

# Convenciones en Git

Sebastián Alejandro Vidal Rojas



## 1 Inicialización del directorio raíz y las ramas principales en Git

Para empezar a desarrollar tu proyecto con el sistema de control de versionado Git, deberás crear un repositorio Git en un proveedor de servicios Git (GitHub, GitLab, Gitee...). La estructura de los repositorios sigue una analogía de ramas que dividen el programa en distintas etapas o versiones de desarrollo. Normalmente un proyecto nuclear se compone de las siguientes ramas: *master* o *main*, *dev* o *unstable*, *feature release* o *stable*, *legacy* o *old-stable*...

- Master compone la rama principal de desarrollo y debe ser usada de manera cautelosa para poder mantener la estabilidad del proyecto.
- Dev es la rama de desarrollo que experimenta y cambia constantemente para poder generar soluciones relevantes para la master branch.
- Una feature release es una versión hito del *software* que contenga estabilidad y características deseadas. Estas versiones normalmente mantienen soporte en conjunto con la feature release siguiente. (Puede que la versión 1.64.3 salga a pesar de que la versión 2.0.3 ya ha salido, ya que la versión 1.xx.x contiene características deseadas o de legado y por ende los usuarios dependen de ésta.)
- Legacy es una rama antigua que sirve como medios de archivación para *software* que ya no se mantiene activamente por el repositorio original o es de una versión deprecada y que ya no es relevante bajo el juicio del desarrollador.

## 2 Vocabulario de trabajo en Git

Los mensajes de cambios en los cometidos en Git deben contener títulos y descripciones cortas y concisas. Normalmente estos siguen una estructura de {qué se hizo};{dónde se hizo}. Una vez hecho los cambios necesarios para una feature release se lanza una versión con un changelog donde se detallan los cambios desde la última versión feature.

## 3 Lidar con los problemas comunes en Git

Cuando se colabora, la comunicación es clave para evitar problemas a la hora de unir los trabajos contribuidos por los miembros del grupo.

## Convenciones en Git

En el caso que estos problemas aparezcan, aquí se detallan algunos consejos:

- Al tratar de cometer cambios sin haber actualizado la rama maestra o la rama principal de desarrollo una vez hecho cambios desde tu pull: podrás hacer un git rebase para poder integrar los cambios hechos por algún miembro del equipo en tus propios cambios y así cometerlos en la rama.