

# Herramientas Gráficas de Git para Ubuntu - Guía Completa

## Introducción

Esta guía recopila las mejores herramientas gráficas para aprender, practicar y visualizar Git en Ubuntu. Incluye aplicaciones nativas, plugins para editores, páginas web interactivas y herramientas de línea de comandos con interfaz visual.

---

## Aplicaciones Gráficas Nativas para Ubuntu

### GitKraken (Recomendado para principiantes)

- **Descripción:** Cliente Git visual más popular y fácil de usar
- **Instalación:**

```
bash

# Descargar desde https://www.gitkraken.com/
# O instalar via Snap:
sudo snap install gitkraken --classic
```

- **Características:**
  - Visualización de árbol de commits muy clara
  - Drag & drop para merge y rebase
  - Interfaz intuitiva para resolver conflictos
  - Integración con GitHub, GitLab, Bitbucket
- **Ideal para:** Principiantes y usuarios que buscan simplicidad

### GitG

- **Descripción:** Cliente Git ligero para GNOME
- **Instalación:**

```
bash

sudo apt update
sudo apt install gitg
```

- **Características:**
  - Visualización de historial de commits
  - Vista de diferencias integrada
  - Interfaz minimalista
  - Rápido y consume pocos recursos

### QGit

- **Descripción:** Visor de repositorios Git basado en Qt

- **Instalación:**

```
bash  
  
sudo apt install qgit
```

- **Características:**

- Navegación rápida por el historial
- Vista de parches y diferencias
- Búsqueda avanzada en commits

## GitAhead

- **Descripción:** Cliente Git moderno y rápido

- **Instalación:** Descargar desde <https://gitahead.github.io/gitahead.com/>

- **Características:**

- Interfaz moderna y responsive
- Vista de diferencias lado a lado
- Staging interactivo
- Merge tool integrado

## Ungit

- **Descripción:** Interfaz web para Git que se ejecuta localmente

- **Instalación:**

```
bash  
  
# Instalar Node.js primero  
sudo apt install nodejs npm  
  
# Instalar Ungit  
npm install -g ungit  
  
# Ejecutar  
ungit
```

- **Características:**

- Interfaz web intuitiva
- Visualización de grafo en tiempo real
- Operaciones con drag & drop

---

## Plugins y Extensiones para Editores

### Visual Studio Code

## GitLens (Imprescindible)

- **Instalación:** Buscar "GitLens" en el marketplace de VSCode
- **Características:**
  - Blame annotations en línea
  - Explorador de repositorio
  - Vista de historial de archivos
  - Comparación de commits

## Git Graph

- **Instalación:** Buscar "Git Graph" en VSCode
- **Características:**
  - Visualización gráfica del historial
  - Operaciones Git desde la interfaz
  - Vista de branches y tags

## Git History

- **Instalación:** Buscar "Git History" en VSCode
- **Características:**
  - Historial de commits de archivos
  - Comparación entre commits
  - Búsqueda en historial

## Vim/Neovim

### Fugitive

- **Instalación:** Usar un plugin manager como vim-plug

```
vim
```

```
Plug 'tpope/vim-fugitive'
```

### GitGutter

- **Instalación:**

```
vim
```

```
Plug 'airblade/vim-gitgutter'
```

## Sublime Text

### GitSavvy

- **Instalación:** Package Control → Install Package → GitSavvy
- 

## Herramientas Web Interactivas (Para Aprender)

### Learn Git Branching

- **URL:** [https://learngitbranching.js.org/?locale=es\\_ES](https://learngitbranching.js.org/?locale=es_ES)
- **Descripción:** Tutorial interactivo visual de Git
- **Características:**
  - Lecciones paso a paso
  - Visualización en tiempo real
  - Desafíos prácticos
  - Disponible en español

### Git Visualizer

- **URL:** <https://git-school.github.io/visualizing-git/>
- **Descripción:** Simulador visual de comandos Git
- **Características:**
  - Ejecuta comandos y ve los resultados
  - Ideal para experimentar sin riesgo

### Visualizing Git

- **URL:** <http://git-school.github.io/visualizing-git/>
- **Descripción:** Herramienta para visualizar conceptos de Git

### Oh My Git!

- **URL:** <https://ohmygit.org/>
- **Descripción:** Juego para aprender Git
- **Instalación en Ubuntu:**

```
bash
```

```
# Descargar desde la página oficial
```

```
# Es un juego que enseña Git de manera interactiva
```

---

## Herramientas de Terminal con Interfaz Visual

### Tig

- **Descripción:** Interfaz de texto para Git

- **Instalación:**

```
bash

sudo apt install tig
```

- **Uso básico:**

```
bash

# Ver historial
tig

# Ver diferencias
tig status
```

## LazyGit

- **Descripción:** Terminal UI simple para comandos Git

- **Instalación:**

```
bash

# Método 1: Desde releases de GitHub
LAZYGIT_VERSION=$(curl -s "https://api.github.com/repos/jesseduffield/lazygit/releases/latest" | grep -Po '"tag_name": "v[0-9]+.[0-9]+.[0-9]+' | sed -E 's/^[^"]*"([^"]*)"$/\1/')
curl -Lo lazygit.tar.gz "https://github.com/jesseduffield/lazygit/releases/latest/download/lazygit_${LAZYGIT_VERSION}.tar.gz"
tar xzf lazygit.tar.gz lazygit
sudo install lazygit /usr/local/bin

# Método 2: Usando go
go install github.com/jesseduffield/lazygit@latest
```

- **Uso:** Ejecutar `lazygit` en cualquier repositorio Git

## GitUI

- **Descripción:** Terminal UI rápida para Git escrita en Rust

- **Instalación:**

```
bash

# Instalar Rust primero
curl --proto '=https' --tlsv1.2 -sSf https://sh.rustup.rs | sh

# Instalar GitUI
cargo install gitui
```

## Atlassian Git Tutorial

- **URL:** <https://www.atlassian.com/git/tutorials>
- **Descripción:** Tutoriales completos con diagramas visuales

## Pro Git Book

- **URL:** <https://git-scm.com/book/es/v2>
- **Descripción:** Libro oficial de Git con ejemplos visuales

## Git Flow Cheatsheet

- **URL:** [https://danielkummer.github.io/git-flow-cheatsheet/index.es\\_ES.html](https://danielkummer.github.io/git-flow-cheatsheet/index.es_ES.html)
- **Descripción:** Guía visual de Git Flow

## Videos y Cursos

- **Git and GitHub for Beginners** (YouTube)
- **Coursera Git Courses** con visualizaciones
- **Udemy Git Courses** con práctica visual

---

## Configuración Recomendada para Ubuntu

### Configurar Git con colores

```
bash

git config --global color.ui auto
git config --global color.branch auto
git config --global color.diff auto
git config --global color.status auto
```

### Alias útiles para visualización

```
bash

git config --global alias.lg "log --oneline --decorate --graph --all"
git config --global alias.lga "log --graph --pretty=format:'%Cred%h%Creset -%C(yellow)%d%Creset %s %Cgreen(%cr'"
git config --global alias.st "status -sb"
```

### Herramientas de diferencias visuales

```
bash
```

```
# Configurar meld como herramienta de diff
```

```
sudo apt install meld
```

```
git config --global diff.tool meld
```

```
git config --global merge.tool meld
```

```
# Configurar kdiff3 (alternativa)
```

```
sudo apt install kdiff3
```

```
git config --global diff.tool kdiff3
```

```
git config --global merge.tool kdiff3
```



## Plan de Aprendizaje Recomendado

### Nivel Principiante (Semanas 1-2)

1. **Learn Git Branching** - Completar todos los tutoriales básicos
2. **GitKraken** - Practicar operaciones básicas (add, commit, push, pull)
3. **VSCode con GitLens** - Familiarizarse con blame y historial

### Nivel Intermedio (Semanas 3-4)

1. **LazyGit** - Aprender atajos de teclado
2. **Git Flow** - Practicar con GitKraken
3. **Resolución de conflictos** - Usar Meld o herramienta visual

### Nivel Avanzado (Semanas 5-6)

1. **Tig** - Navegación rápida por terminal
2. **Rebase interactivo** - Con herramientas visuales
3. **Git hooks** - Automatización de procesos



## Comandos Útiles para Visualización

### Ver historial gráfico en terminal

```
bash

# Historial gráfico simple
git log --graph --oneline --all

# Historial detallado con colores
git log --graph --pretty=format:'%Cred%h%Creset -%C(yellow)%d%Creset %s %Cgreen(%cr) %C(bold blue)<%an>%'

# Ver diferencias visuales
git diff --color-words

# Estado con colores
git status --short --branch
```

## Alias para comandos gráficos

```
bash

# Agregar a ~/.bashrc o ~/.zshrc
alias glog='git log --graph --oneline --all'
alias gst='git status --short --branch'
alias gdiff='git diff --color-words'
```

## Solución de Problemas Comunes

### GitKraken no abre

```
bash

# Verificar instalación
snap list | grep gitkraken

# Reinstalar si es necesario
sudo snap remove gitkraken
sudo snap install gitkraken --classic
```

### Problemas con herramientas de merge

```
bash

# Verificar configuración
git config --global --list | grep -E "(diff|merge)\.tool"

# Reconfigurar si es necesario
git config --global diff.tool meld
```

## Conclusión



Esta guía proporciona un arsenal completo de herramientas para dominar Git de manera visual en Ubuntu. Comienza con herramientas simples como GitKraken y Learn Git Branching, luego progresa hacia herramientas más avanzadas como LazyGit y Tig.

**Recomendación:** Combina el aprendizaje teórico con práctica constante usando diferentes herramientas para encontrar el flujo de trabajo que mejor se adapte a ti.

---

*Última actualización: Septiembre 2025*