

# Actividad 2.3: supuesto práctico para el uso de Git y Github

## Supuesto Práctico Complejo: Proyecto Colaborativo en GitHub

Imagina que eres parte de un equipo de desarrollo que trabaja en un proyecto de software en GitHub. El proyecto está en constante evolución, y cada miembro del equipo es responsable de diferentes funcionalidades. A lo largo de este ejercicio, te enfrentarás a varias situaciones en las que necesitarás aplicar múltiples comandos y estrategias para gestionar de manera efectiva el proyecto.

### Escenario:

Tu equipo está trabajando en una aplicación web llamada "**Task Manager**", que permite a los usuarios gestionar tareas y proyectos. Se te ha asignado la tarea de desarrollar una **nueva funcionalidad** para la aplicación, la cual consiste en **añadir un sistema de notificaciones** que informe a los usuarios cuando una tarea está a punto de vencer. El proyecto está alojado en GitHub en un **repositorio remoto** al cual todos los desarrolladores contribuyen. Antes de comenzar a trabajar, debes asegurarte de que tu entorno local está sincronizado con el último código y seguir un flujo de trabajo adecuado para evitar conflictos con el trabajo de tus compañeros.

### Tareas que Debes Completar:

#### 1. Clonar el Repositorio del Proyecto

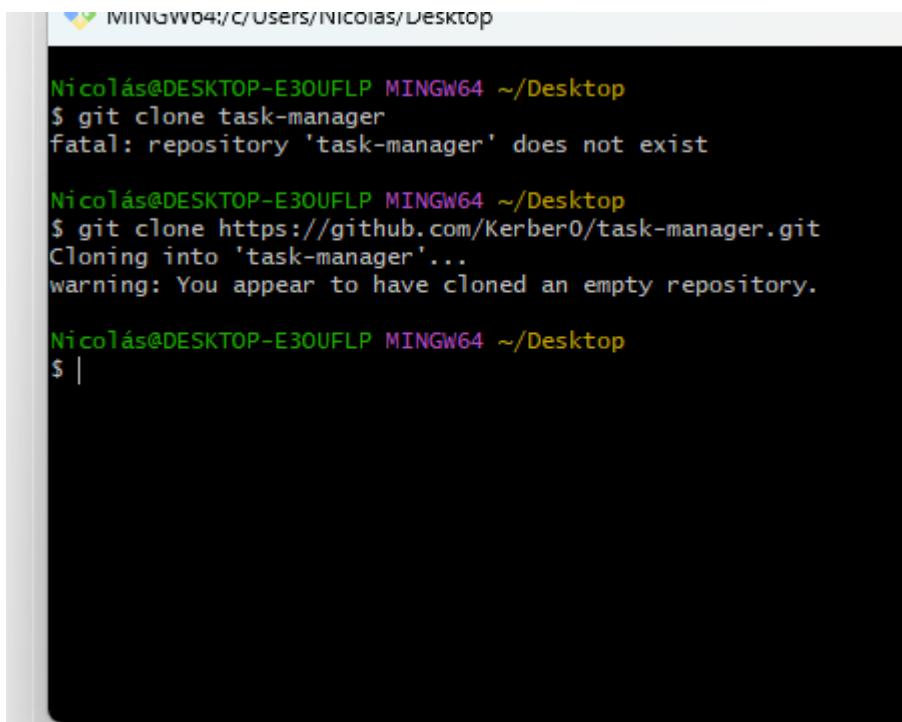
El proyecto está alojado en GitHub, y aún no tienes una copia local del mismo. El primer paso es clonar el repositorio remoto a tu máquina para comenzar a trabajar.

#### • Instrucciones:

- Clona el repositorio remoto llamado `task-manager` a tu equipo.

#### 2. Crear una Nueva Rama para Desarrollar la Funcionalidad

Para no afectar la rama principal (`main`), necesitas crear una nueva rama donde puedas desarrollar la funcionalidad de notificaciones de manera aislada.



```
MINGW64~/C:/Users/NICOLAS/Desktop

Nicolás@DESKTOP-E3OUFLP MINGW64 ~/Desktop
$ git clone task-manager
fatal: repository 'task-manager' does not exist

Nicolás@DESKTOP-E3OUFLP MINGW64 ~/Desktop
$ git clone https://github.com/Kerber0/task-manager.git
Cloning into 'task-manager'...
warning: You appear to have cloned an empty repository.

Nicolás@DESKTOP-E3OUFLP MINGW64 ~/Desktop
$ |
```

```
Nicolás@DESKTOP-E3OUFLP MINGW64 ~/Desktop/task-manager (main)
$ git branch nueva-funcionalidad

Nicolás@DESKTOP-E3OUFLP MINGW64 ~/Desktop/task-manager (main)
$ git branch -a
* main
  nueva-funcionalidad
  remotes/origin/main

Nicolás@DESKTOP-E3OUFLP MINGW64 ~/Desktop/task-manager (main)
$ git branch -b notificaciones
```

●

#### Instrucciones:

- Crea una nueva rama llamada `notificaciones`.
- Cámbiate a esa rama y empieza a trabajar desde allí.

### 3. Realizar Cambios en el Código

Realiza los cambios necesarios en el código para implementar el sistema de notificaciones. A medida que desarrolles la funcionalidad, haz commits pequeños y descriptivos para registrar tu progreso.

```
Nicolás@DESKTOP-E3OUFLP MINGW64 ~/Desktop/task-manager (main)
$ git checkout -b notificaciones
Switched to a new branch 'notificaciones'

Nicolás@DESKTOP-E3OUFLP MINGW64 ~/Desktop/task-manager (notificaciones)
$
```

#### ● Instrucciones:

- Realiza al menos dos cambios importantes en los archivos del proyecto relacionados con las notificaciones. Por ejemplo, haz cambios en el archivo `notificaciones.js` y `archivo.html`
- Añade estos cambios al área de preparación (`staging area`) y realiza un commit con un mensaje descriptivo.

Haz un commit diferente para cada uno de los archivos modificados

```

Nicolás@DESKTOP-E3OUFLP MINGW64 ~/Desktop/task-manager (main)
$ git checkout -b notificaciones
Switched to a new branch 'notificaciones'

Nicolás@DESKTOP-E3OUFLP MINGW64 ~/Desktop/task-manager (notificaciones)
$ git add archivo.html

Nicolás@DESKTOP-E3OUFLP MINGW64 ~/Desktop/task-manager (notificaciones)
$ git status
On branch notificaciones
Changes to be committed:
  (use "git restore --staged <file>..." to unstage)
    new file:   archivo.html

Untracked files:
  (use "git add <file>..." to include in what will be committed)
    notificaciones.js

Nicolás@DESKTOP-E3OUFLP MINGW64 ~/Desktop/task-manager (notificaciones)
$ git add notificaciones.js

Nicolás@DESKTOP-E3OUFLP MINGW64 ~/Desktop/task-manager (notificaciones)
$ git status
On branch notificaciones
Changes to be committed:
  (use "git restore --staged <file>..." to unstage)
    new file:   archivo.html
    new file:   notificaciones.js

Nicolás@DESKTOP-E3OUFLP MINGW64 ~/Desktop/task-manager (notificaciones)
$ git reset notificaciones.js

Nicolás@DESKTOP-E3OUFLP MINGW64 ~/Desktop/task-manager (notificaciones)
$ git status
On branch notificaciones
Changes to be committed:
  (use "git restore --staged <file>..." to unstage)
    new file:   archivo.html

Untracked files:
  (use "git add <file>..." to include in what will be committed)
    notificaciones.js

Nicolás@DESKTOP-E3OUFLP MINGW64 ~/Desktop/task-manager (notificaciones)
$ git commit -m "cambios importantes en archivo.html"
[notificaciones a54b3ed] cambios importantes en archivo.html
1 file changed, 1 insertion(+)
create mode 100644 archivo.html

Nicolás@DESKTOP-E3OUFLP MINGW64 ~/Desktop/task-manager (notificaciones)
$ git add notificaciones.js

Nicolás@DESKTOP-E3OUFLP MINGW64 ~/Desktop/task-manager (notificaciones)
$ git status
On branch notificaciones
Changes to be committed:
  (use "git restore --staged <file>..." to unstage)
    new file:   notificaciones.js

Nicolás@DESKTOP-E3OUFLP MINGW64 ~/Desktop/task-manager (notificaciones)
$ git commit -m "cambios en notificaciones"
[notificaciones 0d6661f] cambios en notificaciones
1 file changed, 1 insertion(+)
create mode 100644 notificaciones.js

Nicolás@DESKTOP-E3OUFLP MINGW64 ~/Desktop/task-manager (notificaciones)
$

```

## 4. Sincronizar los Cambios con el Repositorio Remoto

Mientras estabas trabajando en tu funcionalidad, el equipo realizó actualizaciones importantes en la rama principal. Antes de subir tus cambios, es crucial que integres las últimas actualizaciones para evitar conflictos.

- **Instrucciones:**

- Cambia a la rama `main` y descarga las actualizaciones más recientes del repositorio remoto.
- Integra estos cambios en tu rama `notificaciones`.

```
Nicolás@DESKTOP-E3OUFLP MINGW64 ~/Desktop/task-manager (notificaciones)
$

Nicolás@DESKTOP-E3OUFLP MINGW64 ~/Desktop/task-manager (notificaciones)
$ git checkout main
Switched to branch 'main'
Your branch is up to date with 'origin/main'.

Nicolás@DESKTOP-E3OUFLP MINGW64 ~/Desktop/task-manager (main)
$ git pull origin main
From https://github.com/Kerber0/task-manager
* branch      main      -> FETCH_HEAD
Already up to date.

Nicolás@DESKTOP-E3OUFLP MINGW64 ~/Desktop/task-manager (main)
$ git checkout notificaciones
Switched to branch 'notificaciones'

Nicolás@DESKTOP-E3OUFLP MINGW64 ~/Desktop/task-manager (notificaciones)
$ git merge main
Already up to date.

Nicolás@DESKTOP-E3OUFLP MINGW64 ~/Desktop/task-manager (notificaciones)
$
```

## 5. Resolver Conflictos de Fusión

Al fusionar los cambios de la rama `main` en tu rama `notificaciones`, Git detecta un conflicto en el archivo `notificaciones.js`, ya que otro desarrollador también trabajó en una funcionalidad relacionada. Tu tarea es resolver este conflicto.

- **Instrucciones:**

- Identifica el conflicto y edita manualmente el archivo `notificaciones.js` para resolver el problema.
- Añade el archivo al staging area y realiza un commit con la resolución del conflicto.

## 6. Guardar Cambios Temporales

Estás a punto de finalizar tu trabajo en la funcionalidad de notificaciones cuando recibes una solicitud urgente para corregir un error en la rama `main`. No quieres perder tus cambios actuales, pero tampoco estás listo para hacer un commit.

- **Instrucciones:**

- Guarda temporalmente tus cambios en la rama `notificaciones` sin hacer

un commit.

- Cambia a la rama `main` para corregir el error.

```
Nicolás@DESKTOP-E30UFLP MINGW64 ~/Desktop/task-manager (notificaciones)
$ git status
On branch notificaciones
Changes to be committed:
  (use "git restore --staged <file>..." to unstage)
        modified:   archivo.html
        modified:   notificaciones.js

Changes not staged for commit:
  (use "git add <file>..." to update what will be committed)
  (use "git restore <file>..." to discard changes in working directory)
        modified:   archivo.html

Nicolás@DESKTOP-E30UFLP MINGW64 ~/Desktop/task-manager (notificaciones)
$ git stash
Saved working directory and index state WIP on notificaciones: 84b90a9 revisar esto

Nicolás@DESKTOP-E30UFLP MINGW64 ~/Desktop/task-manager (notificaciones)
$ git status
On branch notificaciones
nothing to commit, working tree clean

Nicolás@DESKTOP-E30UFLP MINGW64 ~/Desktop/task-manager (notificaciones)
$ git checkout main
Switched to branch 'main'
Your branch is ahead of 'origin/main' by 3 commits.
  (use "git push" to publish your local commits)

Nicolás@DESKTOP-E30UFLP MINGW64 ~/Desktop/task-manager (main)
$ |
```

## 7. Corregir un Bug en la Rama Principal

Una vez en la rama `main`, realiza los cambios necesarios para corregir el error que fue reportado. Asegúrate de realizar un commit con un mensaje claro y descriptivo. El archivo sobre el que corregir el bug es `botones.html`

- **Instrucciones:**

- Realiza la corrección del bug y haz un commit con un mensaje que explique qué has corregido.

```

Nicolás@DESKTOP-E3OUFLP MINGW64 ~/Desktop/task-manager (main)
$ git status
On branch main
Your branch is ahead of 'origin/main' by 3 commits.
  (use "git push" to publish your local commits)

Untracked files:
  (use "git add <file>..." to include in what will be committed)
      botones.html

nothing added to commit but untracked files present (use "git add" to track)

Nicolás@DESKTOP-E3OUFLP MINGW64 ~/Desktop/task-manager (main)
$ git add .

Nicolás@DESKTOP-E3OUFLP MINGW64 ~/Desktop/task-manager (main)
$ git commit -m "se corrigieron los bugs generados por bonotes.html"
[main 7257ba4] se corrigieron los bugs generados por bonotes.html
 1 file changed, 3 insertions(+)
 create mode 100644 botones.html

Nicolás@DESKTOP-E3OUFLP MINGW64 ~/Desktop/task-manager (main)
$ |

```

## 8. Volver a tu Trabajo en la Funcionalidad de Notificaciones

Después de corregir el bug en `main`, necesitas retomar tu trabajo en la funcionalidad de notificaciones. Para hacerlo, debes volver a la rama `notificaciones` y restaurar los cambios que guardaste temporalmente.

### • Instrucciones:

- Cambia a la rama `notificaciones` y recupera los cambios que guardaste previamente.

## 9. Finalizar la Funcionalidad y Subirla al Repositorio Remoto

Una vez que hayas completado el desarrollo de la funcionalidad de notificaciones, debes subir tu rama al repositorio remoto en GitHub para que el equipo pueda revisarla.

### • Instrucciones:

- Sube la rama `notificaciones` al repositorio remoto.

```

Nicolás@DESKTOP-E3OUFLP MINGW64 ~/Desktop/task-manager (main)
$ git checkout notificaciones
Switched to branch 'notificaciones'

Nicolás@DESKTOP-E3OUFLP MINGW64 ~/Desktop/task-manager (notificaciones)
$ git stash pop
On branch notificaciones
Changes not staged for commit:
  (use "git add <file>..." to update what will be committed)
  (use "git restore <file>..." to discard changes in working directory)
        modified:   archivo.html
        modified:   notificaciones.js

no changes added to commit (use "git add" and/or "git commit -a")
Dropped refs/stash@{0} (e056e6b241b4eaa0a60ac802fe007ee5d6695fda)

Nicolás@DESKTOP-E3OUFLP MINGW64 ~/Desktop/task-manager (notificaciones)
$ git add .

Nicolás@DESKTOP-E3OUFLP MINGW64 ~/Desktop/task-manager (notificaciones)
$ git commit -m "nueva funcionalidad para revisar, se sube a github"
[notificaciones 14a3ee2] nueva funcionalidad para revisar, se sube a github
 2 files changed, 10 insertions(+), 2 deletions(-)

Nicolás@DESKTOP-E3OUFLP MINGW64 ~/Desktop/task-manager (notificaciones)
$ git status
On branch notificaciones
nothing to commit, working tree clean

Nicolás@DESKTOP-E3OUFLP MINGW64 ~/Desktop/task-manager (notificaciones)
$

```

## 10. Revisar y Fusionar el Pull Request

Tu equipo ha revisado tus cambios y ha aceptado integrarlas en `main`. Procede a fusionar la rama `notificaciones` con `main`.

### • Instrucciones:

- Fusiona la rama `notificaciones` con `main` y elimina la rama una vez que los cambios se hayan incorporado.

```
Nicolás@DESKTOP-E3OUFLP MINGW64 ~/Desktop/task-manager (notificaciones)
$ git checkout main
Switched to branch 'main'
Your branch is ahead of 'origin/main' by 4 commits.
  (use "git push" to publish your local commits)

Nicolás@DESKTOP-E3OUFLP MINGW64 ~/Desktop/task-manager (main)
$ git status
On branch main
Your branch is ahead of 'origin/main' by 4 commits.
  (use "git push" to publish your local commits)

nothing to commit, working tree clean

Nicolás@DESKTOP-E3OUFLP MINGW64 ~/Desktop/task-manager (main)
$ git merge notificaciones
Updating 7257ba4..d9a4d67
Fast-forward
 archivo.html      | 10 ++++++++-
 notificaciones.js |  2 +-
 2 files changed, 10 insertions(+), 2 deletions(-)

Nicolás@DESKTOP-E3OUFLP MINGW64 ~/Desktop/task-manager (main)
$ git status
On branch main
Your branch is ahead of 'origin/main' by 6 commits.
  (use "git push" to publish your local commits)

nothing to commit, working tree clean

Nicolás@DESKTOP-E3OUFLP MINGW64 ~/Desktop/task-manager (main)
$ git branch -d notificaciones
Deleted branch notificaciones (was d9a4d67).

Nicolás@DESKTOP-E3OUFLP MINGW64 ~/Desktop/task-manager (main)
$ git branch -a
* main
  nueva-funcionalidad
  remotes/origin/main
  remotes/origin/notificaciones

Nicolás@DESKTOP-E3OUFLP MINGW64 ~/Desktop/task-manager (main)
$ |
```

<https://github.com/Kerber0/task-manager>