



PERFECCIÓN
DIGITAL

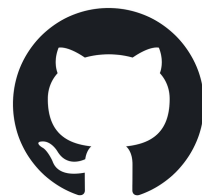


Manejo de **ramas** en **Git** de manera **profesional**

Juan Kuga
Expositor



git



GitHub

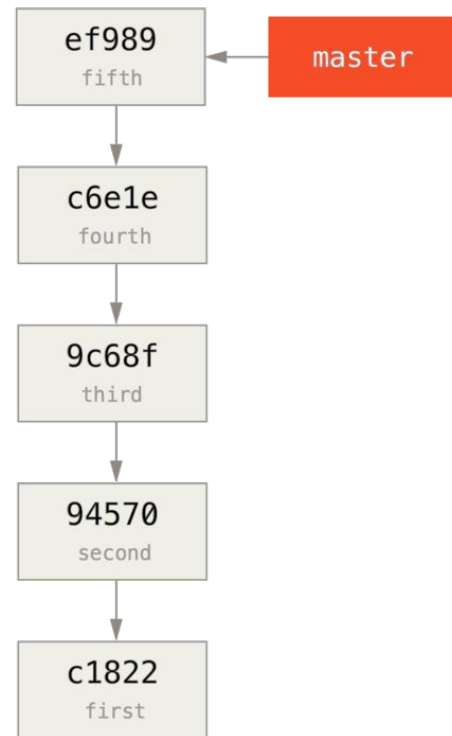
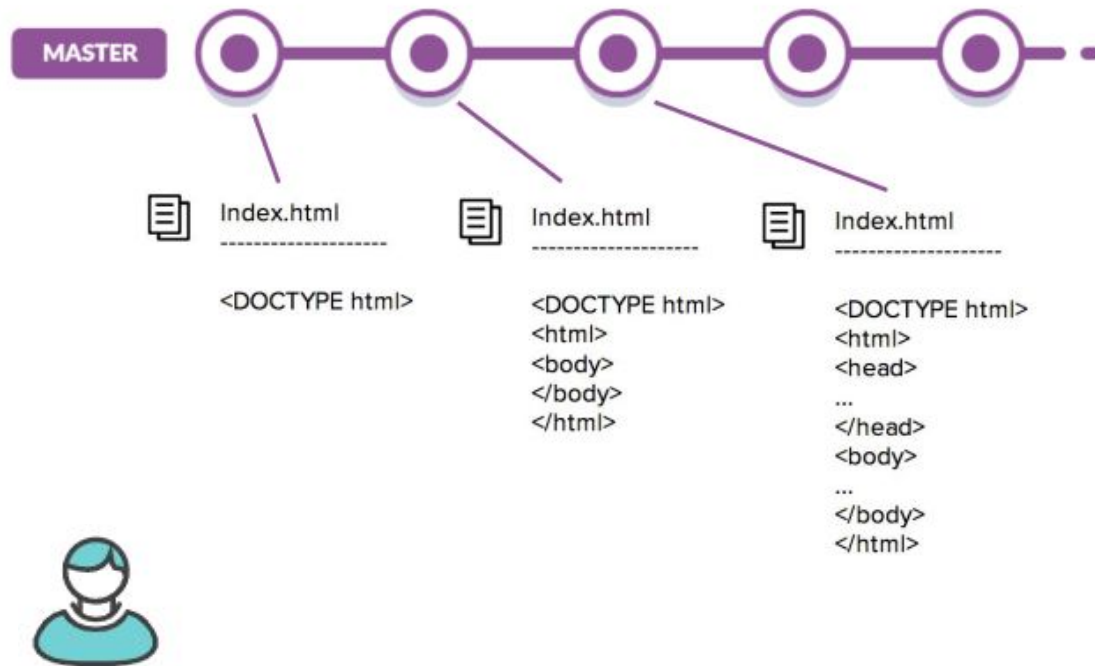


Agenda

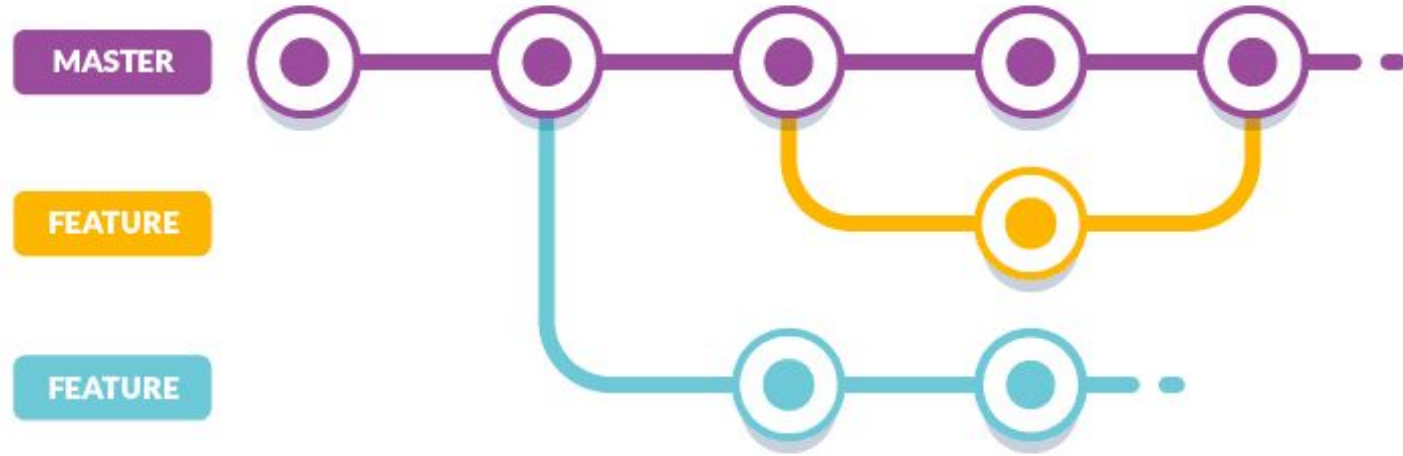
Taller: ¿Cómo revertir o deshacer un cambio en git?

- ¿Que es una rama?
- Ramas de funciones
- ¿Que es un WorkFlow?
- ¿Que es Git Flow?
- Recomendaciones
- Demo

¿Que es una rama?



Ramas de funciones o Feature Branch





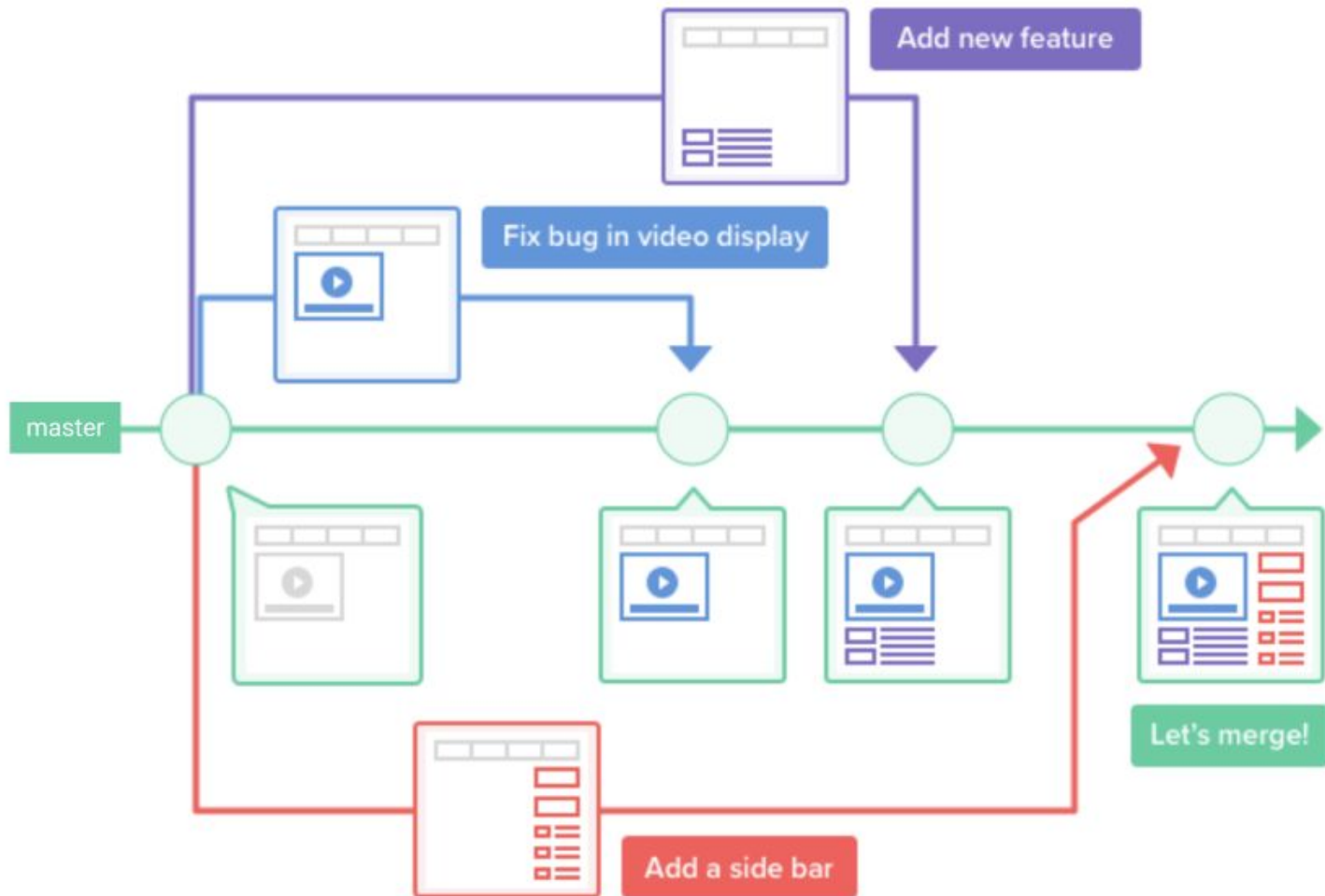
Juan



Maria



Pedro



¿Qué es
Git Workflow?

GitHub Flow

GitLab Flow

Git Flow

Release Flow
(Microsoft)

Master-only Flow

Trunk-Based Development

GitHub Flow

GitLab Flow

Git Flow

Master-only Flow

Release Flow
(Microsoft)

Trunk-Based Development

Git Flow

CLI

Descargar

MAC OS

```
$ brew install git-flow
```

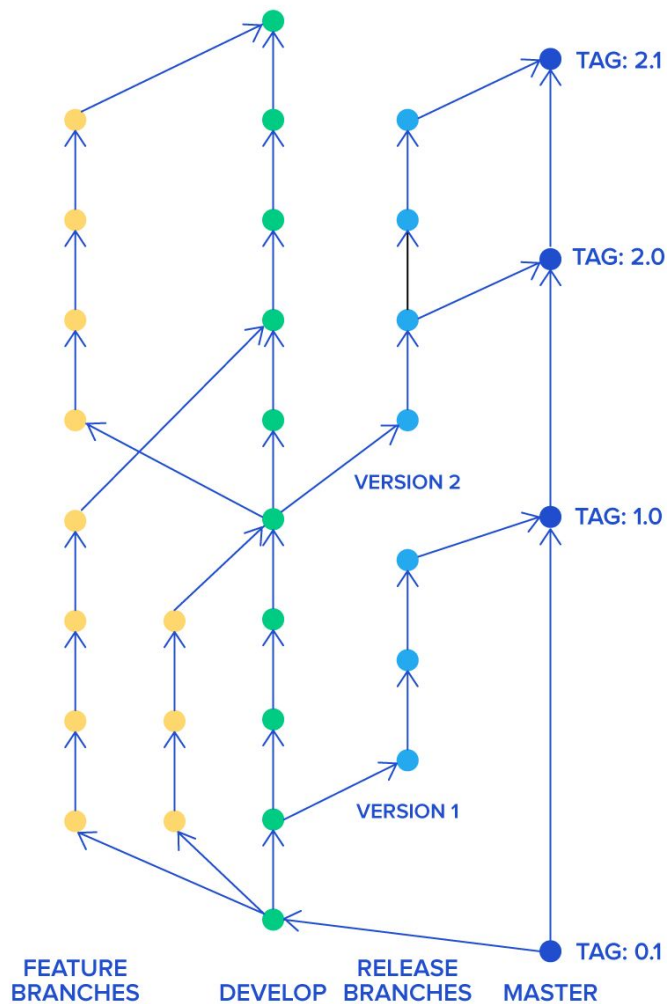
LINUX

```
$ apt-get install git-flow
```

Windows

```
> source tree
```

Git Flow



Git Flow trabaja con 5 ramas

1. master
2. develop
3. feature
4. release
5. hotfix

Ramas de Git Flow

master: Esta es la rama que mantiene el estado de producción. NUNCA deberías trabajar en esta rama. De estas ramas se inician las demás.

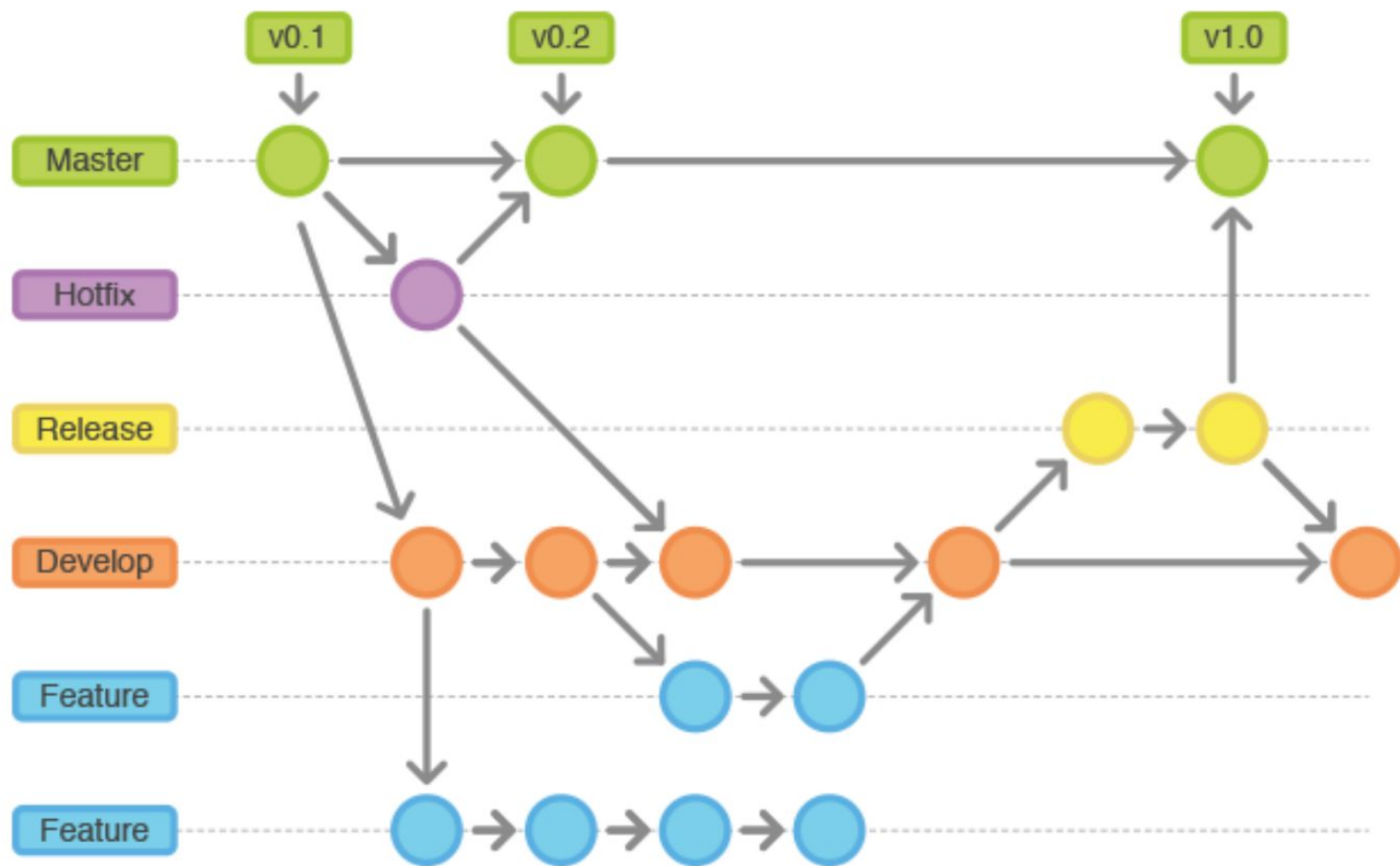
develop: esta es la rama de desarrollo principal desde donde se crean las versiones y características. parte de la rama **master**

feature/{branch-name}: Las ramas de características son aquellas en las que integraría una nueva característica / componente / etc ... Una vez que su trabajo haya finalizado, se fusionará nuevamente con el desarrollo.

Ramas de Git Flow

release/{branch-name}: Las ramas de lanzamiento son las que preparará una nueva versión que contiene múltiples funciones. Una vez que su trabajo haya finalizado, se fusionará nuevamente en la rama de desarrollo, así como en la principal con una nueva etiqueta (generalmente un nombre de versión semver).

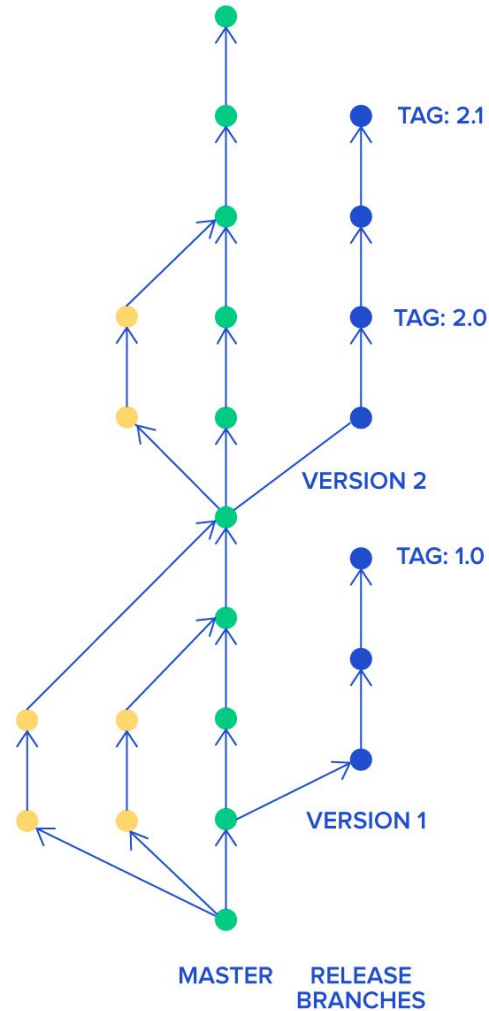
hotfix/{branch-name}: Las ramas de revisión se utilizan para arreglar algo rápidamente. Comienza desde la rama maestra y se fusiona nuevamente en la rama maestra con una nueva etiqueta, así como la rama de desarrollo.



Recomendaciones

- No existe un flujo de trabajo de Git universal
- Un flujo de trabajo debe ser sencillo y mejorar la productividad de tu equipo
- Tus requisitos empresariales deben servir para dar forma al flujo de trabajo de Git

Trunk-Based Development



¡GRACIAS!



IG: @perfecci0ndigital

FB: @perfecci0ndigital