

## Procedimiento Operativo Estándar

### Standard Operating Procedure (SOP)

## Crear una rama (Branch)

### Propósito del documento (Document Purpose)

El propósito de este Procedimiento Operativo Estándar (SOP) es proporcionar una guía clara y detallada sobre el proceso para crear un branch (rama) en GitHub. Este SOP está diseñado para garantizar que los usuarios puedan gestionar de manera eficiente y organizada el desarrollo de código en proyectos colaborativos, facilitando la implementación de nuevas características, corrección de errores y experimentación sin afectar el código principal.

### Escenario donde aplica SOP (Document Scope)

El siguiente SOP se aplica para el grupo de la materia de Sistemas de Manufactura, con el fin de usar la plataforma Github par la gestión de código fuente y colaboración de proyectos.

### Definiciones y acrónimos (Definitions & Acronyms)

- **SOP (Procedimiento Operativo Estándar):** Documento que describe los pasos y procedimientos necesarios para realizar una tarea específica de manera uniforme y eficiente.
- **Branch (Rama):** Es una línea de desarrollo independiente dentro de un repositorio en Git. Permite a los usuarios trabajar en diferentes características, correcciones o experimentos en paralelo sin afectar el código en la rama principal (generalmente `main` o `master`).
- **Git:** Es un sistema de control de versiones distribuido que gestiona el código fuente y su historial. Permite a los desarrolladores colaborar en el desarrollo de software y realizar un seguimiento de los cambios realizados en el código.
- **GitHub:** Es una plataforma en línea que aloja repositorios Git y proporciona herramientas para la colaboración en el desarrollo de software, incluyendo la gestión de ramas, pull requests y la integración continua.
- **Main (o Master) Branch:** Es la rama principal en un repositorio de GitHub que contiene el código de producción estable. Es la rama base desde la cual se crean otras ramas y a la que eventualmente se fusionan los cambios.
- **Commit:** Es una acción en Git que guarda los cambios realizados en el código en un branch. Cada commit tiene un identificador único y un mensaje que describe los cambios realizados.
- **Merge:** Es el proceso de combinar los cambios de una rama en otra, generalmente para integrar los cambios realizados en una rama de desarrollo a la rama principal.



- **Pull Request (PR):** Es una solicitud para integrar cambios de una rama a otra dentro de un repositorio en GitHub. Permite la revisión y discusión de los cambios antes de su integración.
- **Repositorio (Repository):** Es un contenedor de archivos y carpetas que contiene el código fuente y su historial de versiones. Un repositorio en GitHub puede ser público o privado
- **URL (Uniform Resource Locator):** Dirección web utilizada para acceder a recursos en línea, como un repositorio en GitHub.
- **CI/CD (Continuous Integration/Continuous Deployment):** Conjunto de prácticas de desarrollo de software que permite integrar y desplegar cambios en el código de manera continua y automatizada.

## Procedimiento a seguir

1. Al haber clonado el repositorio (SOP 1), en el programa “Visual Studio Core, en la Terminal, ingresar el comando “ls SistemasDeManufactura-2024” y presionar “Enter”

```
Luis Hernán@DESKTOP-6GVBBRM MINGW64 ~/Desktop/Itq
● $ ls SistemasDeManufactura-2024
```

2. Ingresar el comando “cd SistemasDeManufactura-2024” y presionar “Enter”

```
Luis Hernán@DESKTOP-6GVBBRM MINGW64 ~/Desktop/Itq
● $ cd SistemasDeManufactura-2024
```

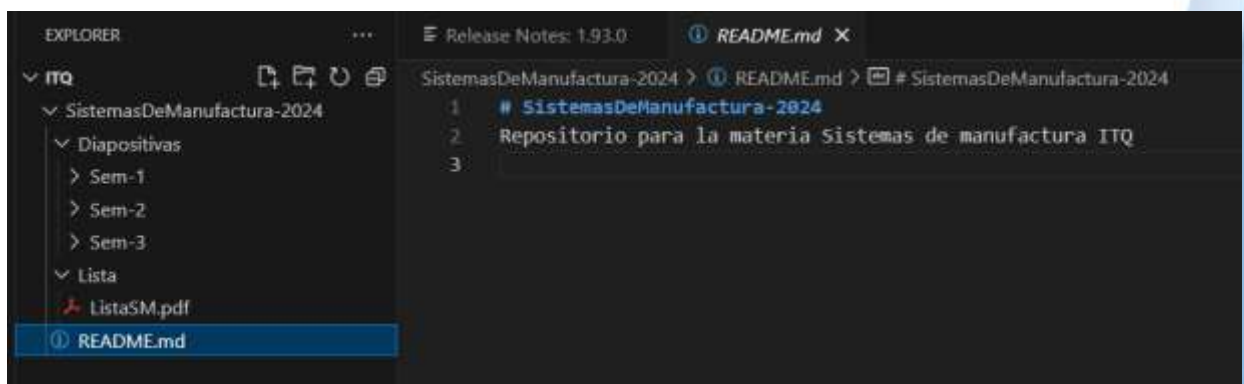
3. Ingresar el comando “git checkout –b Nombre del alumno” y presionar “Enter”

```
Luis Hernán@DESKTOP-6GVBBRM MINGW64 ~/Desktop/Itq/SistemasDeManufactura-2024 (main)
● $ git checkout -b LuisHernanRojasMarin
Switched to a new branch 'LuisHernanRojasMarin'
```

4. Ingresar el comando “git branch”, al presionar “Enter” tu branch se crea automáticamente, con el nombre del alumno.

```
Luis Hernán@DESKTOP-6GVBBRM MINGW64 ~/Desktop/Itq/SistemasDeManufactura-2024 (LuisHernanRojasMarin)
● $ git branch
* LuisHernanRojasMarin
  main
```

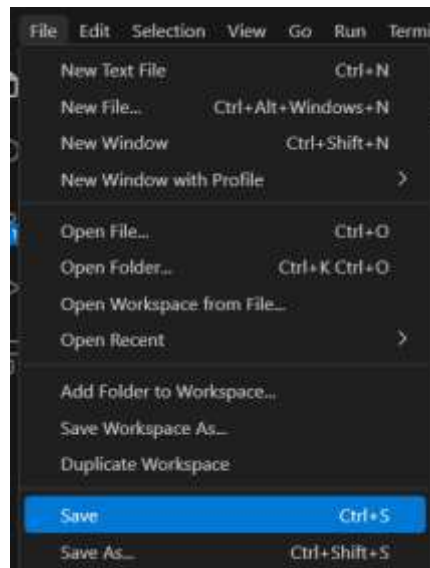
5. En el apartado “EXPLORER”, situarse en el archivo “README.md” y dar click para abrirlo.



6. Automáticamente te aparece una ventana con el título del repositorio; escribe “### Nombre del branch”



7. Situarse y dar click en la pestaña “Archivo/File”, y seleccionar la opción “Save/Guardar”



8. En la Terminal, ingresar el comando git commit –am “Hola Github, este es mi nuevo branch” y presionar “Enter”

```
Luis Hernán@DESKTOP-6GVBBRM MINGW64 ~/Desktop/Itq/SistemasDeManufactura-2024 (LuisHernanRojasMarin)
$ git commit -am "Hola Github, este es mi nuevo branch"
[LuisHernanRojasMarin 4e0adf1] Hola Github, este es mi nuevo branch
1 file changed, 2 insertions(+)
```

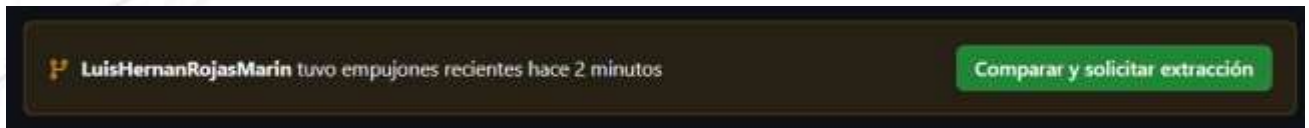
9. Ingresar el comando “git push –u origin (Nombre del branch)” y presionar “Enter”

```
Luis Hernán@DESKTOP-6GVBBRM MINGW64 ~/Desktop/Itq/SistemasDeManufactura-2024 (LuisHernanRojasMarin)
$ git push -u origin LuisHernanRojasMarin
Enumerating objects: 5, done.
Counting objects: 100% (5/5), done.
Delta compression using up to 8 threads
Compressing objects: 100% (3/3), done.
Writing objects: 100% (3/3), 332 bytes | 332.00 KiB/s, done.
Total 3 (delta 2), reused 0 (delta 0), pack-reused 0 (from 0)
remote: Resolving deltas: 100% (2/2), completed with 2 local objects.
remote:
remote: Create a pull request for 'LuisHernanRojasMarin' on Github by visiting:
remote: https://github.com/DriAlbertoAngelsh/SistemasDeManufactura-2024/pull/new/LuisHernanRojasMarin
```

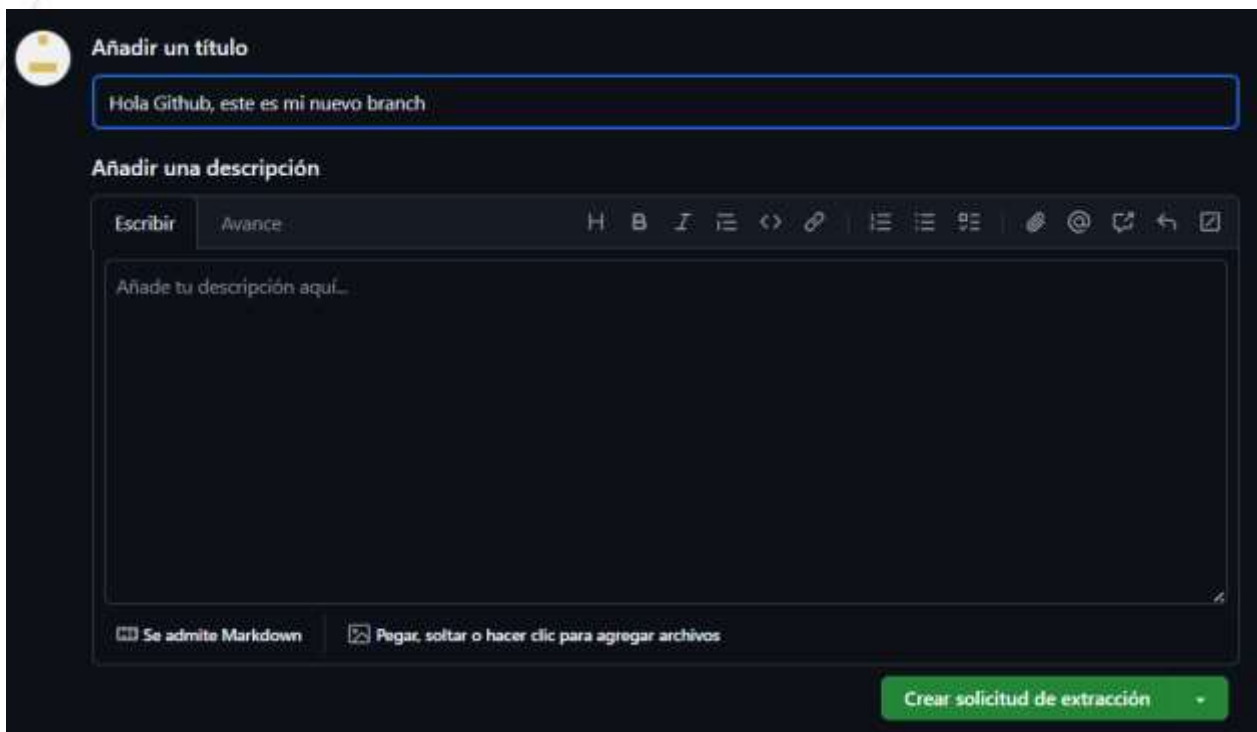


```
Writing objects: 100% (3/3), 332 bytes | 332.00 KiB/s, done.  
Total 3 (delta 2), reused 0 (delta 0), pack-reused 0 (from 0)  
remote: Resolving deltas: 100% (2/2), completed with 2 local objects.  
remote:  
remote: Create a pull request for 'LuisHernanRojasMarin' on GitHub by visiting:  
remote:   https://github.com/DrLAlbertoAngelesH/SistemasDeManufactura-2024/pull/new/LuisHernanRojasMarin  
remote:  
To https://github.com/DrLAlbertoAngelesH/SistemasDeManufactura-2024.git  
* [new branch]      LuisHernanRojasMarin -> LuisHernanRojasMarin  
branch 'LuisHernanRojasMarin' set up to track 'origin/LuisHernanRojasMarin'.
```

10. Verificar que en ningún comando haya notificado algún error, ingresar a Github desde el navegador, confirmar que se haya creado tu nuevo branch y dar click en el botón “Comparar y solicitar extracción/Compare & pull request”



11. Al aparecer el mensaje antes creado, presionar el botón “Crear solicitud de extracción/Create pull request”



12. Al presionar el botón anterior, se confirma que se haya creado tu nuevo branch y esperar que haga merge sobre tus cambios y que el administrador confirme. **(NO PRESIONAR “SOLICITUD DE EXTRACCIÓN DE FUSIÓN”)**



**FIN DEL PROCESO**