

# HERRAMIENTA DE DESARROLLO

## Laboratorio: Estrategias de Ramificación y Fusión en Git

### Objetivo:

El objetivo de este laboratorio es enseñar a los estudiantes cómo usar las estrategias de ramificación (branching) y fusión (merging) en Git para gestionar de manera eficiente el flujo de trabajo en proyectos de software. Los estudiantes aprenderán a crear ramas para desarrollar nuevas características, fusionarlas correctamente y gestionar posibles conflictos.

---

### Procedimientos Básicos:

#### 1. Crear un Repositorio Local:

Para crear un repositorio nuevo:

```
git init
```

#### 2. Clonar un Repositorio Remoto:

Para clonar un repositorio desde GitHub (o cualquier servicio de Git):

```
git clone <URL_del_repositorio>
```

#### 3. Crear una Nueva Rama:

Para crear una nueva rama para una nueva característica o corrección:

```
git branch <nombre_rama>
```

#### 4. Cambiar de Rama:

Para cambiar a la rama recién creada:

```
git checkout <nombre_rama>
```

También puedes crear y cambiar de rama en un solo paso:

```
git checkout -b <nombre_rama>
```

#### 5. Ver las Ramas Existentes:

Para listar todas las ramas locales en tu repositorio:

```
git branch
```

## 6. Fusionar una Rama:

Para fusionar una rama en la rama actual (por ejemplo, fusionar `feature/login` en `main`):

```
git checkout main
git merge feature/login
```

## 7. Resolución de Conflictos:

Cuando ocurre un conflicto durante la fusión, Git marcará los archivos conflictivos. Deberás editar los archivos en conflicto, resolver las diferencias y hacer un commit de la fusión:

```
git add <archivo_resuelto>
git commit -m "Resuelto conflicto de fusión"
```

## 8. Eliminar una Rama:

Una vez que una rama ha sido fusionada, puedes eliminarla:

```
git branch -d <nombre_rama>
```

## 9. Subir Cambios a un Repositorio Remoto:

Para subir la rama y sus cambios al repositorio remoto:

```
git push origin <nombre_rama>
```

## 10. Obtener Cambios del Repositorio Remoto:

Para actualizar tu repositorio local con los cambios del repositorio remoto:

```
git pull origin <nombre_rama>
```

---

## Preguntas

1. **¿Qué es una estrategia de ramificación en Git?**
  - o **Respuesta:** Es una técnica para crear y gestionar múltiples ramas en un repositorio para desarrollar nuevas características, corregir errores o experimentar sin afectar el código principal. Las ramas permiten un desarrollo paralelo y controlado.
2. **¿Cuál es la diferencia entre `git merge` y `git rebase`?**
  - o **Respuesta:** `git merge` une dos ramas manteniendo el historial de ambas ramas. `git rebase` toma los cambios de una rama y los aplica sobre otra, reescribiendo el historial, lo que resulta en un historial más lineal.
3. **¿Qué es un conflicto de fusión (merge conflict) en Git?**

- **Respuesta:** Un conflicto de fusión ocurre cuando Git no puede combinar automáticamente los cambios de dos ramas debido a que los cambios afectan las mismas líneas de un archivo de manera incompatible. El conflicto debe ser resuelto manualmente.
- 4. **¿Por qué es importante crear ramas para nuevas funcionalidades o correcciones?**
  - **Respuesta:**
- 5. **¿Qué comando se utiliza para fusionar una rama con la rama actual?**
  - **Respuesta:**
- 6. **¿Qué significa hacer un `git pull` y qué hace en el contexto de ramas?**
  - **Respuesta:**
- 7. **Crea un repositorio local en tu máquina**
  - **Respuesta:**
- 8. **Crea una nueva rama llamada `feature/signup`**
  - **Respuesta:**
- 9. **Cambia a la rama `feature/signup`**
  - **Respuesta:**
- 10. **Resuelve un conflicto de fusión (simulado) y haz un commit para completar la fusión.**
  - **Respuesta:**