

# cheatsheet do git-flow

criado por [Daniel Kummer](#)

Tweetar

sistema de branches eficiente usando o git-flow, do [Vincent Driessen](#)

traduções: [English](#) - [Castellano](#) - [Português Brasileiro](#) - [繁體中文](#)  
([Traditional Chinese](#)) - [简体中文](#)([Simplified Chinese](#)) - [日本語](#) - [Türkçe](#) - [한국어](#)([Korean](#)) - [Français](#) - [Italiano](#) - [Nederlands](#) - [Русский](#)([Russian](#)) -  
[Deutsch](#)([German](#)) - [Català](#)([Catalan](#)) - [Română](#)([Romanian](#)) - [Ελληνικά](#)  
([Greek](#)) - [Українська](#)([Ukrainian](#)) - [Tiếng Việt](#)([Vietnamese](#)) - [Polski](#) - [العربية](#)  
[فارسی](#) - [Azərbaycanca](#)([Azerbaijani](#)) [Bahasa Indonesia](#)

## Sobre

O git-flow é um conjunto de extensões para o git que provê operações de alto-nível para repositórios usando o modelo de branches do Vincent Driessen. [mais](#)



Este cheatsheet demonstra o uso básico e o efeito das operações do git-flow



## Dicas básicas

- ★ O git-flow oferece na linha de comando ajuda e saídas excelentes. Leia atentamente para ver o que está acontecendo...
- ★ O cliente do macOS [Sourcetree](#) é um excelente gui para o git e tem suporte ao git-flow



# Instalação

- ★ Você precisa do git instalado como pré-requisito.
- ★ O git flow funciona no macOS, Linux e no Windows



## macOS

Homebrew

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

Macports

```
$ port install git-flow-avh
```

## Linux

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

## Windows (Cygwin)

```
$ wget -q -O - --no-check-certificate  
https://raw.githubusercontent.com/  
petervanderdoes/gitflow-  
avh/develop/contrib/git-  
flow-installer.sh  
install stable | bash
```

Para instruções detalhadas sobre a instalação, visite a [wiki do git-flow](https://danielkummer.github.io/git-flow-cheatsheet/index.pt_BR.html).



# Começando

O git-flow precisa ser inicializado para personalizar a configuração de seu projeto.



## Inicialização

Comece o uso do git-flow fazendo sua inicialização dentro de um repositório git existente:

```
git flow init
```

Você precisa responder algumas questões relativas às convenções de nomenclatura dos seus branches. É recomendado que sejam usados os valores padrões.

## Funcionalidades/features

- ★ Desenvolva novas funcionalidades para as versões futuras
- ★ Normalmente existem apenas nos repositórios dos desenvolvedores

★ ★ ★

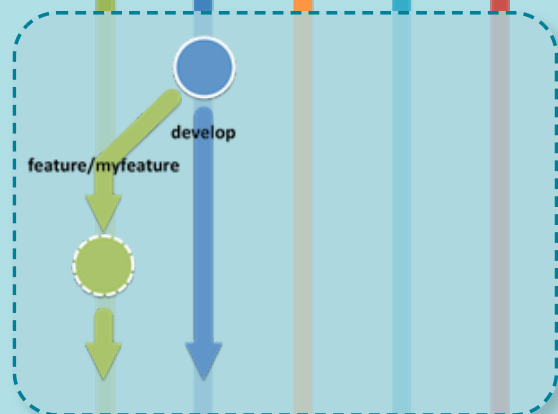
## Começar uma nova funcionalidade

O desenvolvimento de novas funcionalidades começa no branch 'develop'.

Comece o desenvolvimento de uma nova funcionalidade com

```
git flow feature start MYFEATURE
```

Esse comando cria um novo branch da funcionalidade baseado no 'develop' e alterna para ele



## Finalizar uma funcionalidade

Finaliza o desenvolvimento de uma funcionalidade. Esse comando faz o seguinte

- ★ Mescla MYFEATURE no 'develop'
- ★ Remove o branch da funcionalidade
- ★ Volta para o branch 'develop'

```
git flow feature finish MYFEATURE
```

## Publicar uma funcionalidade

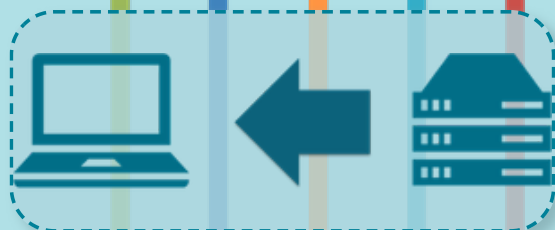
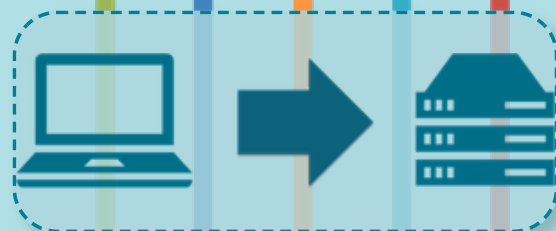
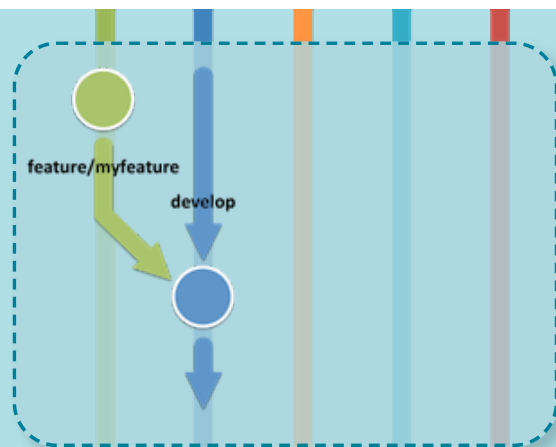
Você está desenvolvendo uma funcionalidade colaborativamente? Publique uma funcionalidade para o servidor remoto, assim ela pode ser utilizada por outros usuários.

```
git flow feature publish MYFEATURE
```

## Obter uma funcionalidade publicada

Obtenha uma funcionalidade publicada por outro usuário e acompanhe as alterações remotas.

```
git flow feature pull MYFEATURE
```



# *Criar uma versão/release*

- ★ Auxilia a preparação de uma nova versão de produção
- ★ Permite correções de bugs menores e a preparação de metadados de uma versão

★★★

## *Começar uma versão*

Para começar uma versão, use o comando `git flow release`. Ele cria um branch da versão baseado no branch 'develop'.

```
git flow release start  
RELEASE [BASE]
```

Você pode opcionalmente fornecer um hash sha-1 do commit `[BASE]` de onde começar a versão. O commit precisa estar no branch 'develop'

★★★

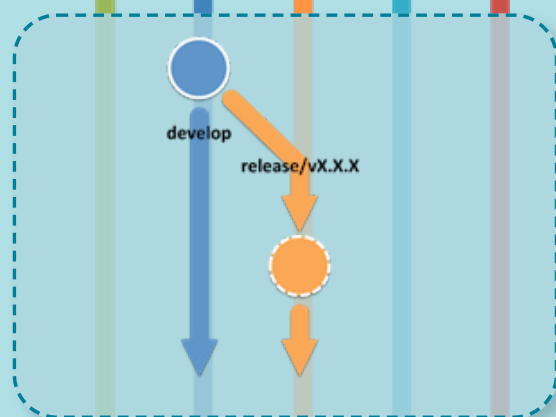
É sensato publicar o branch da versão depois de criá-lo, para permitir commits por outros desenvolvedores. É semelhante à publicação de uma funcionalidade com o comando:

```
git flow release  
publish RELEASE
```

(Você pode acompanhar uma versão remota com o comando

```
git flow release track RELEASE
```

)

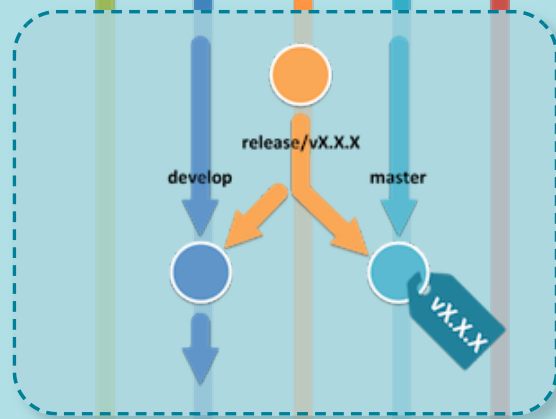


## Finalizar uma versão

A finalização de uma versão é um dos grandes passos na ramificação/branching do git. Ele executa várias ações:

- ★ Mescla o branch da versão no 'master'
- ★ Etiqueta a versão com seu nome
- ★ Mescla o branch da versão de volta no 'develop'
- ★ Remove o branch da versão

```
git flow release finish  
RELEASE
```



## Hotfixes

- ★ Os hotfixes surgem da necessidade de agir imediatamente sobre uma situação indesejada na versão de produção ativa
- ★ Pode ser criado a partir da tag correspondente no branch master que indica a versão em produção.

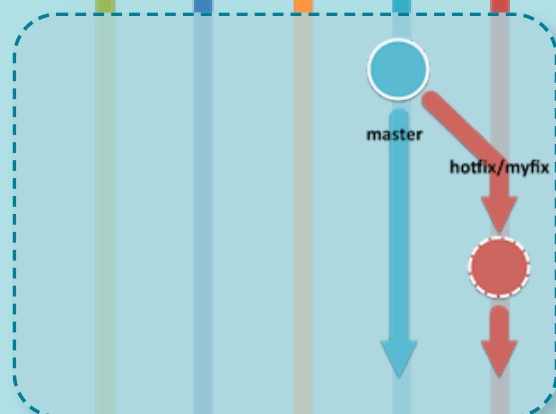
★★★

### git flow hotfix start

Assim como os outros comandos do git flow, um hotfix inicia com

```
git flow hotfix start  
VERSION [BASENAME]
```

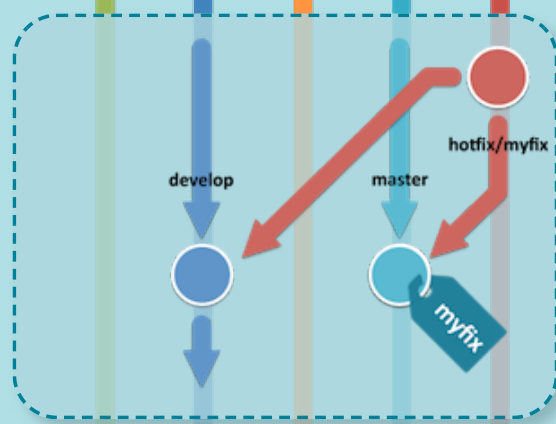
O argumento version marca o nome do novo hotfix. Opcionalmente, você pode especificar um basename para começar.



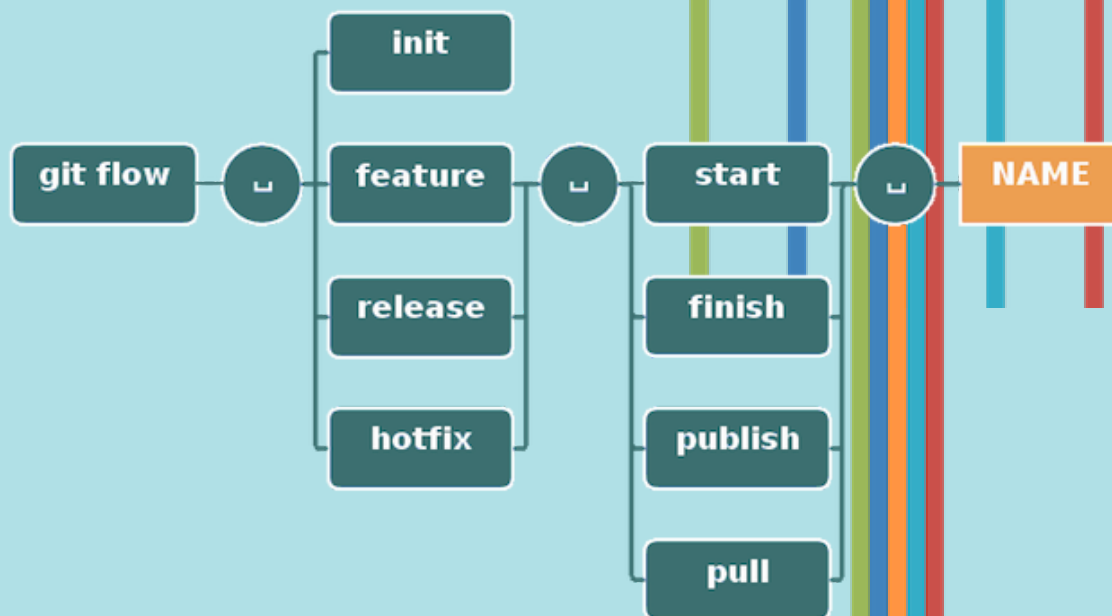
## Finalizar um hotfix

Ao finalizar um hotfix ele é mesclado tanto no develop quanto no master. Além disso, o merge no master é etiquetado.

**git flow hotfix finish  
VERSION**



## Comandos



## Backlog



- ★ Nem todos os comandos disponíveis estão cobertos aqui, apenas os mais importantes deles
- ★ Você pode continuar a usar o git e todos os comandos dele normalmente como você já conhece, o git flow é apenas uma coleção de ferramentas
- ★ A funcionalidade 'support' continua no estágio beta, seu uso não é indicado

★ Se você quiser fornecer traduções, ficarei feliz de integrá-las



# *Comentários*

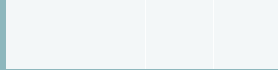


23 Comments

 Login ▾

Join the discussion...

LOG IN WITH

OR SIGN UP WITH DISQUS 

Name



18

• Share

Best

Newest

Oldest

**Eduardo Moreira de Lima** 

6 years ago

O comando para obter uma funcionalidade como está no cheatsheet foi depreciado na versão 2.0.0. Agora devemos usar "git flow feature track MYFEATURE"  
Sugiro essa correção.



14



0

Reply

• Share &gt;

**Camilo Vitorino da Costa** 

5 years ago

A sua página explicando o plugin é a melhor representação de um Git flow que eu já vi!  
Não usarei o plugin pois uso o SourceTree, mas com certeza iremos melhorar o nosso Gitflow.

Obrigado!



6



0

Reply

• Share &gt;

**Rafael Corrêa Gomes** 

6 years ago

Muito bom!



2



0

Reply

• Share &gt;

**Antônio Júnior** 

3 years ago

A didática foi a melhor que já encontrei, parabéns show.



0



0

Reply

• Share &gt;

**Marco Lima** 

3 years ago

Muito bom, mostra de forma clara como é um Git Flow, está muito intuitivo e de fácil entendimento, Parabéns!!!



0



0

Reply

• Share &gt;

**khalleb Ribeiro** 

4 years ago

Show



0



0

Reply

• Share &gt;

**Aislan Maia** 

4 years ago

Onde é que está o projeto "oficial", já que existem vários forks mortos por aí.

0 0 Reply • Share ›



**Aislan Maia**

→ Aislan Maia

4 years ago

Achei este site aqui que cobre muita coisa mais atualizada que este cheatsheet.

<https://www.atlassian.com/g...>

0 0 Reply • Share ›



**Natanael Fonseca**

5 years ago

Excelente explicação!

0 0 Reply • Share ›



**Salesio Gomes**

5 years ago

Muito bom!

0 0 Reply • Share ›



**Jônatas Rodrigues**

5 years ago

Excelente amigo. Parabéns.

0 0 Reply • Share ›



**Roberto Ramos**

5 years ago

Muito bom!!! Obrigado, me ajudou bastante!

0 0 Reply • Share ›



**Dreyfi Ferreira**

5 years ago

zika da balada

0 0 Reply • Share ›



**Raffaello**

6 years ago

maneiro

0 0 Reply • Share ›



**Renato Lucena**

6 years ago

Top.

0 0 Reply • Share ›



**Gudang**

6 years ago

Para instalar em distros Linux CentOS o comando é yum install git-flow

0 0 Reply • Share ›

**Gabriel Rodrigues**

6 years ago

Muito obrigado.

0 0 Reply • Share ›

**Herlon Aguiar**

6 years ago

Show!

0 0 Reply • Share ›

**Nayara Valadares**

7 years ago

Muito bom!!

0 0 Reply • Share ›

**Valdeir S.**

7 years ago

Show demais, estou começando a utiliza-lo e gostei bastante.

0 0 Reply • Share ›

**Maykon Oliveira**

4 years ago edited

Esse site devia ser mais moderno ^-^

0 2 Reply • Share ›

**Thiago Ghelere** → Maykon Oliveira 

4 years ago

Nada nos impede de contribuir com um visual mais moderno, é um projeto open-source e aceita pull requests :D

5 0 Reply • Share ›

**Diego** → Maykon Oliveira 

4 years ago

Esse site está cumprindo muito bem seu propósito. Não vejo "falta de modernidade" nele.

2 0 Reply • Share ›



Subscribe

Privacy

Do Not Sell My Data