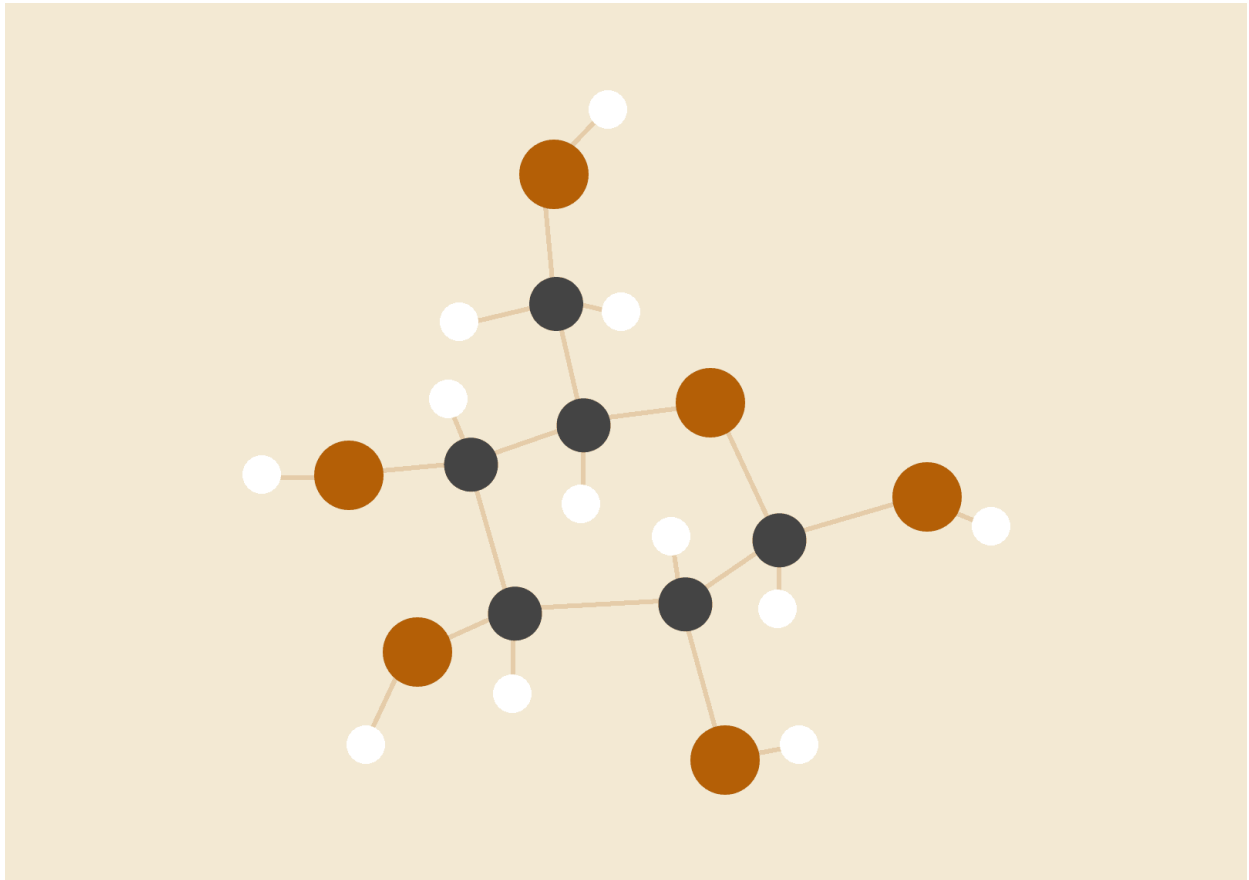


# CONCEITOS E PRÁTICA GIT FLOW



**Guilherme Carvalho da Silva**

14.04.2021

GRUPO GO AHEAD STUDY GROUP

## INTRODUÇÃO

O Git Flow é apenas uma ideia abstrata do fluxo de trabalho, sendo uma extensão do Git. Isto significa que ele dita que tipos de ramificações configurar e como fazer a mesclagem. Vamos falar sobre os objetivos das ramificações abaixo. O conjunto de ferramentas git-flow é uma ferramenta de linha de comando, muito utilizada atualmente para fazer o gerenciamento de atualizações de recursos em projetos.

## AS BRANCHES PRINCIPAIS.

As branches principais, são aqueles que tem a vida útil infinita. ao decorrer do desenvolvimento e mesclagem de branches, eles continuam intacto e sem nenhuma exclusão, são eles:

- **Master**
- **Develop**

## BRANCHES DE SUPORTE

- **Feature**
- **Release**
- **HotFix**

## CONCEITO SOBRE AS BRANCHES E SUAS PRINCIPAIS FUNÇÕES

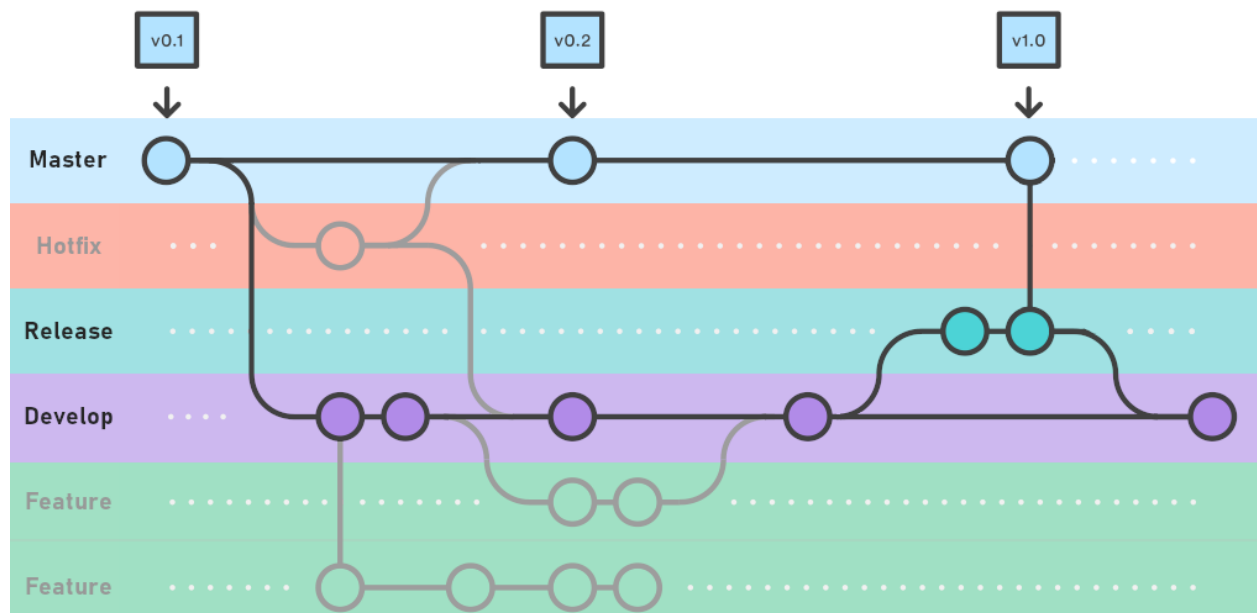
**Master** - Branch principal. Não alterada diretamente sendo o ambiente de produção. Status sempre ativo.

**Develop** - Branch que recebe primeiramente os projetos, servindo de intermediário para as features e releases, conhecida também como ramo de integração. Status sempre ativo.

**Feature** - Onde é feita as alterações e implementações de novos recursos no projeto. Com término das alterações e fechamento da feature, automaticamente é feito o merge com a branch Develop. A feature é efêmera, sendo automaticamente excluída ao término de seu uso.

**Release** - Ambiente pré produção/homologação onde é feito testes nas alterações para aprovação. Release tem o mesmo conceito da feature, após o término de seu uso, é automaticamente excluída.

**HotFix** - As ramificações de manutenção ou de “hotfix” são usadas para corrigir com rapidez lançamentos de produção. As ramificações de hotfix se parecem com ramificações de release e de feature, com a diferença de serem baseadas a partir da branch master, para não ter que passar por todo processo novamente.



## ● TRADUÇÃO DE ALGUMAS PALAVRAS

PALAVRAS	TRADUÇÃO
BRANCH	RAMO/RAMIFICAÇÃO
FEATURE	RECURSO/CARACTERISTICA
MERGE	FUNDIR/MISTURAR
RELEASE	LIBERAÇÃO/LANÇAMENTO

## MÃO NA MASSA

### Instalação do Git Flow.

#### Instalando no Ubuntu 18.04

```
sudo apt-get update  
sudo apt-get upgrade  
sudo apt-get install git-flow
```

**(Vou deixar o link no final do documento para instalação do git flow no windows e Mac).**

Linha de comando para executar o **git flow**

**git flow init** - Para dar início no repositório.

**git branch** - Ver suas branches, **ao iniciar automaticamente você começa na branch de develop.**

**git flow feature start (nome da feature)** - Criando uma feature.

(Como estou usando o ubuntu, irei usar o editor de texto vim, para simular uma alteração em algum arquivo)

**vim soma**

"alterando recurso soma, atualização da feature"

**git add .** - Adicionando as alterações feitas ao repositório.

**git commit -m 'Alterando recurso soma'** - Fazendo o commit da alteração.

**git flow feature finish (nome da feature)** - Fechando a feature e fazendo o merge automaticamente com a develop.

Aparece para adicionar comentário no **merge da feature e develop**, com o editor de arquivos Nano, comandos **CTRL + S** para salvar e **CTRL + X** sair do arquivo.

**git flow release start (tag)** - Abrindo uma release, para execução de testes das alterações feitas.

**git flow release finish (tag)** - Fechamento da release e enviando para a Master. Colocando comentários da mudança feita e comentários sobre a tag.

## RESUMO

**O fluxo geral do Git Flow é:**

1. Uma branch de develop é criada a partir da branch master.
2. Uma branch de feature é criada a partir da branch de develop.
3. Quando uma feature é concluída, ela é mesclada na branch de develop.
4. Ramificações de release são criadas a partir da branch de develop.
5. Quando a branch de release é concluída, ela é mesclada nas branches de develop e master.

6. Caso um problema seja detectado na branch master, uma ramificação de hotfix é criada a partir da master.
7. Após a conclusão da branch de hotfix, ela é mesclada para as ramificações de develop e master.

## REFERÊNCIAS

**Download git-flow Windows:** <https://git-scm.com/download/win>

**Atlassian:** <https://www.atlassian.com/br/git/tutorials/comparing-workflows/gitflow-workflow#:~:text=O%20conjunto%20de%20ferramentas%20git,tem%20um%20processo%20de%20instala%C3%A7%C3%A3o.&text=O%20Git%2Dflow%20%C3%A9%20um,ser%20criar%20ramifica%C3%A7%C3%B5es%20para%20voc%C3%AA>

**Angelo Luz:** <https://www.youtube.com/watch?v=wzxBR4pOTTs>

**Codigo Fonte:** <https://www.youtube.com/watch?v=oweffeS8TRc>