

# Git and Git Flow

Camilo Andrés Rodríguez Garzón

Universidad EAFIT

16 de abril de 2020

# Contenido

What is Git?

Some commands

Git flow

Instalación

Work flow

Inicialización

Feature

Release

Hotfix

Bibliografía

# What is Git?

- ▶ Started **2005**
- ▶ Created for (**Linus Torvalds**)
- ▶ Git is a free and open source distributed **version control system**

# Some commands

## when you begin a new project

- ▶ `git clone url` – downloads branch global in git local
- ▶ `git fetch` – if you need to change to other branch
- ▶ `git checkout`

## when you work in a common day

- ▶ `git status` – show you local changes
- ▶ `git pull origin name-branch`
- ▶ `git add name-file` – add all current changes for the next commit
- ▶ `git commit -m "information of state"` – commit previously staged changes
- ▶ `git push origin name-branch` – put changes in global repository

# Other commands

when you require to solve advanced issues

- ▶ `git checkout -b name-new-branch` – create new branch with the same features of the current branch
- ▶ `git merge`
- ▶ `git stash`
- ▶ `git stash pop`
- ▶ `git reset`
- ▶ `git revert`
- ▶ `git cherry-pick`

## Git flow

El modelo de ramificación **git flow** de **Vincent Driessen** es un flujo de trabajo de gestión de lanzamiento y ramificación git que ayuda a los desarrolladores a realizar un seguimiento de las características, revisiones y lanzamientos en proyectos de software.

# Instalación

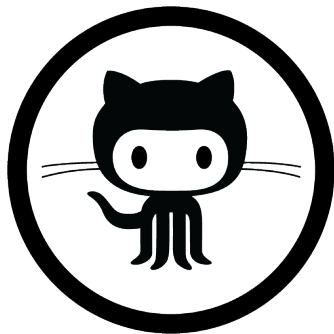
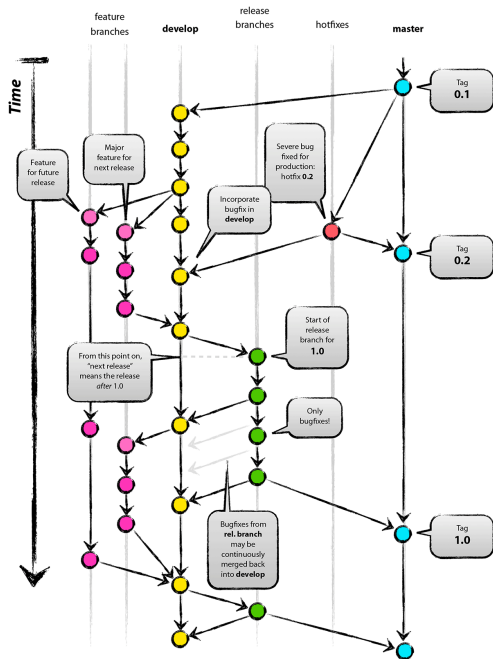


Figura 1: *Instalando git-flow*

# Work flow





# Inicialización

Para inicializar un nuevo repositorio con la estructura de rama básica, use:

- ▶ `git flow init -d`

# Crear feature

Para list/start/finish feature branches, usar:

- ▶ git flow feature
- ▶ git flow feature start <name> [<base>]
- ▶ git flow feature finish <name>

Ejemplo desde el branch feature/CA-006

```
git flow feature start CA-007 develop
```

# Crear feature

Para push/pull una feature branch a el repositorio remoto, usar:

- ▶ `git flow feature publish <name>`
- ▶ `git flow feature track <name>` ó
- ▶ `git flow feature pull <origin> <name>`

# Crear release

Para list/start/finish release branches, usar:

- ▶ `git flow release`
- ▶ `git flow release start <release> [<base>]`
- ▶ `git flow release finish <release>`

Ejemplo desde el branch `feature/CA-007`

```
git flow release start oca-1.0.0 develop
```

# Crear hotfix

Para list/start/finish hotfix branches, usar:

- ▶ `git flow hotfix`
- ▶ `git flow hotfix start <release> [<base>]`
- ▶ `git flow hotfix finish <release>`

Ejemplo desde el branch `feature/CA-007`

```
git flow hotfix start oca-007 master
```

# Bibliografía I



"Git flow." <https://github.com/nvie/gitflow>.

Accessed: 2019-08-25.

**Gracias...**

# Como funciona Git flow

Un ejemplo de como funciona esto