

# PROCEDIMIENTO OPERATIVO ESTÁNDAR.

## Propósito:

Por sus siglas en inglés SOP (Standard Operating Procedure), se detalla la manera en que se lleva a cabo un procedimiento o proceso de trabajo específico. En este caso el procedimiento será crear el archivo **.gitignore** sirve para excluir archivos o carpetas que no quieres que Git rastree ni suba al repositorio, como archivos temporales, configuraciones locales o dependencias. Ayuda a mantener el proyecto limpio y seguro.

## Escenario en que se utiliza:

- El archivo **.gitignore** se usa para evitar que Git rastree archivos innecesarios, como configuraciones locales, archivos temporales o dependencias grandes, manteniendo el repositorio limpio y eficiente.

## Definiciones y acrónimos (creación propia):

**Definición de Repositorio:** Un repositorio es un lugar donde se almacenan y gestionan archivos y su historial de versiones.

**Creación de un Repositorio con Git:** Para iniciar un proyecto en Git, primero debes crear un repositorio. Esto se puede hacer en un servidor remoto (como GitHub, GitLab, o Bitbucket) o en tu máquina local.

**Concepto de Rama (Branch):** En Git, una rama es una versión independiente del repositorio principal (llamada "master" o "main" en la mayoría de los casos). Las ramas permiten trabajar en diferentes funcionalidades o correcciones de errores de forma aislada antes de integrar los cambios al proyecto principal.

## Comandos utilizados:

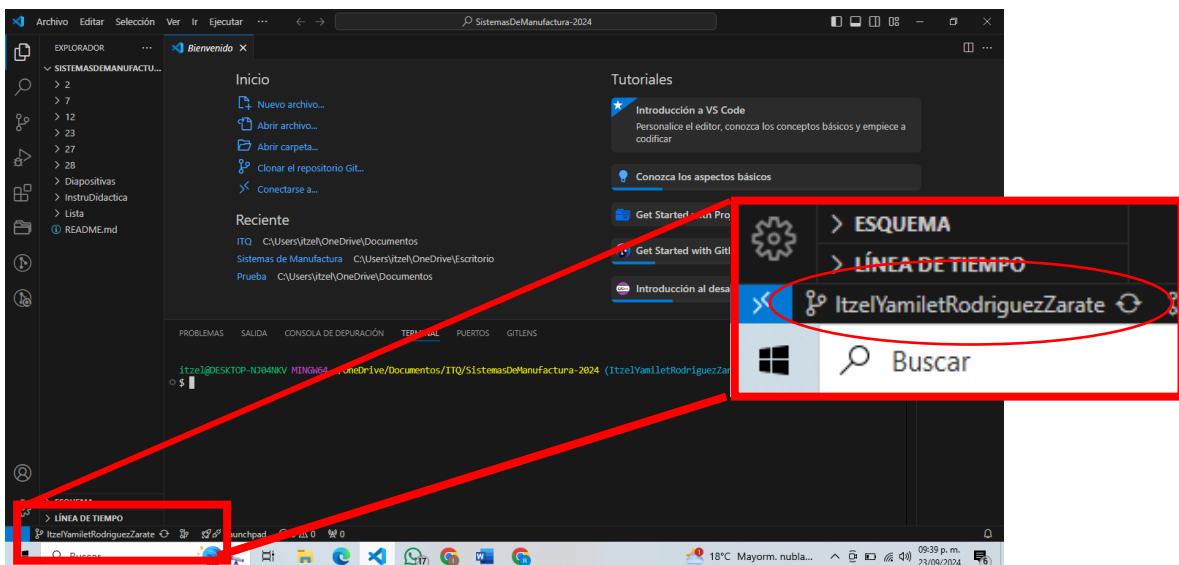
- **.tex** se utiliza para archivos de **LaTeX**, un sistema de preparación de documentos que es ampliamente utilizado en la creación de documentos científicos, técnicos y académicos, etc.
- **.bib** se utiliza para archivos de **BibTeX**, que es un formato para gestionar bibliografías en documentos científicos y académicos, especialmente cuando se usa en combinación con LaTeX.
- **img** es bastante común en proyectos de desarrollo web, diseño, y en otros tipos de proyectos donde se maneja información visual.

Según Asana (2024), una página que explica todo sobre un SOP, dice que los escenarios donde comúnmente se aplica este documento:

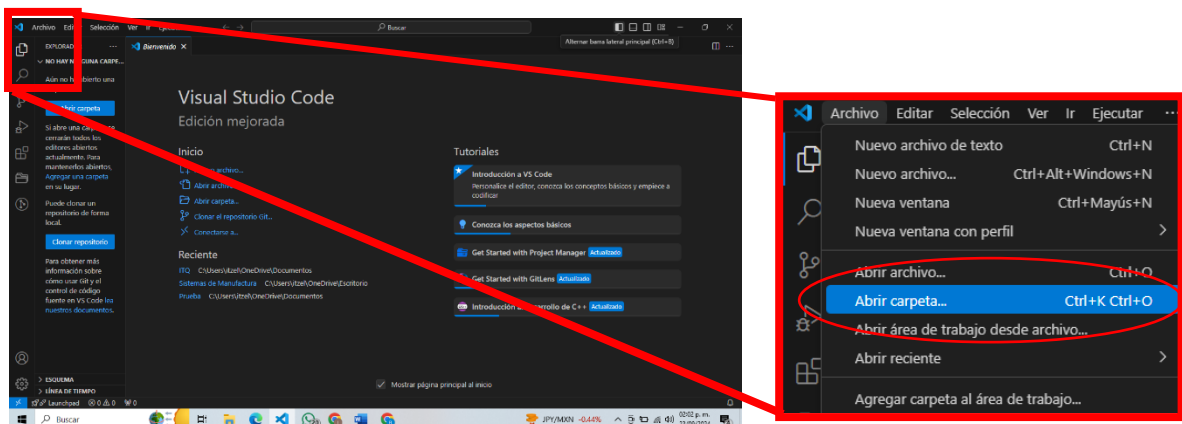
1. Una guía paso a paso para el primer día de trabajo
2. Cómo nombrar archivos según su tipo y ubicación
3. Cómo navegar el software de gestión de proyectos
4. Una descripción general del proceso de revisión del desempeño de tu equipo

### Notas importantes.

Siempre verificar que estés en tu rama, **no** en la main.

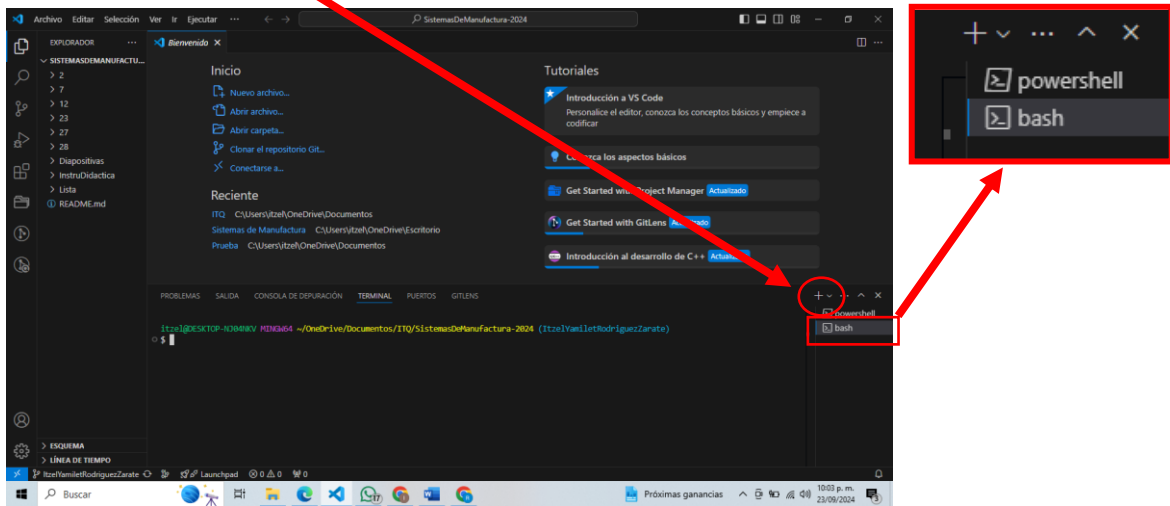


Si no te aparece de esa manera, tienes que abrir tu carpeta. Archivo> abrir carpeta.

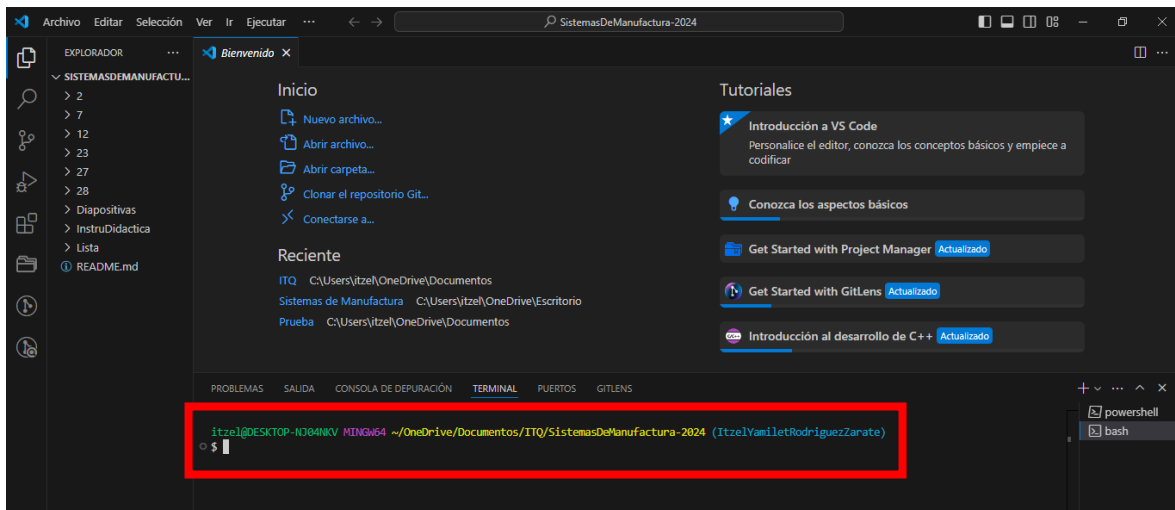


Así abres tu proyecto

1. Verifica en que rama estás. Después control Ñ (Ctrl + Ñ) para abrir la terminal. Abre Git Bash en el **v** para desplegar el git bash



2. Asegúrate de estar en tu Branch para empezar a crear archivo o carpeta.

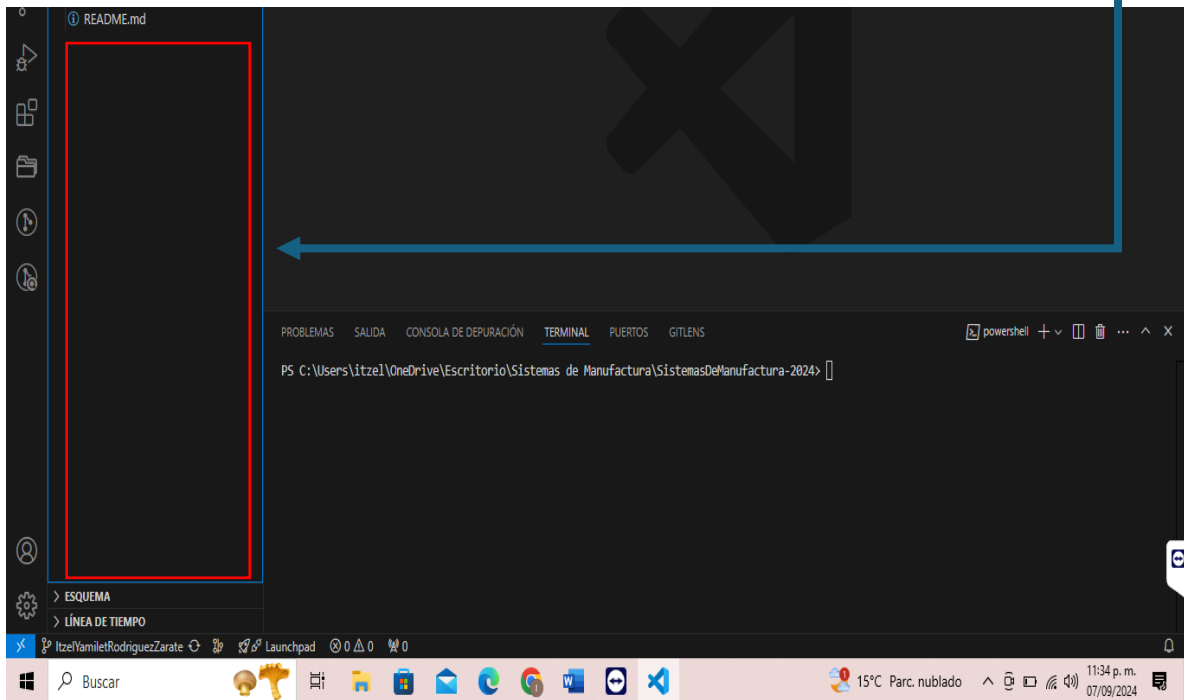


3. Crea el archivo .gitignore

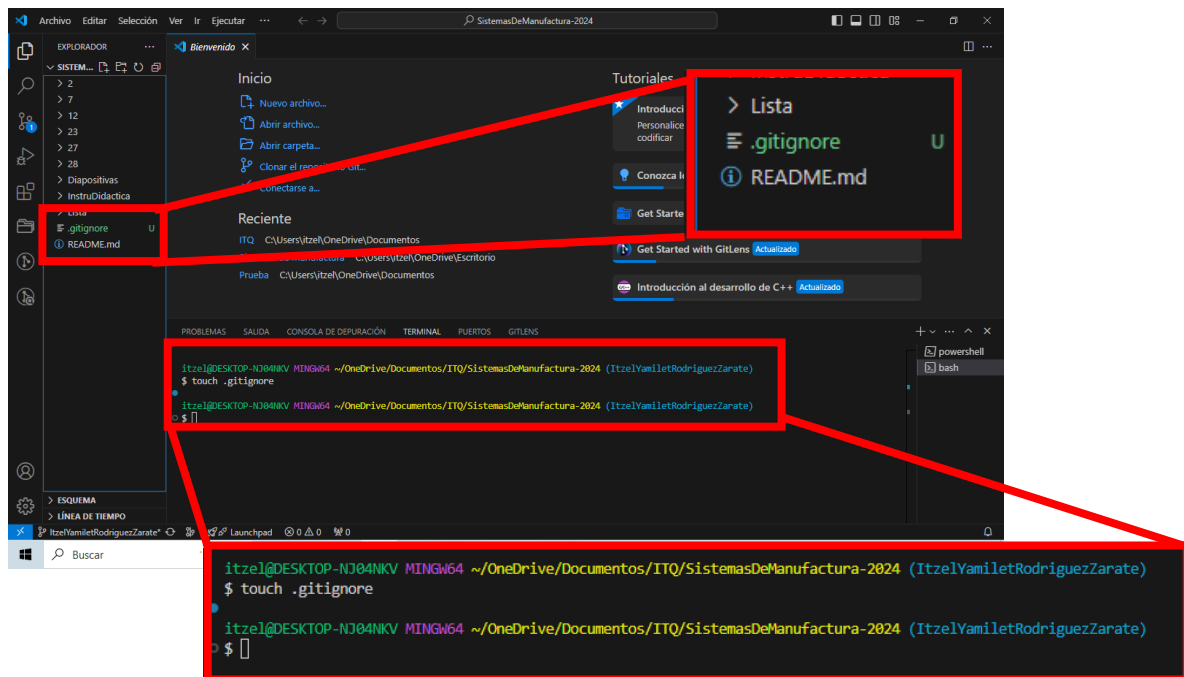
En la línea de comandos escribe: **touch .gitignore**

Para que se cree correctamente, es necesario salirte de tu carpeta creada con el nombre que le pusiste, en este caso el número de lista. Ponerte en explorador, de esta manera.

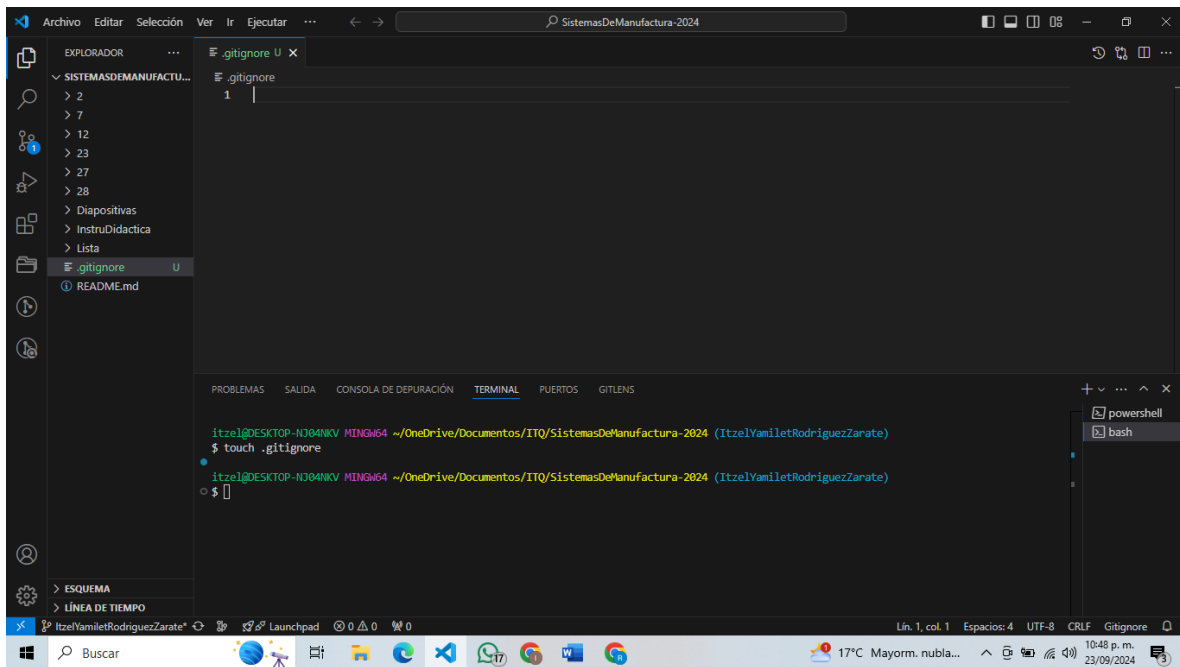
Damos click aquí en el explorador, dentro del cuadro rojo, al hacerlo se pone en azul los bordes.



Escribes el comando, al hacerlo correctamente se verá así tu pantalla.



3. Das doble click en **.gitignore** aparecerá de esta manera.



4. Copiarás esto, el numero 15 lo cambiarás de acuerdo a tu número de lista:

# Ignora todos los nuevos archivos que se van a crear

/\*

# Registra todos los cambios de mi carpeta e ignora la de mis compañeros

!/15

# Si alguien crea un nuevo archivo que no sea yo ignoralo

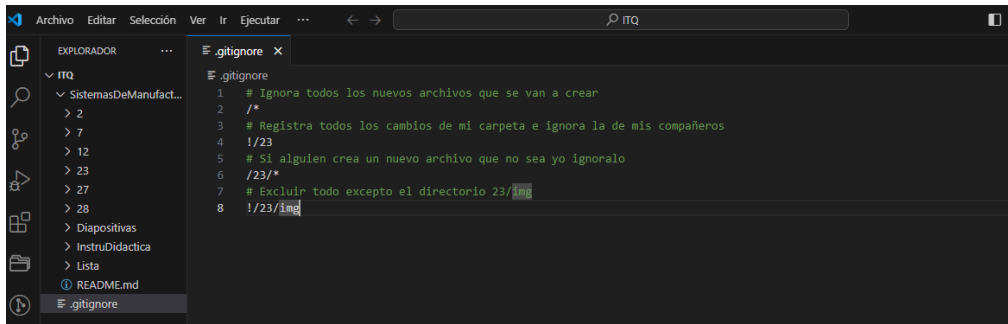
/15/\*

# Excluir todo excepto el directorio 15/img

!/15/img

Pegarás y se verá así:

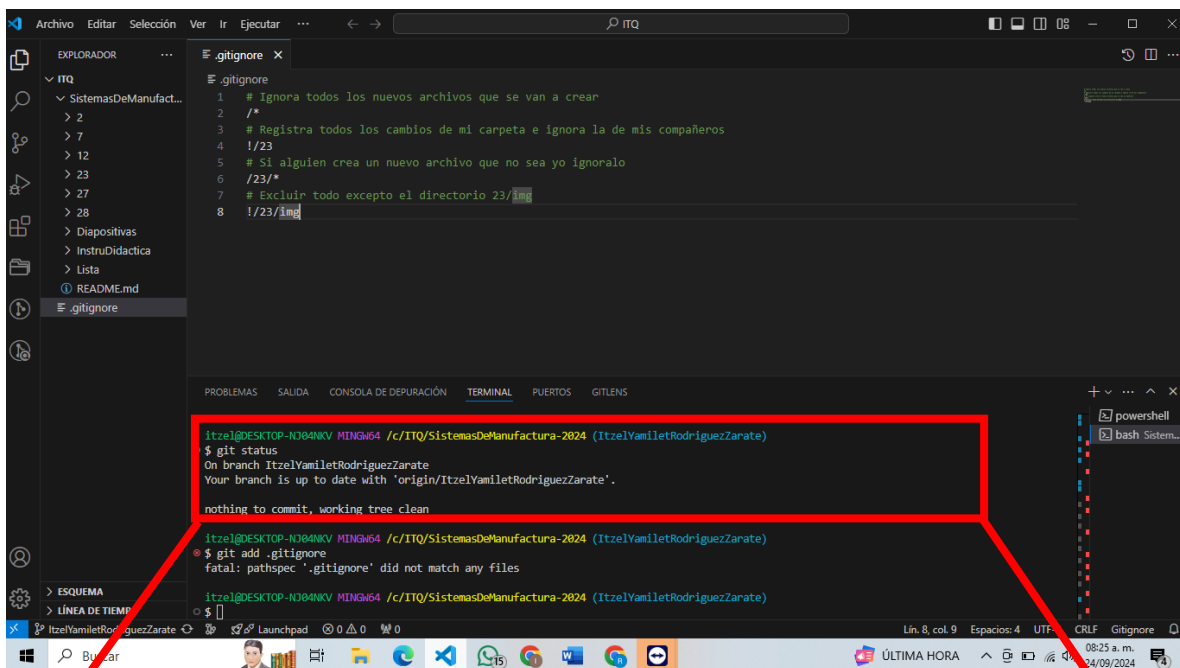
Queda de esta manera.



The screenshot shows the Visual Studio Code editor with the file explorer on the left displaying a project structure. The main editor window shows the content of the `.gitignore` file:

```
1 # Ignora todos los nuevos archivos que se van a crear
2 /*
3 # Registra todos los cambios de mi carpeta e ignora la de mis compañeros
4 !/23
5 # Si alguien crea un nuevo archivo que no sea yo ignoralo
6 /23/*
7 # Excluir todo excepto el directorio 23/img
8 !/23/img
```

5. Pondrás el `> git commit -am "este es mi .gitignore"` y das Enter.



The screenshot shows the Visual Studio Code terminal window with the following output:

```
itzel@DESKTOP-NJ04NKKV MINGW64 /c:/ITQ/SistemasDeManufatura-2024 (ItzelYamiletRodriguezZarate)
$ git status
On branch ItzelYamiletRodriguezZarate
Your branch is up to date with 'origin/ItzelYamiletRodriguezZarate'.

nothing to commit, working tree clean

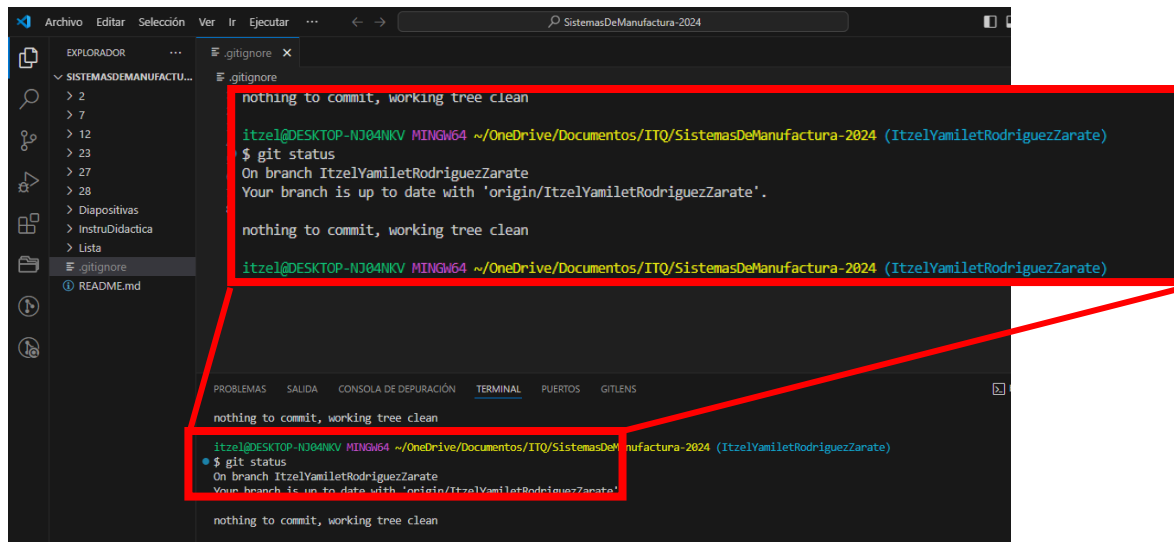
itzel@DESKTOP-NJ04NKKV MINGW64 /c:/ITQ/SistemasDeManufatura-2024 (ItzelYamiletRodriguezZarate)
$ git add .gitignore
fatal: pathspec '.gitignore' did not match any files

itzel@DESKTOP-NJ04NKKV MINGW64 /c:/ITQ/SistemasDeManufatura-2024 (ItzelYamiletRodriguezZarate)
$
```

```
itzel@DESKTOP-NJ04NKKV MINGW64 ~/OneDrive/Documentos/ITQ/SistemasDeManufatura-2024 (ItzelYamiletRodriguezZarate)
$ git commit -am "este es mi .gitignore"
On branch ItzelYamiletRodriguezZarate
Your branch is up to date with 'origin/ItzelYamiletRodriguezZarate'.

nothing to commit, working tree clean
```

6. Escribir > **git status** y Enter.



The screenshot shows a Windows terminal window with the command prompt. The user has entered the command `git status` and the output is displayed. The output indicates that the working tree is clean and the branch is up-to-date with the origin. The terminal window is titled "SistemasDeManufactura-2024".

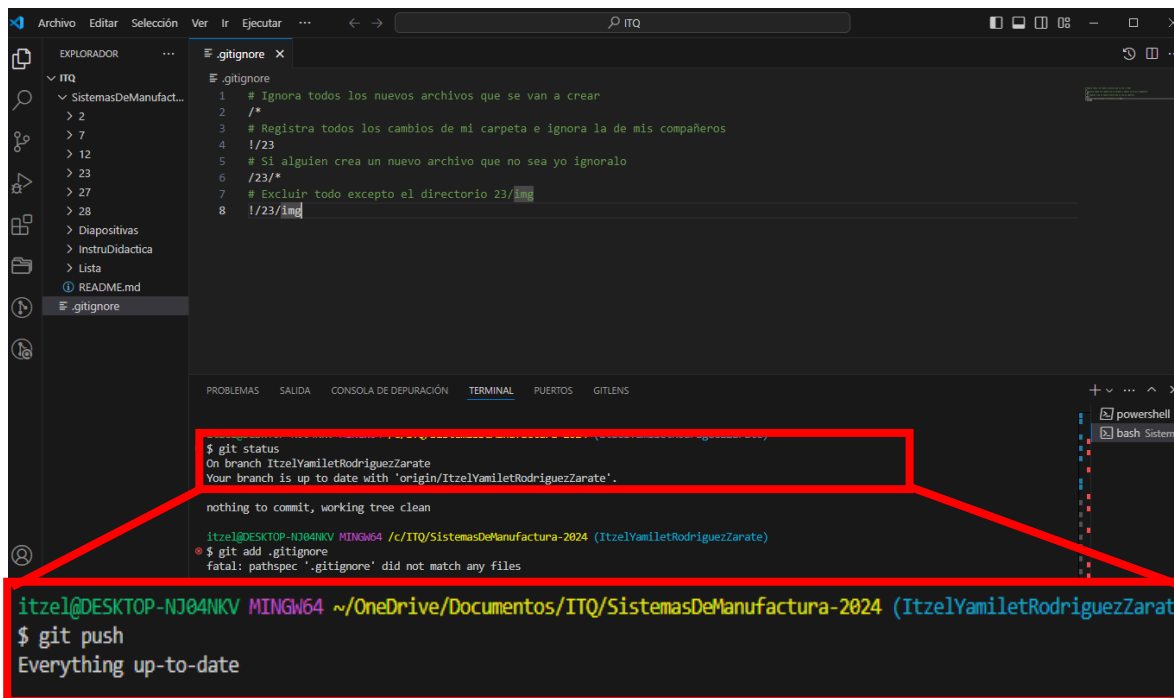
```
nothing to commit, working tree clean

itzel@DESKTOP-NJ04NKKV MINGW64 ~/OneDrive/Documentos/ITQ/SistemasDeManufactura-2024 (ItzelYamiletRodriguezZarate)
$ git status
On branch ItzelYamiletRodriguezZarate
Your branch is up to date with 'origin/ItzelYamiletRodriguezZarate'.

nothing to commit, working tree clean

itzel@DESKTOP-NJ04NKKV MINGW64 ~/OneDrive/Documentos/ITQ/SistemasDeManufactura-2024 (ItzelYamiletRodriguezZarate)
```

7. Después haremos el > **git push** y Enter.



The screenshot shows a Windows terminal window with the command prompt. The user has entered the command `git push` and the output is displayed. The output indicates that everything is up-to-date. The terminal window is titled "ITQ".

```
$ git status
On branch ItzelYamiletRodriguezZarate
Your branch is up to date with 'origin/ItzelYamiletRodriguezZarate'.

nothing to commit, working tree clean

itzel@DESKTOP-NJ04NKKV MINGW64 /c/ITQ/SistemasDeManufactura-2024 (ItzelYamiletRodriguezZarate)
$ git add .gitignore
fatal: pathspec '.gitignore' did not match any files

itzel@DESKTOP-NJ04NKKV MINGW64 ~/OneDrive/Documentos/ITQ/SistemasDeManufactura-2024 (ItzelYamiletRodriguezZarate)
$ git push
Everything up-to-date
```

8. A partir de este paso puedes usar el SOP de cómo crear directorios u archivos para subirlos en tu carpeta.

**Nota:** Si por error modificas otro archivo git no te permitirá subir los archivos a GitHub.