 3.2.1. Clonar el repositorio remoto.

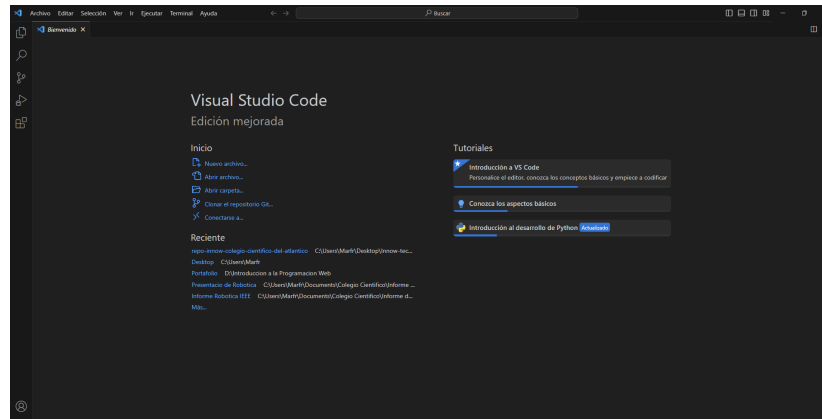


Figura 7: Primer paso

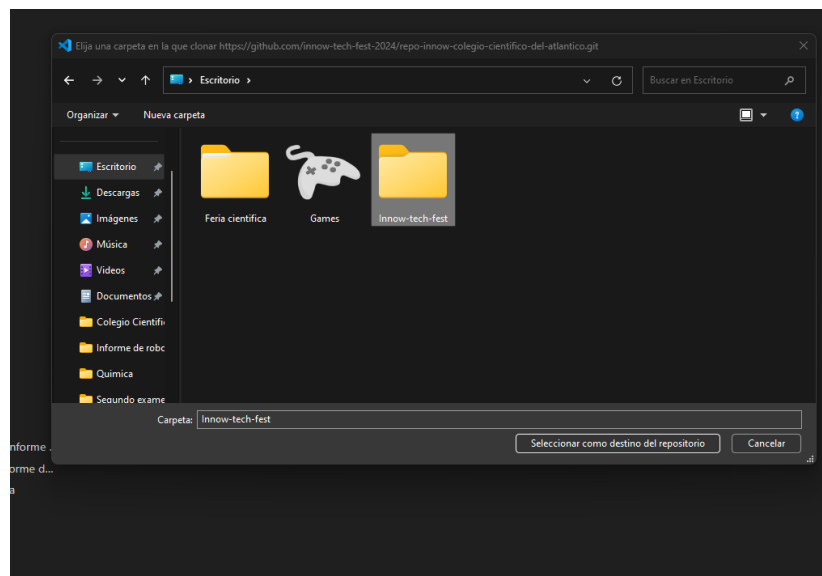


Figura 8: Segundo paso

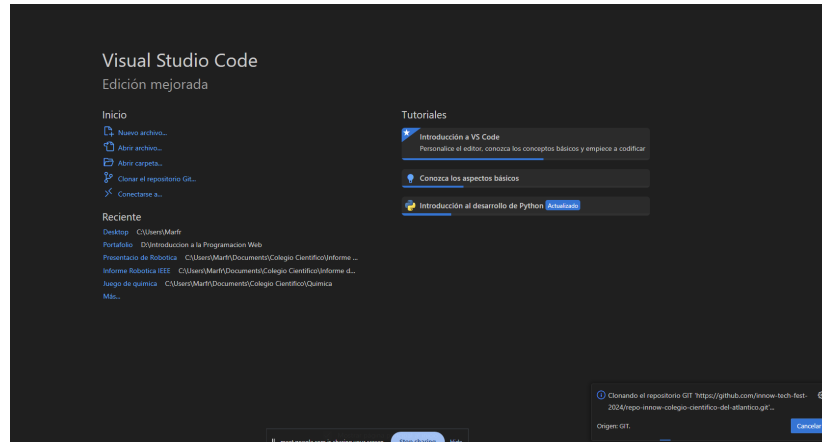


Figura 9: Tercer paso