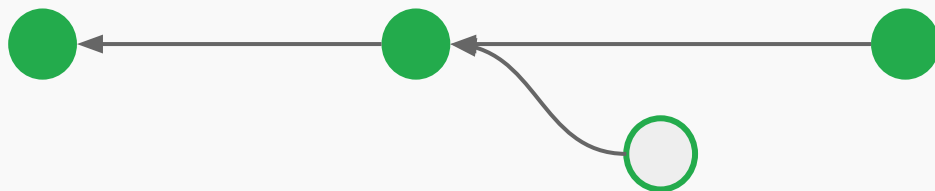


Github



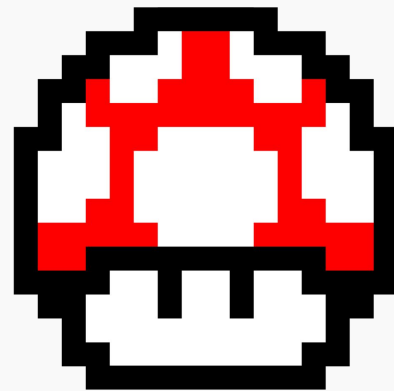
Versionando projetos de maneira
eficiente e compreensiva

Por Júnior Mendes - [@jrmmendes](#)



Júnior Mendes

Desenvolvedor @ **CITi**
junior.mendes@citi.org.br



THE
MATRIX

INCEPTION

city 友

Material disponível em:



piig.me/citicamp-github

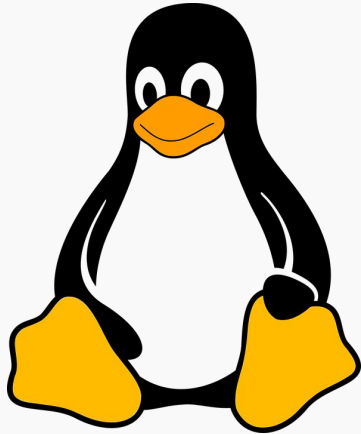
Setup Inicial



Versionamento e Trabalho em equipe



Kernel Linux:



+18 MI de linhas

+15 Mil contribuidores



O Git é um sistema de **controle de versões distribuído**, usado principalmente no **desenvolvimento de software**, mas pode ser usado para registrar o histórico de edições de qualquer tipo de arquivo.

Documentação: piig.me/git

Por que usar o Git?

Velocidade

Não-linearidade

Design Simples

Totalmente distribuído

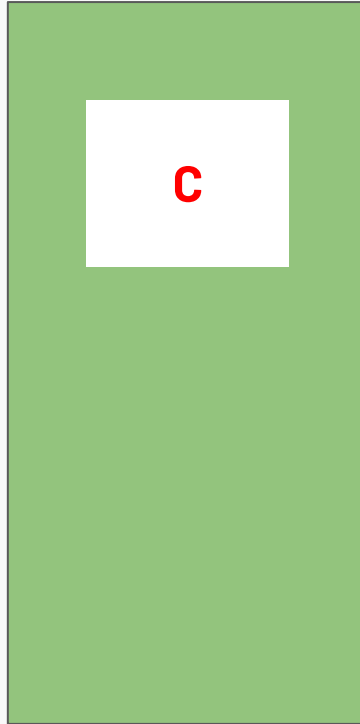
Escalável



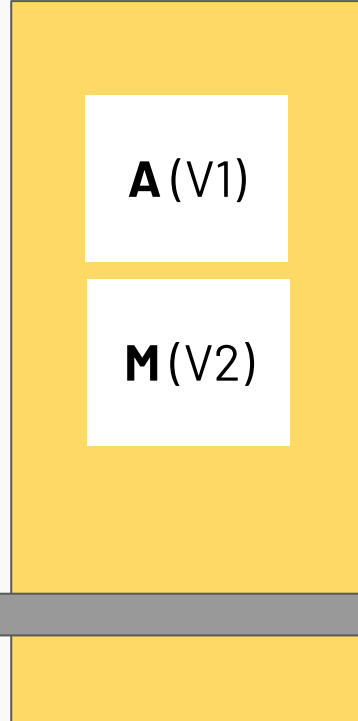
Como funciona o Git?



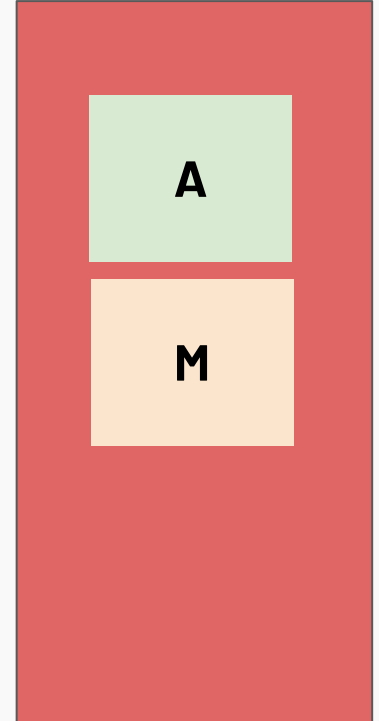
Repositório



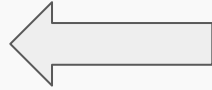
Index(Stage)



Diretório de Trabalho



Commit



Add



Checkout



Repositório Local



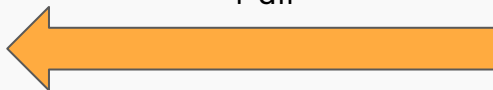
Push



Repositório Remoto



Pull



Plataformas de hospedagem



GitHub

GitHub é uma **plataforma** de hospedagem de código-fonte com controle de versão usando o **Git**. Ele permite que programadores, utilitários ou qualquer usuário cadastrado na plataforma **contribuam em projetos**.

Página inicial: github.com





GitLab



cit

Operações básicas



Git Básico

Add

Commit

Push

Pull & Fetch

Branch

Adiciona o conteúdo de arquivos ou diretórios ao Index (Staging).

```
git add <arquivos|diretórios>
```



Git Básico

Add

Commit

Push

Pull & Fetch

Branch

Registra as modificações adicionadas
Index no repositório.

```
git commit -m "mensagem do commit"
```



Git Básico

Add

Commit

Push

Pull & Fetch

Branch

Envia as modificações do repositório local para o remoto

```
git push <remoto> <branch>
```



Git Básico

Add

Commit

Push

Pull & Fetch

Branch

Recebem as modificações do remoto, sendo que o primeiro baixa também os arquivos modificados.

```
git pull <remoto> <branch>
```

```
git fetch <remoto> <branch>
```



Git Básico

Add

Commit

Push

Pull & Fetch

Branch

Lista, cria e deleta branches.

```
git branch
```

```
git branch -d <nome-da-branch>
```



Github

Issues

Pull Requests

Forks



Github

Issues

Pull Requests

Forks

Pull Requests



Github

Issues

Pull Requests

Forks

Forks



Trabalhando com Branchs

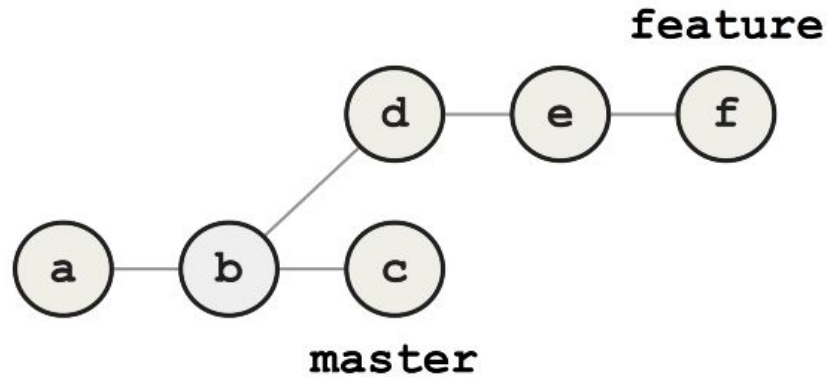


Branchs

Merge

Rebase

Conflitos

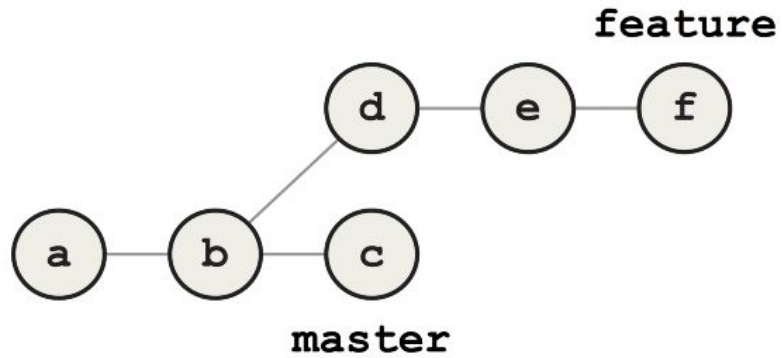


Branchs

Merge

Rebase

Conflitos



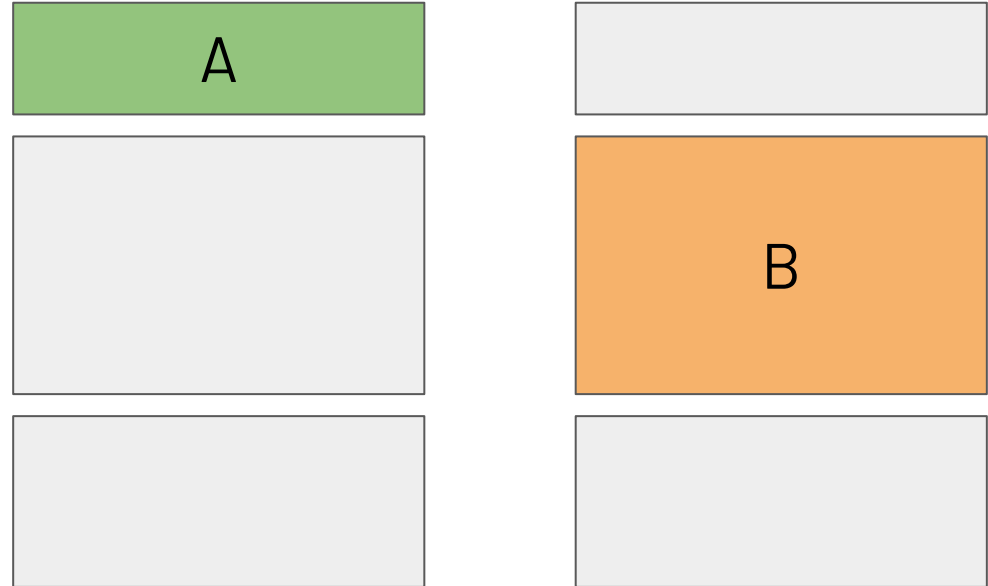
Branchs

Merge

Rebase

Conflitos

Situação normal



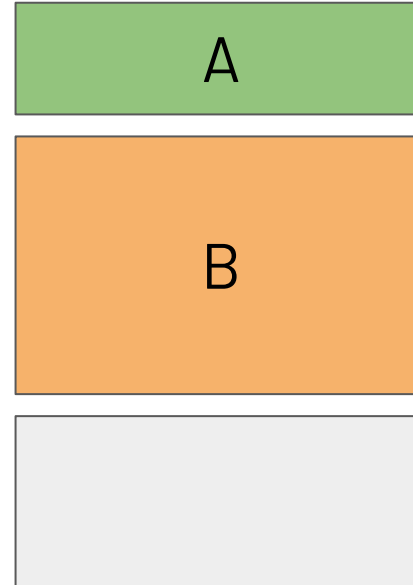
Branchs

Merge

Rebase

Conflitos

Situação normal



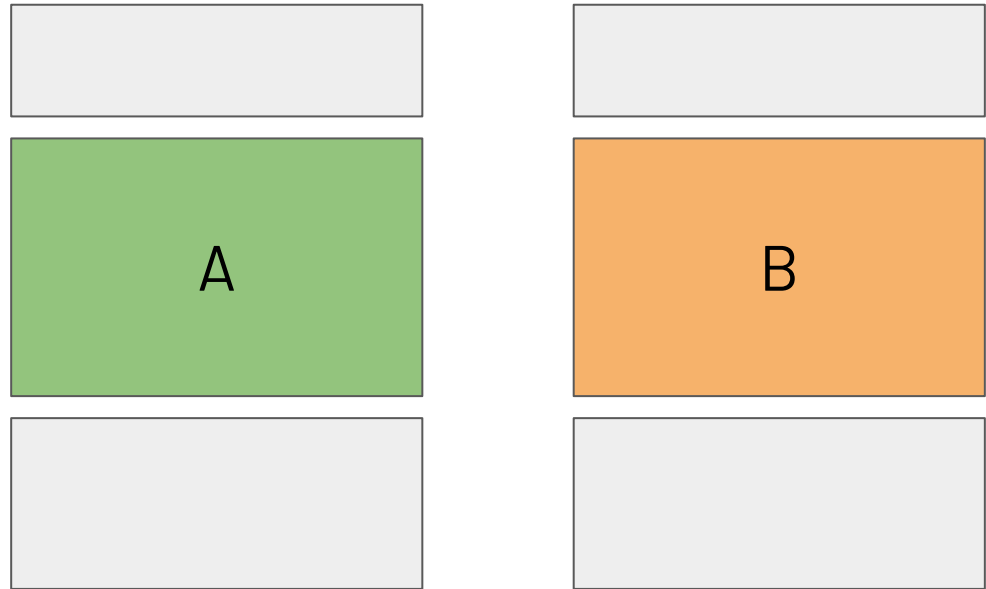
Branchs

Merge

Rebase

Conflitos

Qual escolher?



```
git checkout <branch>
```

```
git checkout -b <novo-branch>
```


Desfazendo
coisas



Desfazendo

Revert

Reset

Checkout

Cria um commit com modificações opostas a do commit que se deseja modificar

```
git revert <commit>
```



Desfazendo

Revert

Reset

Checkout

Modifica a posição do HEAD para algum commit específico

```
git reset [--soft|--hard] <commit>
```



Obtendo Informação



Desfazendo

Status

Diff

Log

Exibe o status dos arquivos no diretório de trabalho

```
git status
```



Desfazendo

Status

Diff

Log

Exibe as modificações realizadas nos arquivos

```
git diff <commit>
```



Desfazendo

Status

Diff

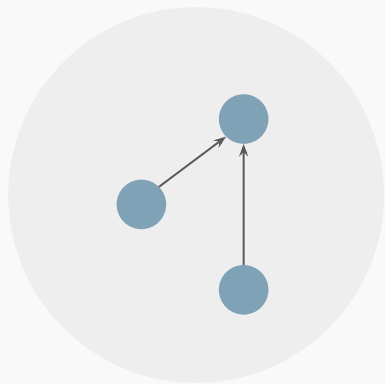
Log

Mostra o log de commits.

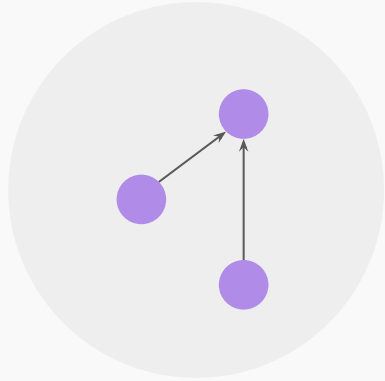
```
git log
```



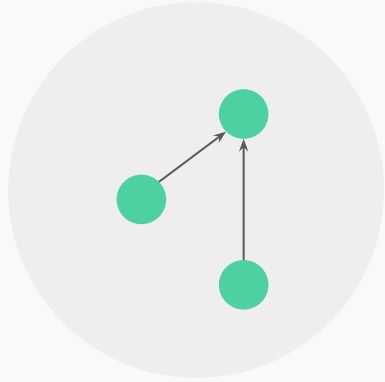
Fluxo de trabalho com o Gitflow



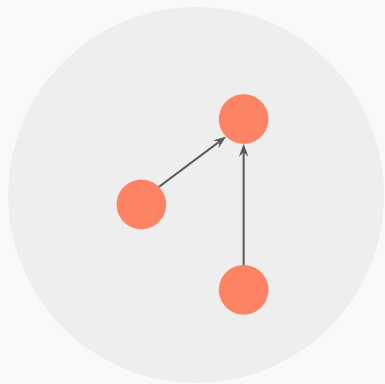
Master



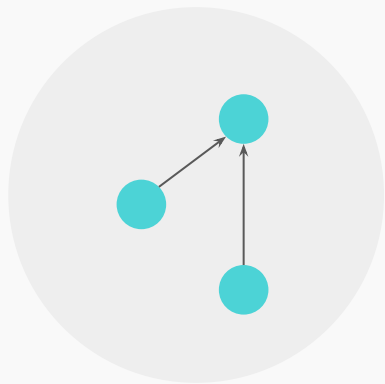
Develop



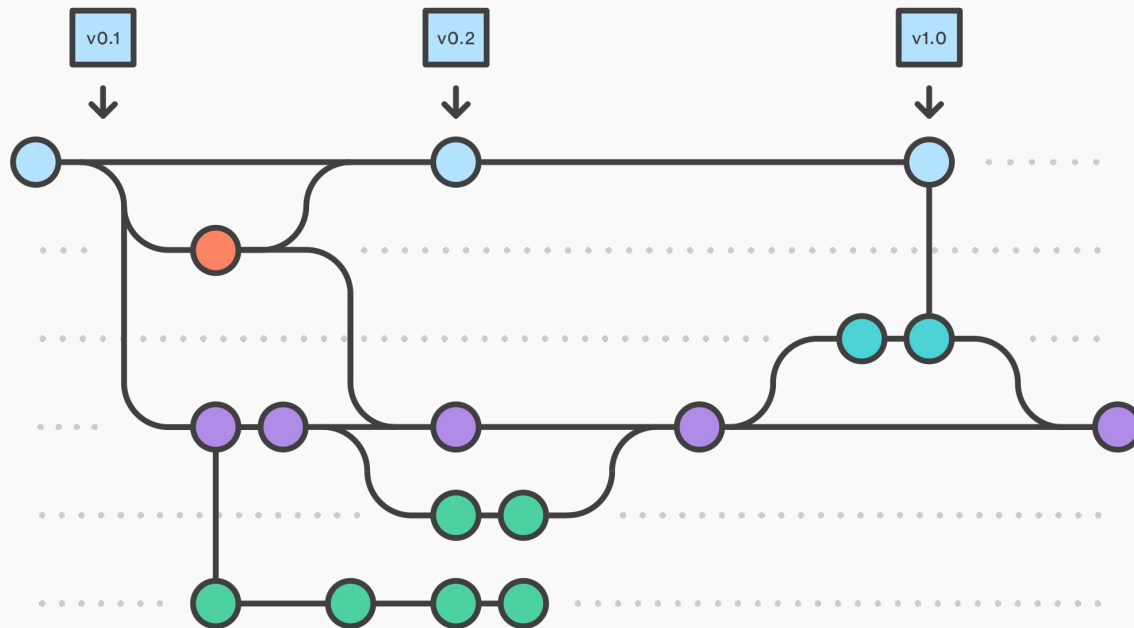
Feature



Hotfix



Release



**Bônus: 5 coisas legais
pro dia a dia**

#1: GUI's

#2: Editores com
boa integração

#3: Gist's

#4: Zenhub

#5: Gitignore.io



Obrigado!