



Git e GitHub

Introdução conceitual

André Estevam

a166348@dac.unicamp.br

Notas

1. Não sou especialista em Git, assistam com dúvidas e apontem possíveis erros;
2. O conteúdo foi desenvolvido com base em experiência pessoal, experimentação nas ferramentas e pesquisa na documentação do git;
3. Como este é um assunto essencialmente técnico, existem inúmeras fontes que discorrem sobre o mesmo assunto. Preferi deixar nos slides links para as fontes que utilizei ao invés de uma seção de bibliografia pois considero não confiável toda fonte que não seja a documentação oficial, assim, esses links devem ser interpretados como um “saiba mais” pois testei nas devidas ferramentas as funcionalidades que comento.

Objetivos

1. Mostrar os principais tópicos que envolvem Git/GitHub **conceptualmente**;
2. Motivar a **pesquisa** mais profunda pelas ferramentas;
3. Exemplificar os **benefícios** da ferramenta no dia a dia de um desenvolvedor.

Antes de tudo, um problema

Dangit, git!

Git is hard: screwing up is easy, and figuring out how to fix your mistakes is nigh on impossible.

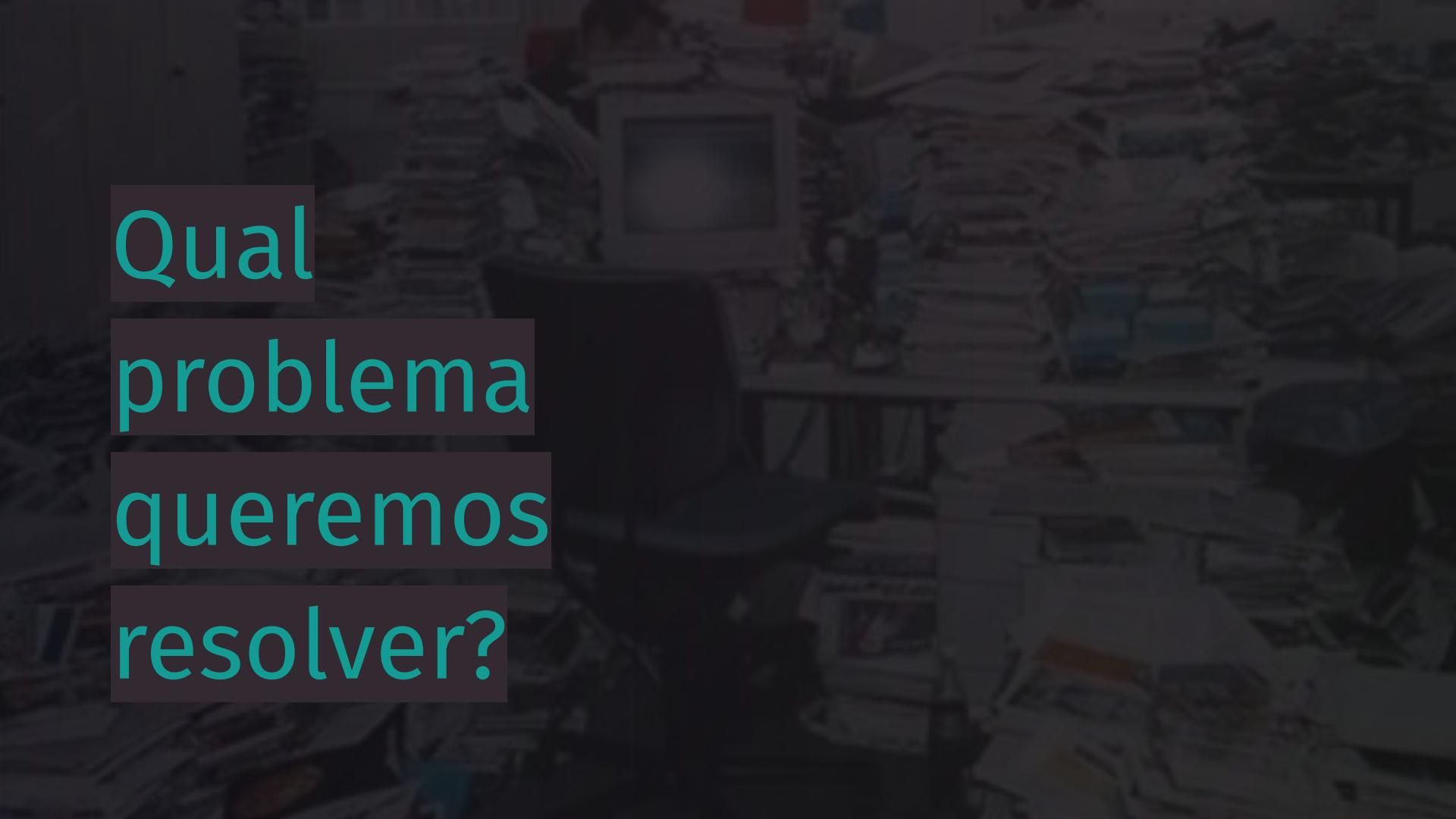
Git documentation has this chicken and egg problem where you can't search for how to get yourself out of a mess, unless you *already know the name of the thing you need to know about* in order to fix your problem.

So here are some bad situations I've gotten myself into, and how I've



Captura do site: <https://dangitgit.com/>

Gráfico do artigo: <https://medium.com/@afeinman/knowing-git-ab8f60550e2b>



A dark, grainy photograph of a person working at a computer terminal in a control room filled with monitors.

Qual
problema
queremos
resolver?



Search Mail

Search the Web

[Show search options](#)
[Create a filter](#)[Compose Mail](#)[Send](#)[Save Draft](#)[Discard Draft](#)[Inbox \(3\)](#)[Starred](#) [Sent Mail](#)[Drafts \(1\)](#)[All Mail](#)[Spam](#)[Trash](#)[Contacts](#)[Labels](#)[Edit labels](#)

To: <gustavo@bol.com.br> <anderson1@hotmail.com> <gabri.el@ gmail.com>

Cc:

Bcc:

Subject: Minha alterações no trabalho

Attachments:

- trabalho.doc - 1kb
- grad.ggb - 23kb
- Image URL.png - 3348kb

[Browse...](#) No file selected.[Attach More Files](#)

Estou enviando minha parte do trabalho

[Send](#)[Save Draft](#)[Discard Draft](#)Use the search box or **search options** to find messages quickly!

[Compose Mail](#)[Send](#)[Save Draft](#)[Discard Draft](#)[Inbox \(3\)](#)[Starred](#)[Sent Mail](#)[Drafts \(1\)](#)[All Mail](#)[Spam](#)[Trash](#)[Contacts](#)[Labels](#)[Edit labels](#)

To: <gustavo@bol.com.br> <anderson1@hotmail.com> <gabri.elia@gmail.com>

Cc:

Bcc:

Subject: Minha alterações no trabalho

Attachments:

- trabalho.doc - 1kb
- grad.ggb - 23kb
- Image URL.png - 3348kb

[Browse...](#) No file selected.[Attach More Files](#)

Estou enviando minha parte do trabalho

[Send](#)[Save Draft](#)[Discard Draft](#)Use the search box or [search options](#) to find messages quickly!

[Compose Mail](#)[Send](#)[Save Draft](#)[Discard Draft](#)[Inbox \(3\)](#)[Starred](#)[Sent Mail](#)[Drafts \(1\)](#)[All Mail](#)[Spam](#)[Trash](#)[Contacts](#)[Labels](#)[Edit labels](#)

To: <gustavo@bol.com.br> <anderson1@hotmail.com> <gabriela.elia@gmail.com>

Cc:

Bcc:

Subject: Minha alterações no trabalho

Attachments:

- trabalho.doc - 1kb
- grad.ggb - 23kb
- Image URL.png - 3348kb

[Browse...](#)

No file selected.

[Attach More Files](#)

Estou enviando minha parte do trabalho

[Send](#)[Save Draft](#)[Discard Draft](#)

Trabalho versão Final 4



Trabalho com correções do orientador 4



Trabalho versão Final 5



Trabalho com mais correções do orientador



Trabalho versão Final 3



Trabalho com correções do orientador 3





Drive



Pesquisar no Drive

Com.

Inbox

Starre

Sent M

Drafts

All Ma

Spam

Trash

Cont

Labels

Edit la



Novo

Meu Drive > BL203 > trabalho

[Show search options](#)
[Create a file](#)

Nome

trabalho - FINAL1FINAL

trabalho - FINAL1FINALAGORAVAI

trabalho - FINAL1FINALAGORAVAI - esse é o certo

trabalho - FINAL1FINALAGORAVAI - esse é o certo1

Cópia de parte gustavo 2.0

parte gustavo 1.0

trabalho FINAL1

trabalho - FINAL

BL203-1 partes juntas (correção Gustavo)

BL203 - partes juntas

Trabalho de BL203 - tópicos a discutir

artigo-util-ler.pdf

anderson-correção.docx

anderson - Cópia.docx



Trabalho com correções do orientador

Trabalho final 2

Trabalho versão Final 3

Trabalho com correções do orientador 3

Trabalho com correções do orientador 4

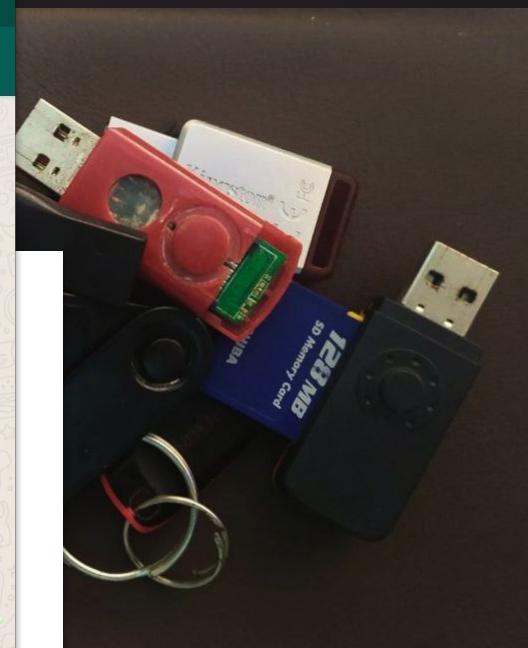
Trabalho versão Final 5

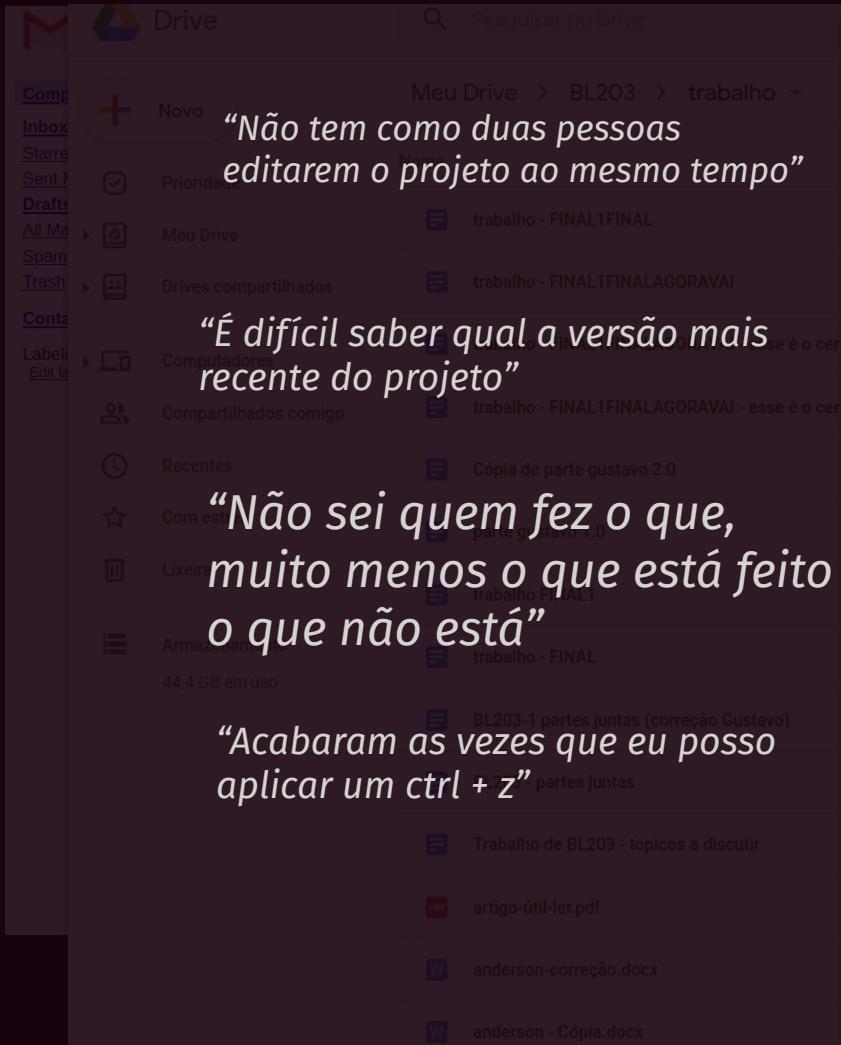
[Com.](#)[Inbox](#)[Starre](#)[Sent M](#)[Drafts](#)[All Ma](#)[Spam](#)[Trash](#)[Cont](#)[Labels](#)[Edit la](#)[Novo](#)

Meu Drive > BL203 > trabalho

Nome

- [trabalho - FINAL1FINAL](#)
- [trabalho - FINAL1FINALGORAVAI](#)
- [trabalho - FINAL1FINALGORAVAI - esse é o cer](#)
- [trabalho - FINAL1FINALGORAVAI - esse é o cer](#)
- [Cópia de parte gustavo 2.0](#)
- [parte gustavo 1.0](#)
- [trabalho FINAL1](#)
- [trabalho - FINAL](#)
- [BL203-1 partes juntas \(correção Gustavo\)](#)
- [BL203 - partes juntas](#)
- [Trabalho de BL203 - tópicos a discutir](#)
- [artigo-útil-ler.pdf](#)
- [anderson-correção.docx](#)
- [anderson - Cópia.docx](#)



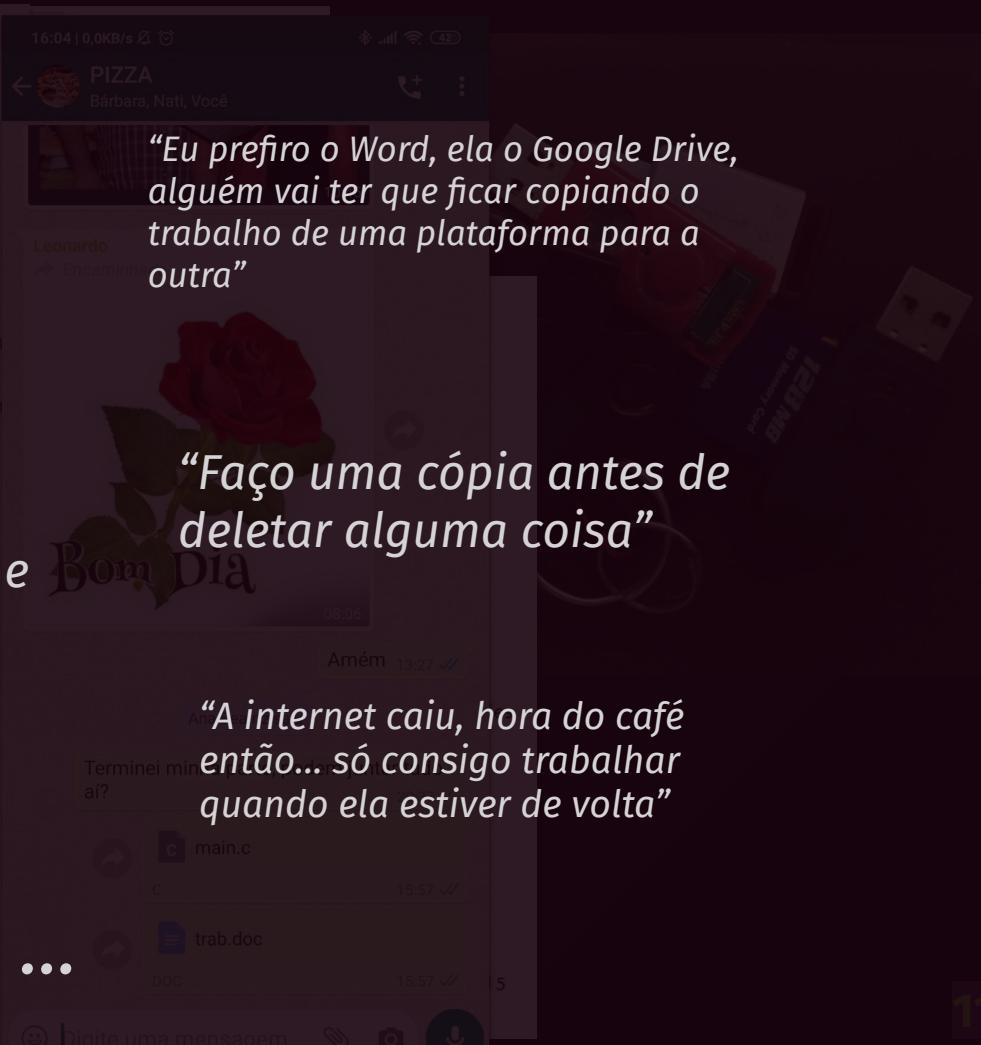
A screenshot of the Google Drive interface. On the left, there's a sidebar with navigation links like 'Inbox', 'Novo', 'Meu Drive', 'Drives compartilhados', 'Labels', 'Contas', and 'Ajuda'. The main area shows a list of files and folders, including 'trabalho - FINAL1FINAL', 'trabalho - FINAL1FINALAGORAVAI', 'Cópia de parte gustavo 2.0', and 'trabalho - FINAL'. A message bubble at the top right says "PIZZA" and "Bárbara, Nati, Você".

“Não tem como duas pessoas editarem o projeto ao mesmo tempo”

“É difícil saber qual a versão mais recente do projeto”

“Não sei quem fez o que, muito menos o que está feito e o que não está”

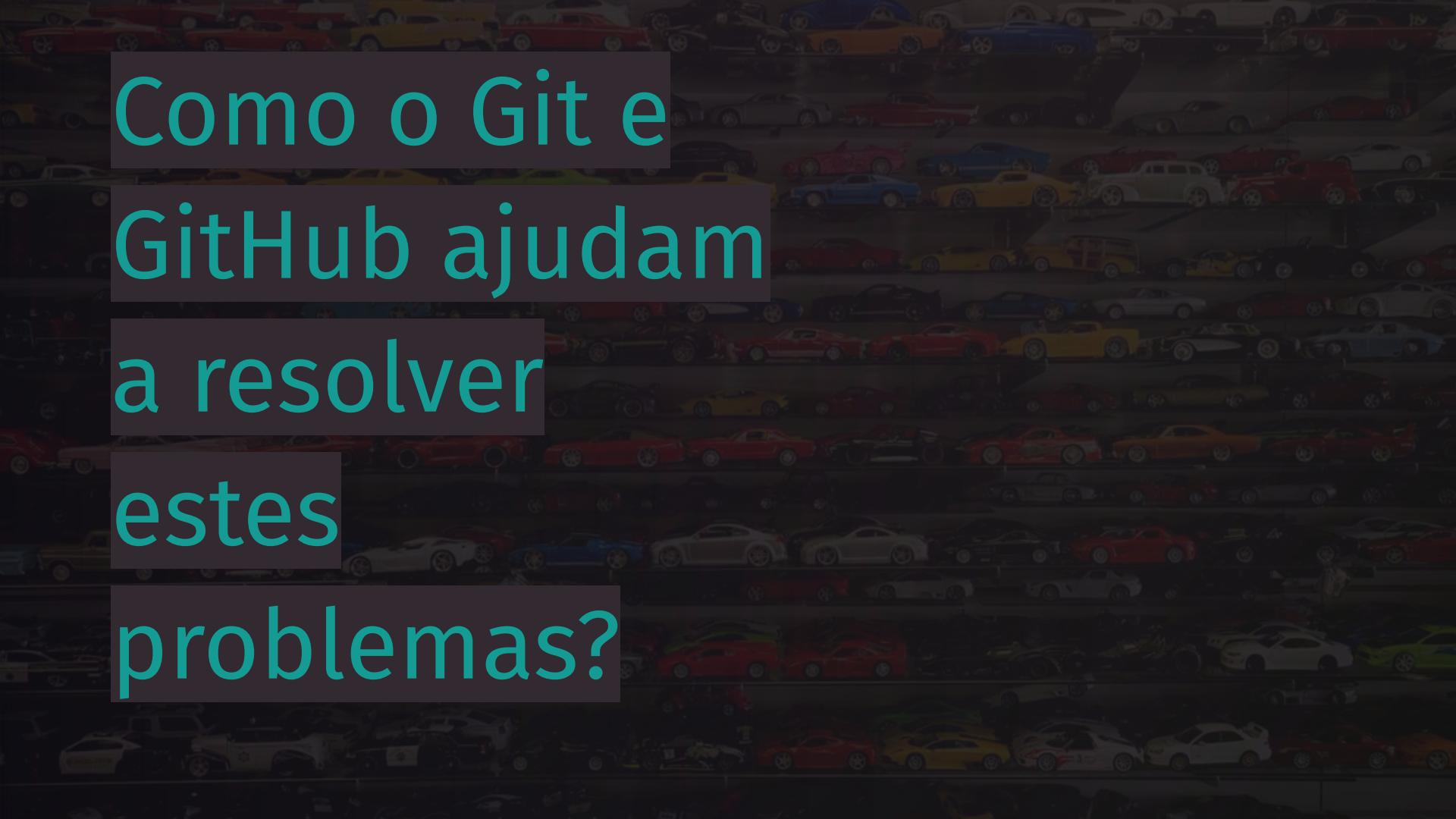
“Acabaram as vezes que eu posso aplicar um ctrl + z”

A screenshot of a messaging app. At the top, it says "16:04 | 0,0KB/s". The conversation starts with a message from "PIZZA" saying "Bárbara, Nati, Você". Below it, a message from "Leonardo" says "Encantada". A large red rose graphic is overlaid on the screen. A message from "Amém" at 13:27 says "Bom Dia". Another message from "Amém" at 13:27 says "Terminei min... ai?". At the bottom, there's a file viewer showing "main.c" and "trab.doc". A message input field says "Digite uma mensagem".

“Eu prefiro o Word, ela o Google Drive, alguém vai ter que ficar copiando o trabalho de uma plataforma para a outra”

“Faço uma cópia antes de deletar alguma coisa”

“A internet caiu, hora do café então... só consigo trabalhar quando ela estiver de volta”



Como o Git e
GitHub ajudam
a resolver
estes
problemas?

 **Organização de arquivos e versões como um todo:** todas as versões estão no mesmo diretório, mas só uma é mostrada por vez.

 **Histórico e rastreabilidade de alterações:** git blame.

 **Gerenciamento de versão:** commits e tags - disponível inclusive de forma gráfica.

 **Gerenciamento de conflito:** git merge - um pouco complexo, mas nenhum trabalho será sobreescrito.

 **Alterações não destrutivas, versões paralelas e experimentação:** branches.

 **Alterações atômicas e bem descritas:** commit.

 **Ambiente de trabalho individual:** não existe dependência com plataformas ou ferramentas específicas - seu projeto é um conjunto de arquivos no seu sistema operacional.

 **Alterações simultâneas em ambientes individuais:** suas alterações são estão na sua máquina e serão compartilhadas quando você querer.

 **Trabalho offline:** não é preciso se conectar a um serviço para fazer alterações locais.

O que é o Git?



git --everything-is-local



Search entire site...

Git is a **free and open source** distributed version control system designed to handle everything from small to very large projects with **speed and efficiency**.

Git is **easy to learn** and has a **tiny footprint with lightning fast performance**. It outclasses SCM tools like Subversion, CVS, Perforce, and ClearCase with features like **cheap local branching**, convenient **staging areas**, and **multiple workflows**.



Extaído de: <https://git-scm.com/>

Origens do Git

- Linus Torvalds:
 - Criador e principal desenvolvedor do Kernel do Linux.
- Intrinsicamente ligado com ***open-source***
- Como gerenciar mudanças no kernel feitas por dezenas de desenvolvedores?
 - *Patches* por e-mail em arquivos compactados;
 - BitKeeper;
 - Proprietário, mas gratuito para os desenvolvedores do Linux;
 - Desentendimento com a comunidade.
 - Git (~2005).



Linus Torvalds
torvalds

[Follow](#)

Linux Foundation

Portland, OR



git

Notícia do PCWorld sobre o início do git:

<https://www.pcworld.idg.com.au/article/129776/after-controversy-torvalds-begins-work-git/>

Breve história do git - livro “Pro Git”, de S. Chacon e B. Straub, p. 13): <https://git-scm.com/book/en/v2>

Mais sobre Linux / Kernels (fórum da comunidade brasileira): <https://www.vivaolinux.com.br/linux/>

Origens do GitHub

- Fundado em 2007;
- É um **host**, uma central de **repositórios Git**;
- Disponibiliza as funcionalidades do **Git** mais **funcionalidades próprias**;
- Recentemente adquirido pela **Microsoft**.



Mais informações em: <https://github.com/about/>

Git vs GitHub vs GUIs

- Possuem a filosofia do Unix/Linux
 - Usar o GitHub é: instalar o Git na sua máquina e dizer que vai armazenar o repositório no GitHub
 - O uso da linha de comando pode ser “maquiado” por uma gama de *GUIs* (*Graphical User Interfaces*)

“The Style.”

- (i) Make each program do one thing well. To do a new job, build afresh rather than complicate old programs by adding new “features.”
 - (ii) Expect the output of every program to become the input to another, as yet unknown, program. Don’t clutter output with extraneous information. Avoid stringently columnar or binary input formats. Don’t insist on interactive input.
 - (iii) Design and build software even operating systems to be
- (i) Faça cada programa fazer uma coisa bem. Para fazer um novo trabalho, construa algo novo ao invés de complicar programas antigos ao adicionar novas “funcionalidades”.
- (ii) Espere que a saída de todo programa se torne a entrada de outro programa ainda desconhecido. [...] Evite [...] formatos de entrada binários.

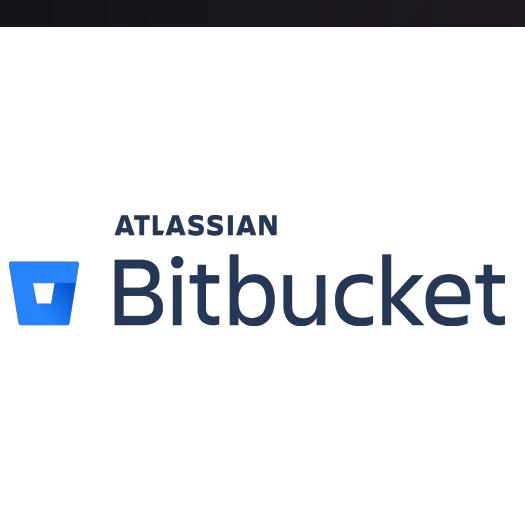
(McIlroy, M.D.; Pinson, E.N.; Tague, B.A., 1978, tradução nossa)

Trecho do artigo publicado no Bell System Technical Journal (1978) - artigo completo:

<https://archive.org/details/bstj57-6-1899/mode/2up>

Exemplos de hosts Git





Host Git **focado** em empresas.

Permite que as empresas tenham um “GitHub” em sua própria infraestrutura.

É vendido em conjunto com softwares de gerenciamento de projetos ágeis



Host Git para propósito geral / projetos **open-source** e empresas.

Possui ferramentas para empresas, mas o foco e a popularidade são menores.



Host Git **focado** em empresas.

Permite que as empresas tenham um “GitHub” em sua própria infraestrutura.

O que é um repositório?

- **Repositório:** um lugar onde se armazena uma coleção de coisas;
- **Repositório Git:** uma coleção de diferentes versões de um projeto gerenciadas pelo Git;
- **GitHub:** uma central de repositórios Git.
- O repositório central, chamado de *remote* fica nos servidores do GitHub;
- Os desenvolvedores constantemente copiam o *remote* para seus computadores, a cópia local chama-se *origin*.



Mais informações em:
<https://www.geeksforgeeks.org/what-is-a-git-repository/>

Onde está / como é um diretório git fisicamente?

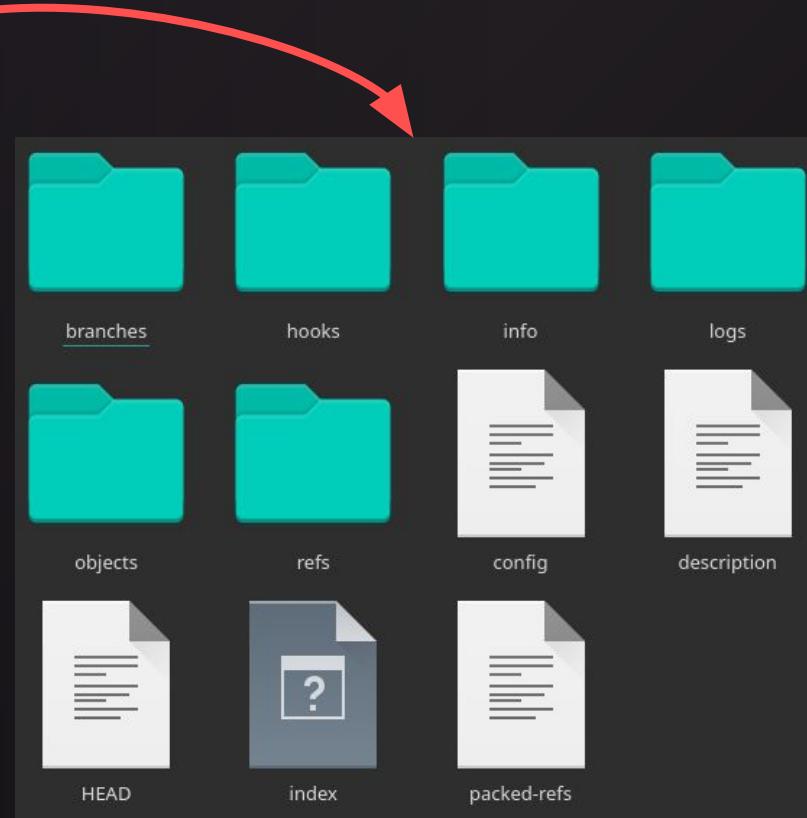
Arquivos de um projeto qualquer não gerenciado pelo Git:



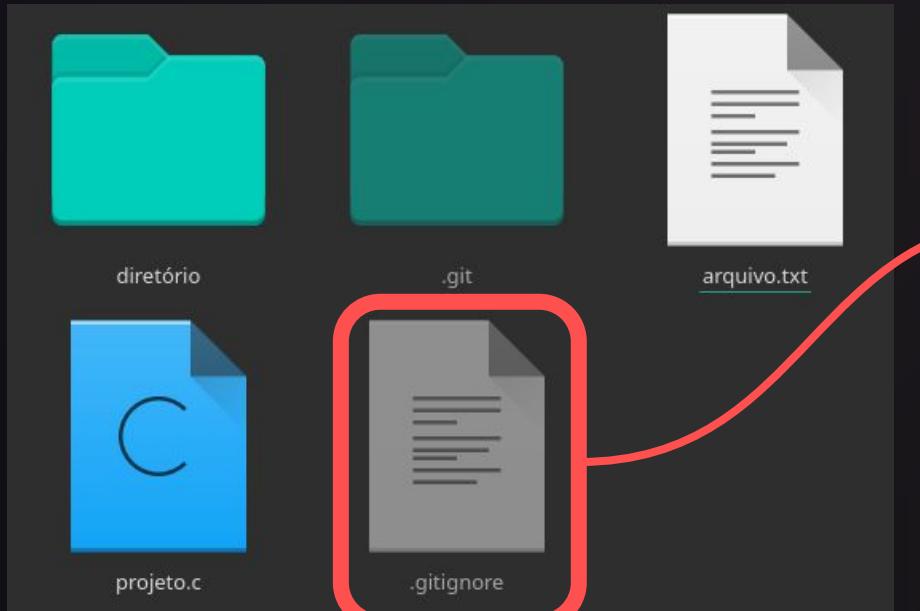
Um projeto rastreado pelo Git:

- Arquivos do projeto são os mesmos;
- Arquivos do repositório ficam em `./.git/` - um diretório **oculto**.





“Meu projeto tem arquivos com senhas, específicos ao meu sistema e gera arquivos de saída muito grandes, ou que mudam a cada execução - não quero, preciso ou posso compartilhar estes arquivos”



Ajuda para gerar .gitignore: <https://www.gitignore.io>

```
1  ### Code ###
2  .vscode/*
3  !.vscode/settings.json
4  !.vscode/tasks.json
5  !.vscode/launch.json
6  !.vscode/extensions.json
7
8  ### C ###
9
10 # Object files
11 *.o
12 *.ko
13 *.obj
14 *.elf
15
16 # Shared objects (inc. Windows DLLs)
17 *.dll
18 *.so
19 *.so.*
20 *.dylib
```


Principais
funcionalidades do
GitHub

Transporte de dados no git





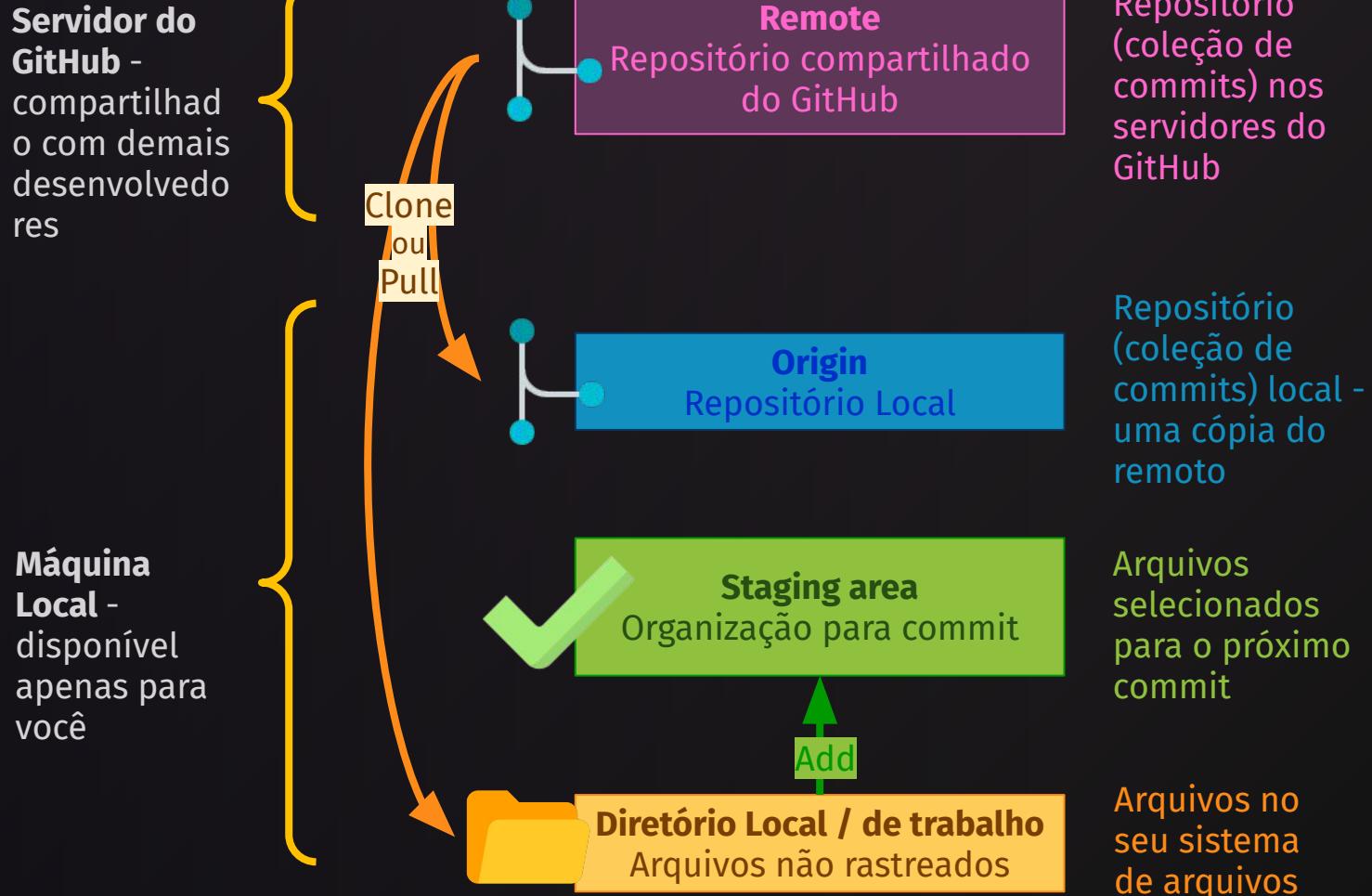
Servidor do GitHub -
compartilhado
com demais
desenvolvedores

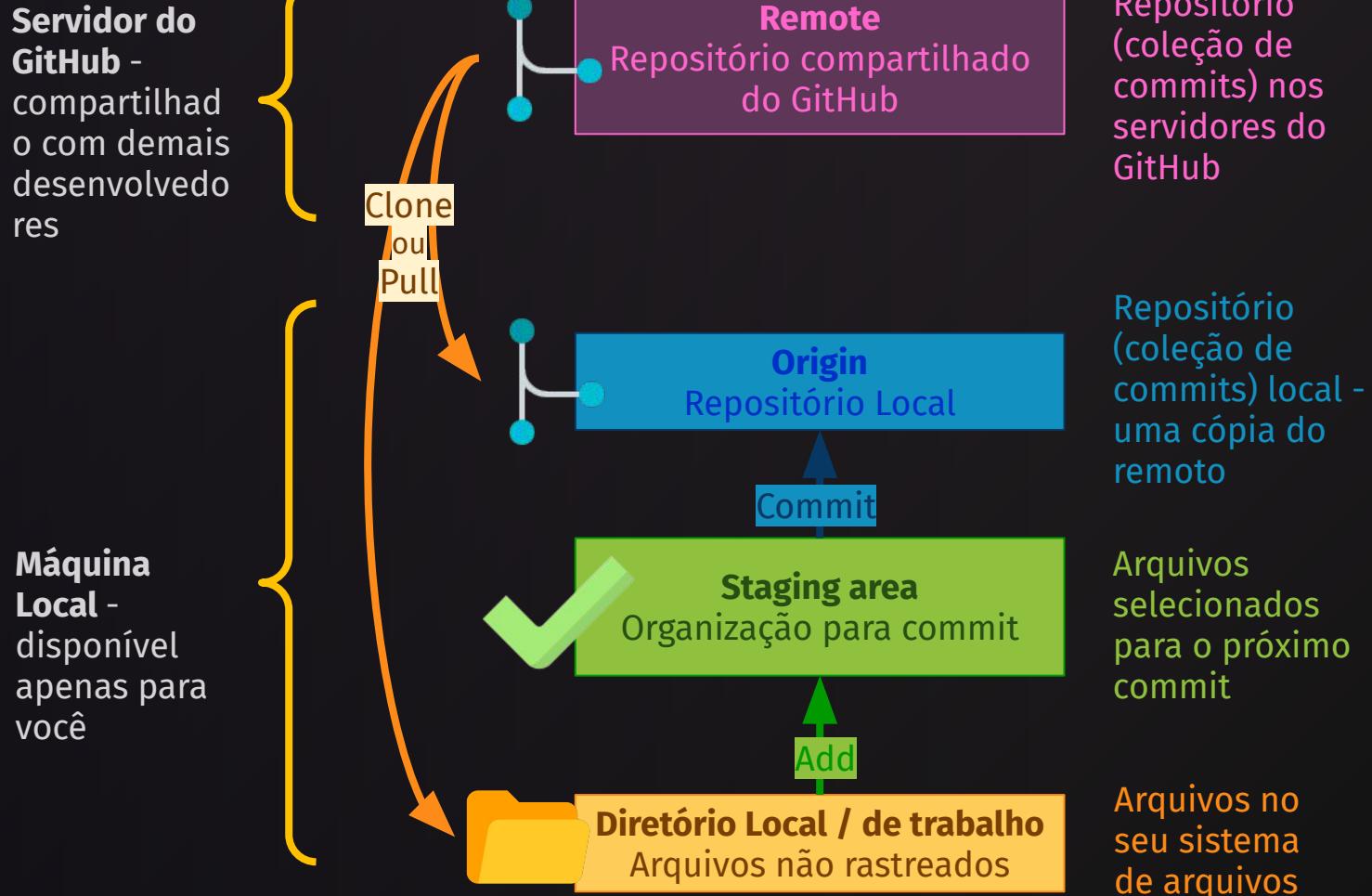


Máquina Local -
disponível
apenas para
você

O que é um *commit*?

- Uma “foto” do estado do projeto em algum momento do tempo.
- Um pacote de alterações armazenado em uma estrutura de dados gerenciada pelo git (o repositório).
 - A partir desta estrutura de dados é possível reconstruir o diretório de trabalho do projeto no momento de qualquer commit
- Geralmente os desenvolvedores os separam por funcionalidade - “*este commit implementa a funcionalidade x*”.
- Um repositório é uma coleção de commits e as versões dos arquivos para que seja possível reconstruir o diretório de trabalho de qualquer commit.







Servidor do GitHub - compartilhado com demais desenvolvedores

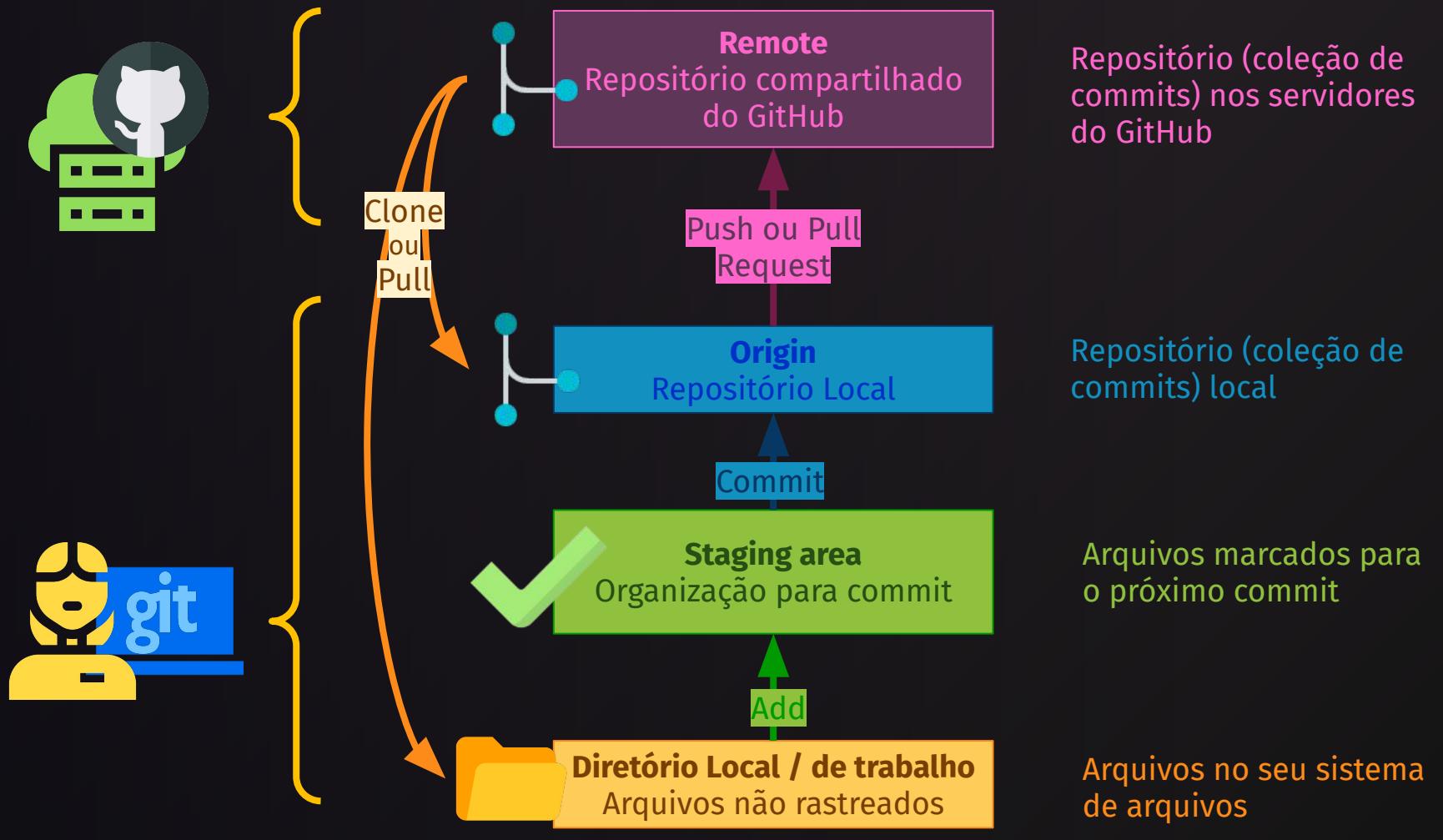


Repositório (coleção de commits) nos servidores do GitHub

Repositório (coleção de commits) local - uma cópia do remoto

Arquivos selecionados para o próximo commit

Arquivos no seu sistema de arquivos



The background of the slide is a dark, slightly grainy aerial photograph of a river system. The river flows from the top right towards the bottom left, creating a winding path through a landscape of green fields and clusters of trees. The water appears dark and reflects the surrounding environment.

Branches,
merge e
conflitos

Branches

- commits em uma branch qualquer
- commits na master

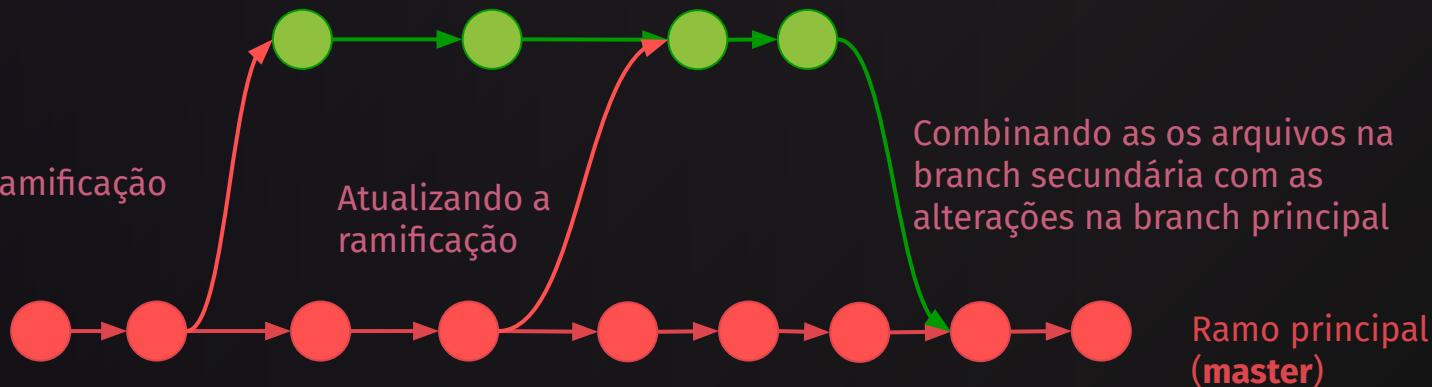
Ramificações nomeadas

Criando uma ramificação do projeto

Atualizando a ramificação

Exemplo de ramo secundário

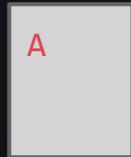
Combinando as os arquivos na branch secundária com as alterações na branch principal



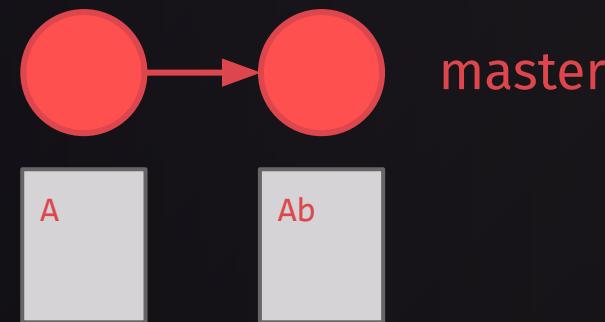
Tempo



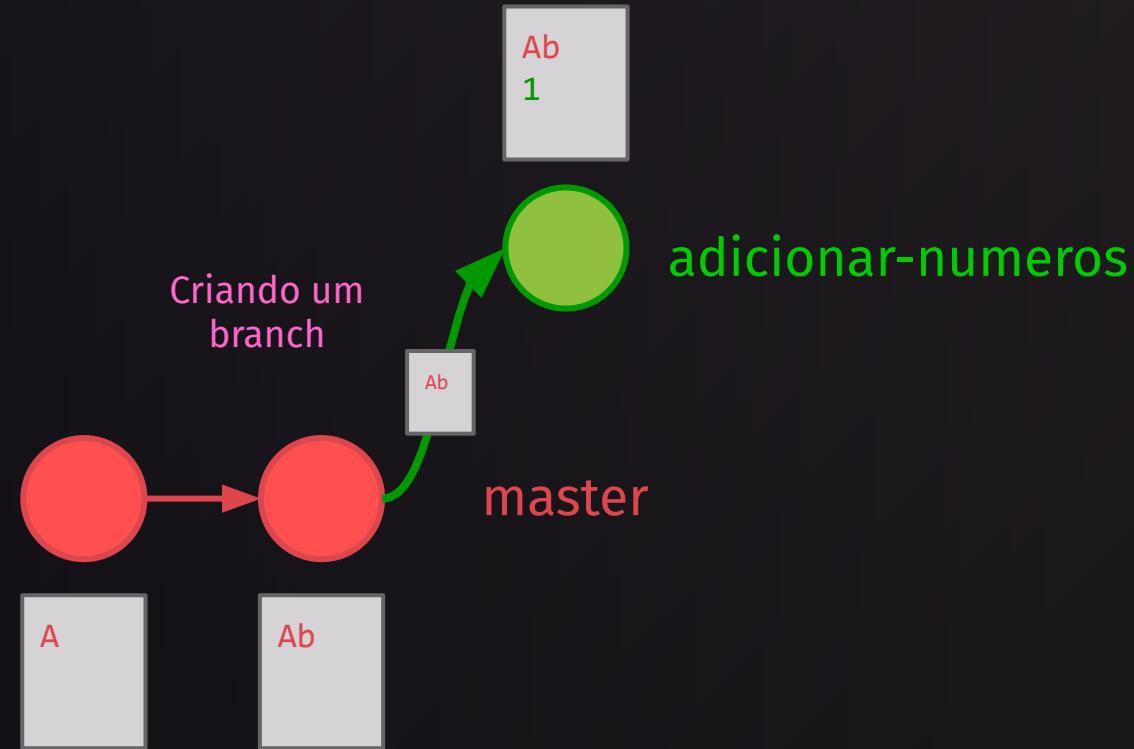
master



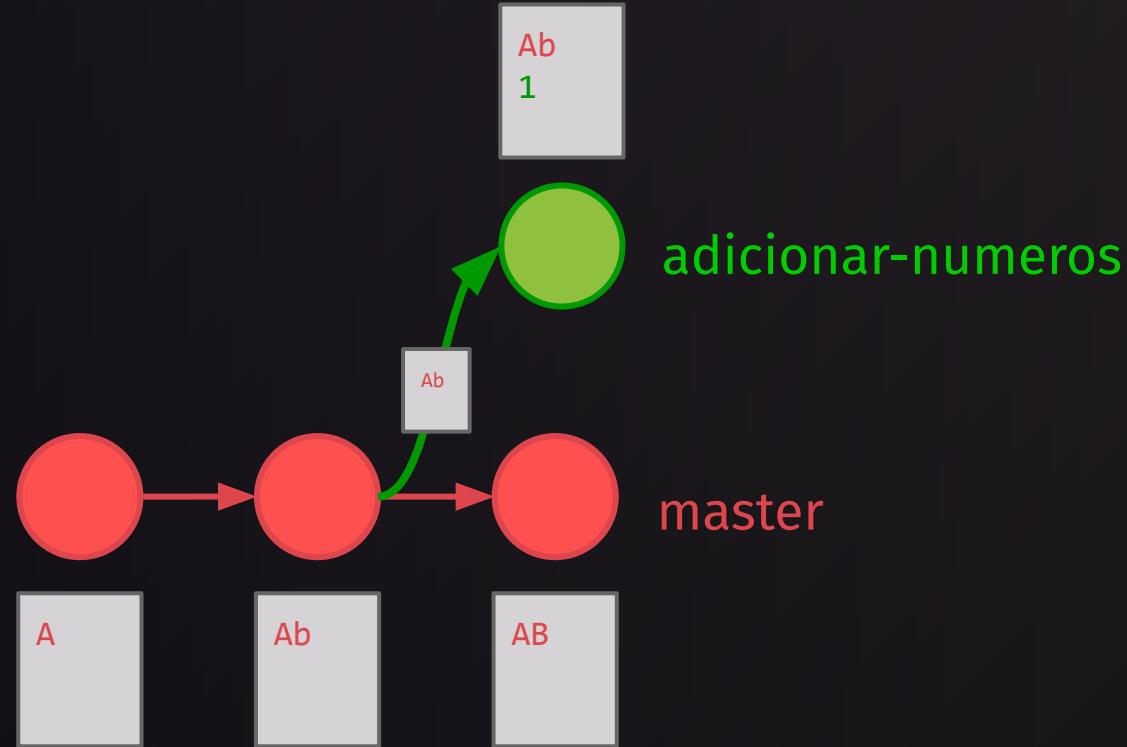
Tempo



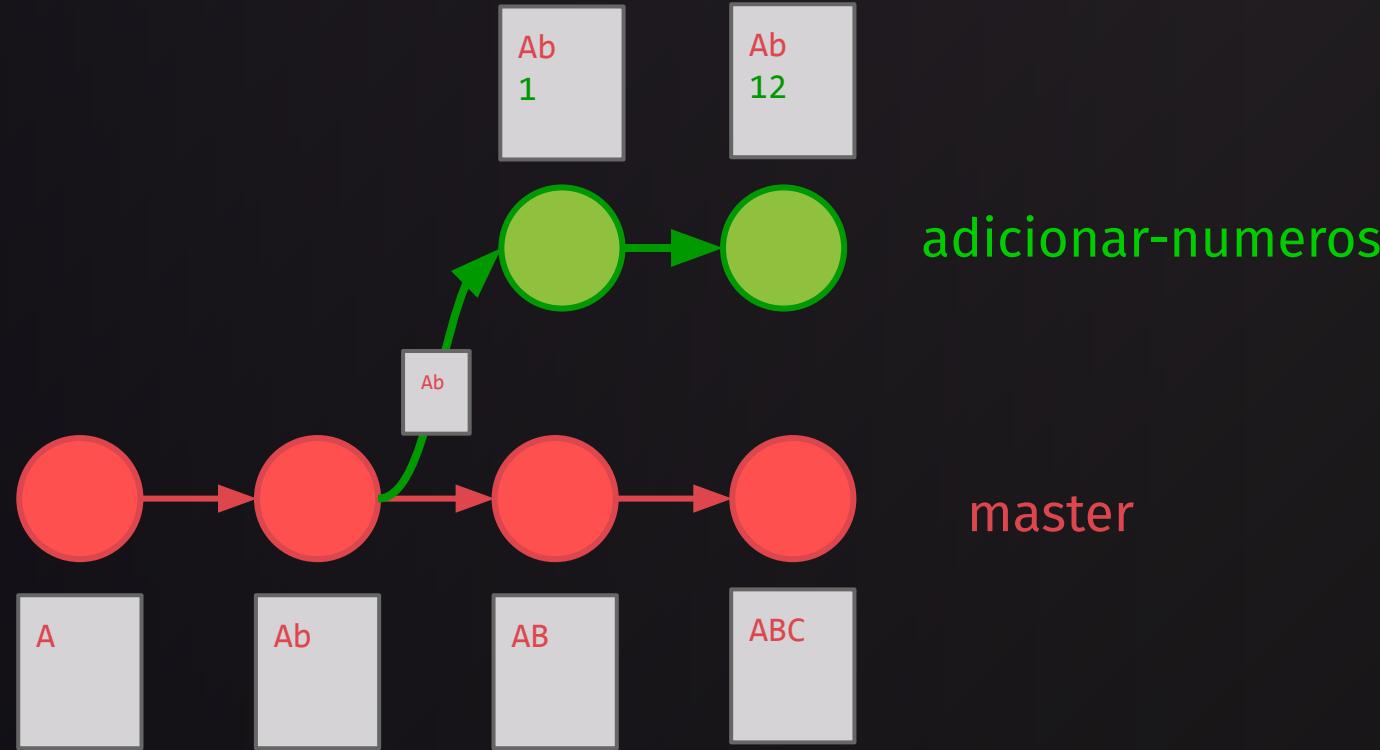
Tempo



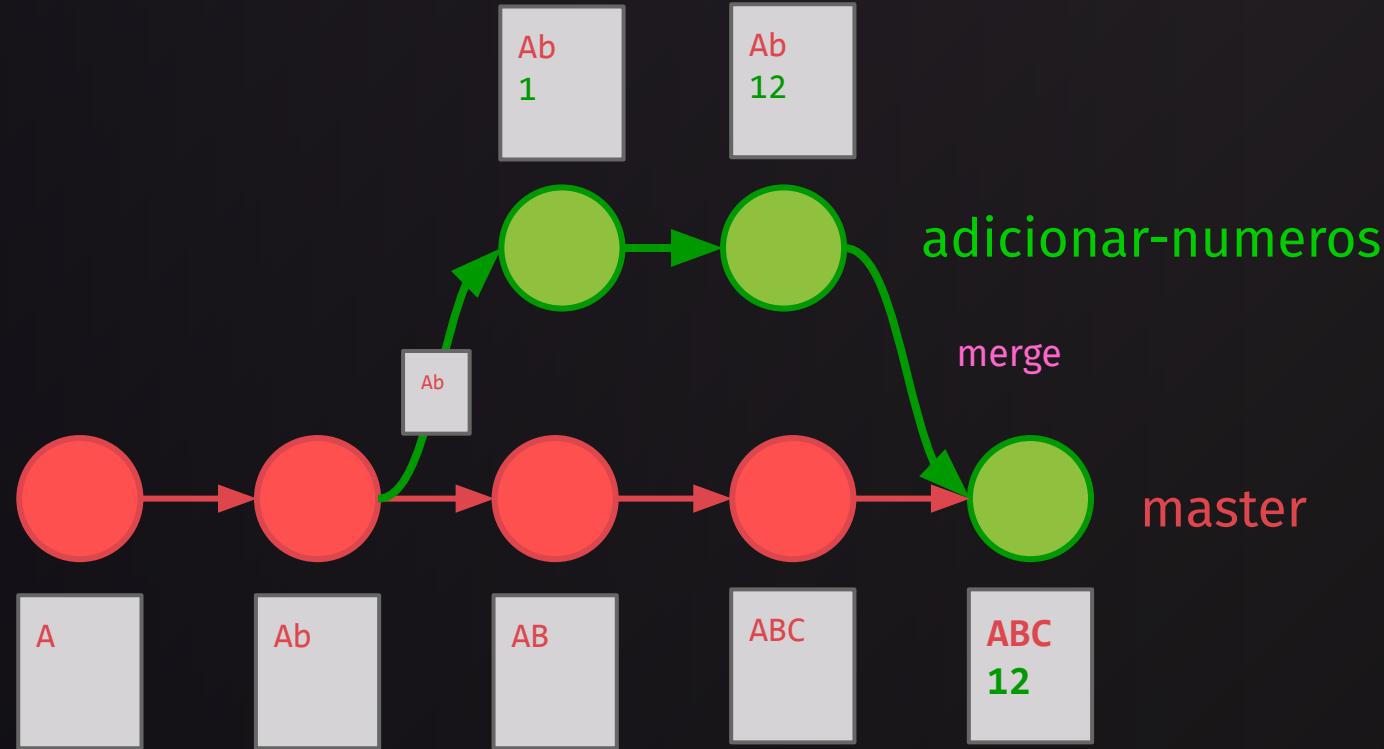
Tempo



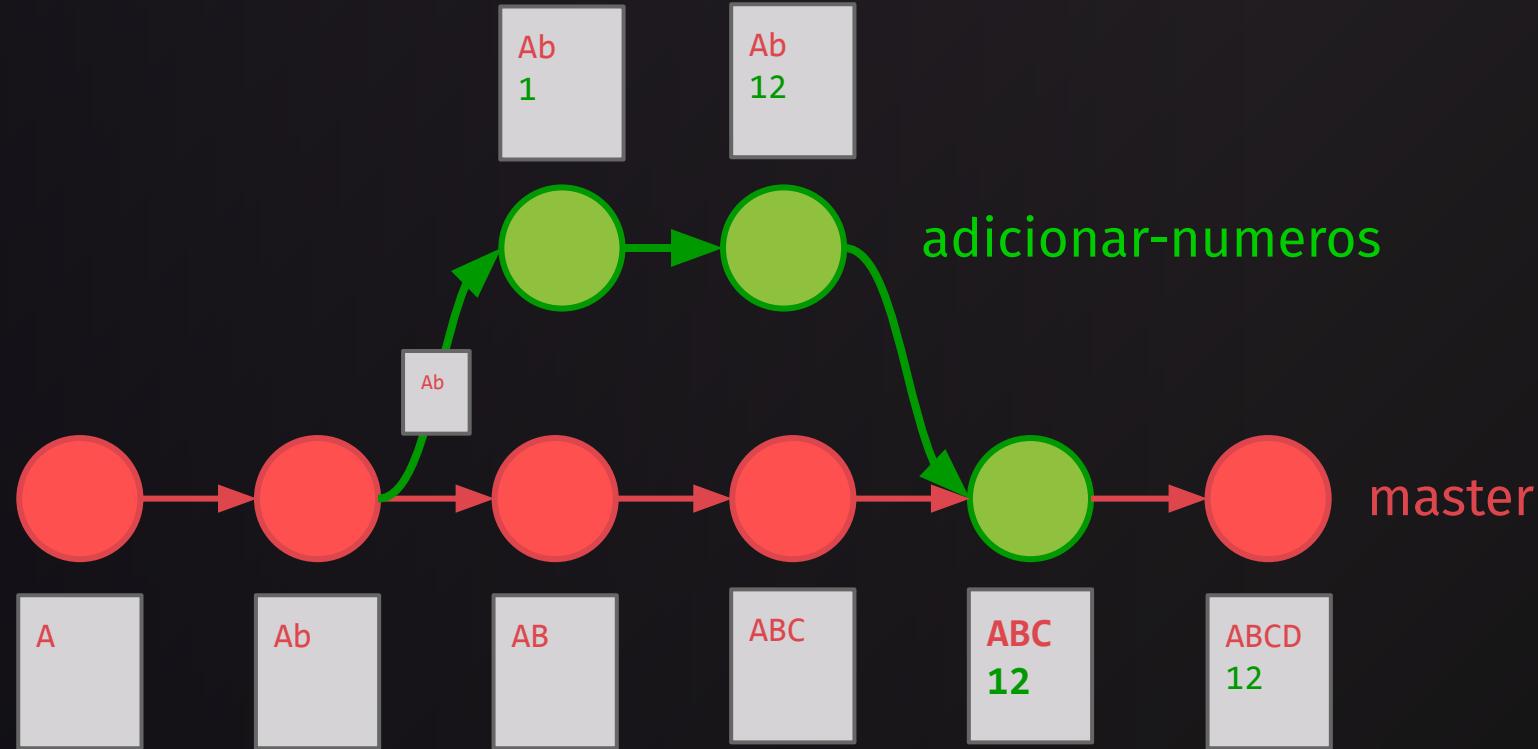
Tempo



Tempo

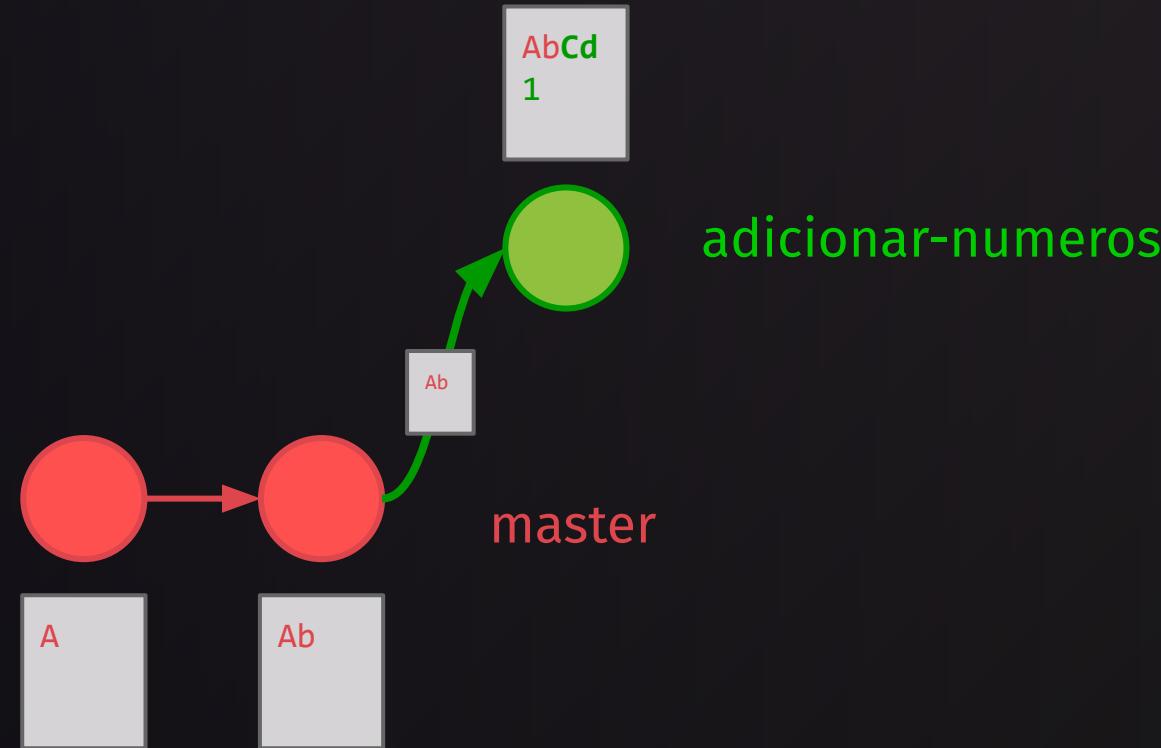


Tempo

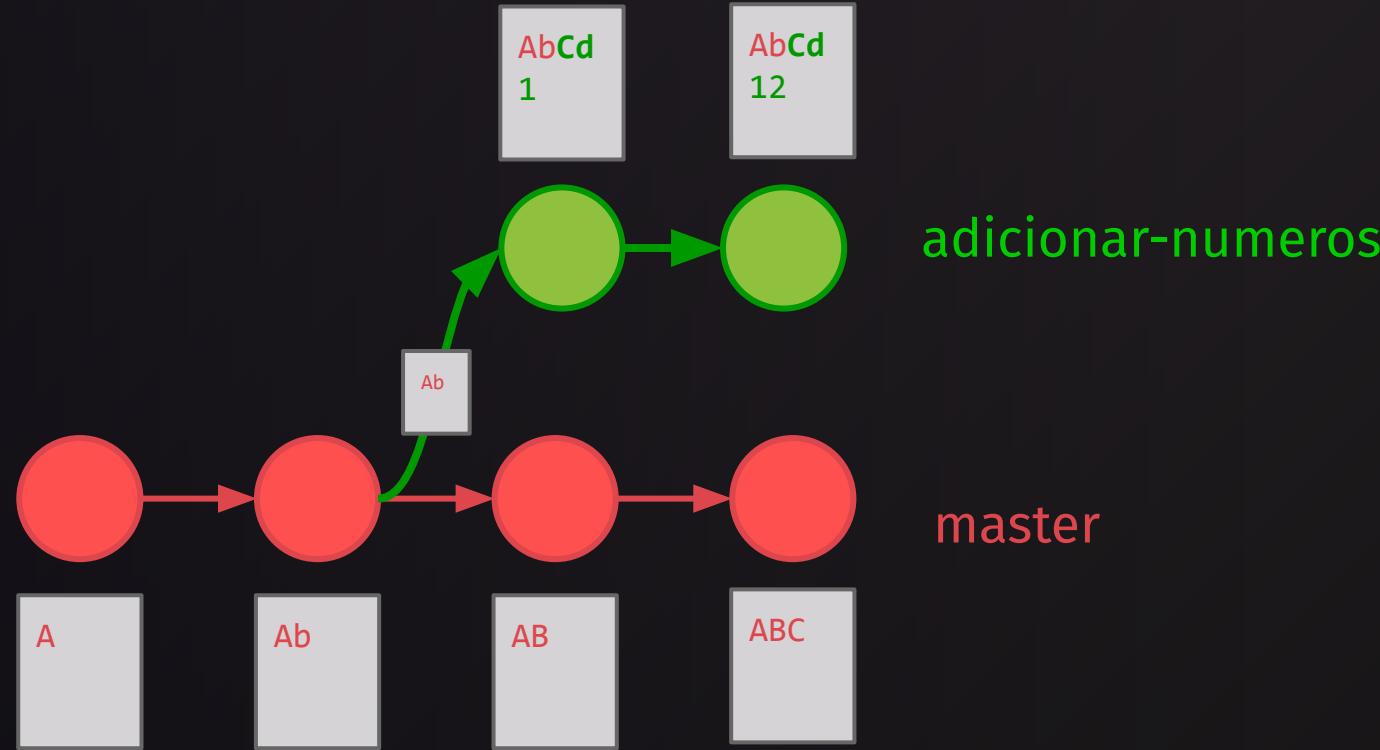


Conflitos

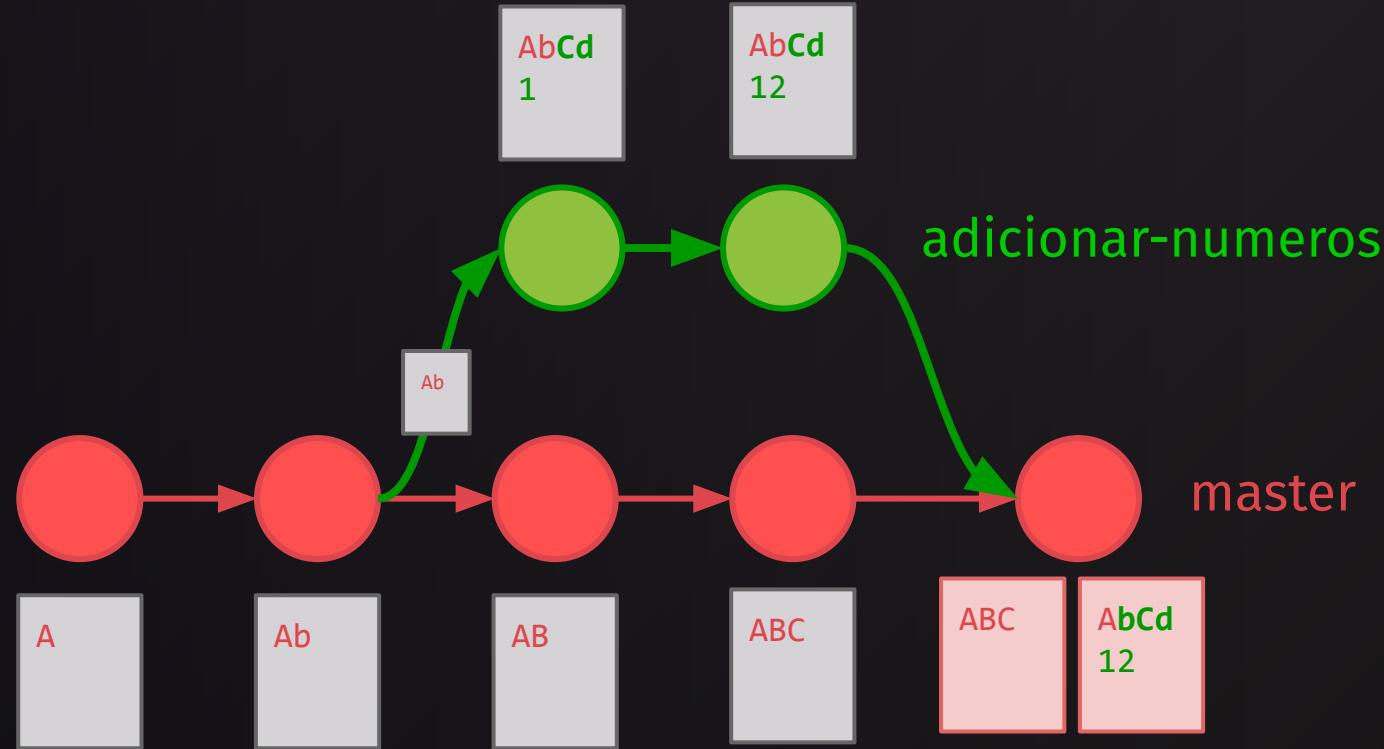
Tempo



Tempo



Tempo



master

ABC

adicionar-numeros

AbCd
12

Master afirma que é

Branch quer que seja

Não há versão mais recente na master - sem problemas

<<<<< HEAD
ABC
=====

AbCd
>>>>> adicionar-numeros

12

ABC
12

Accept Current Change | Accept Incoming Change | Accept Both Changes | Compare Changes |

<<<<< HEAD (Current Change)

ourselves

```
function add(a){  
  console.log(a)  
}
```

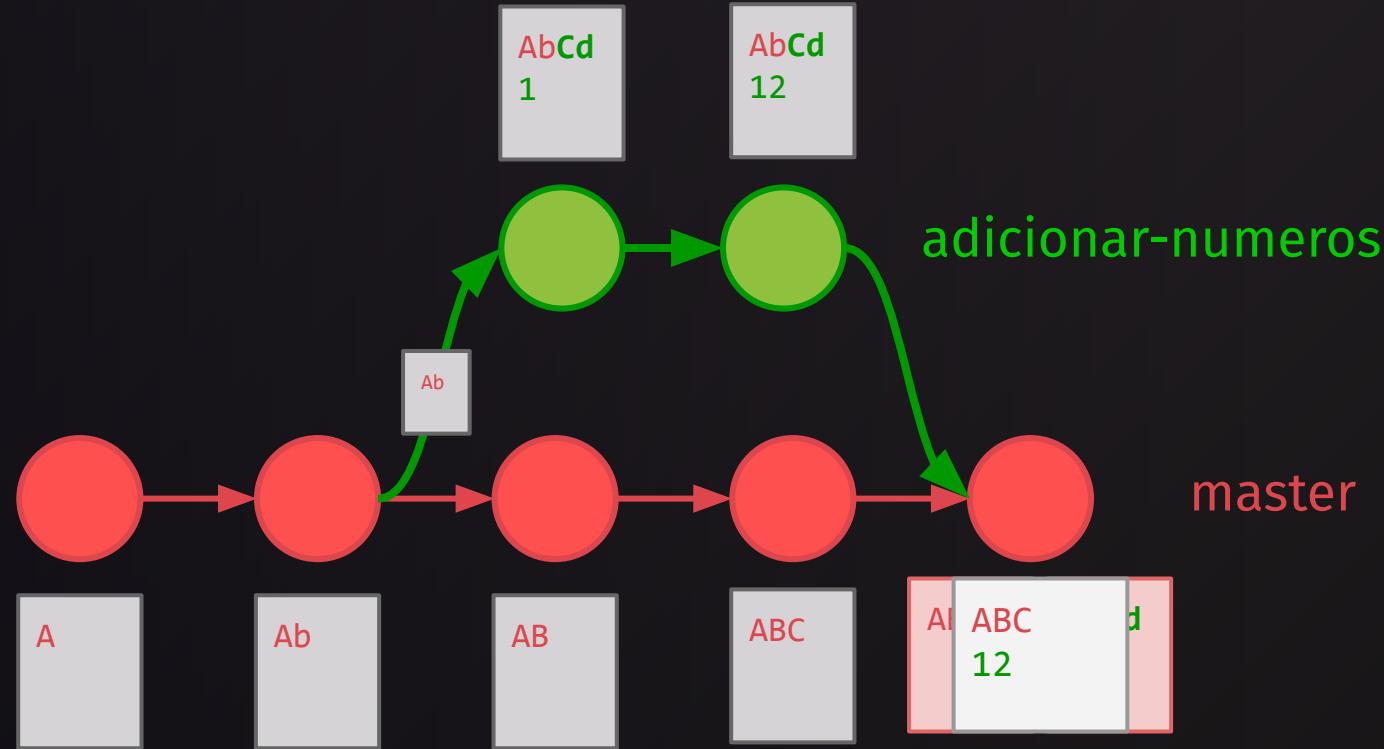
=====

themselves

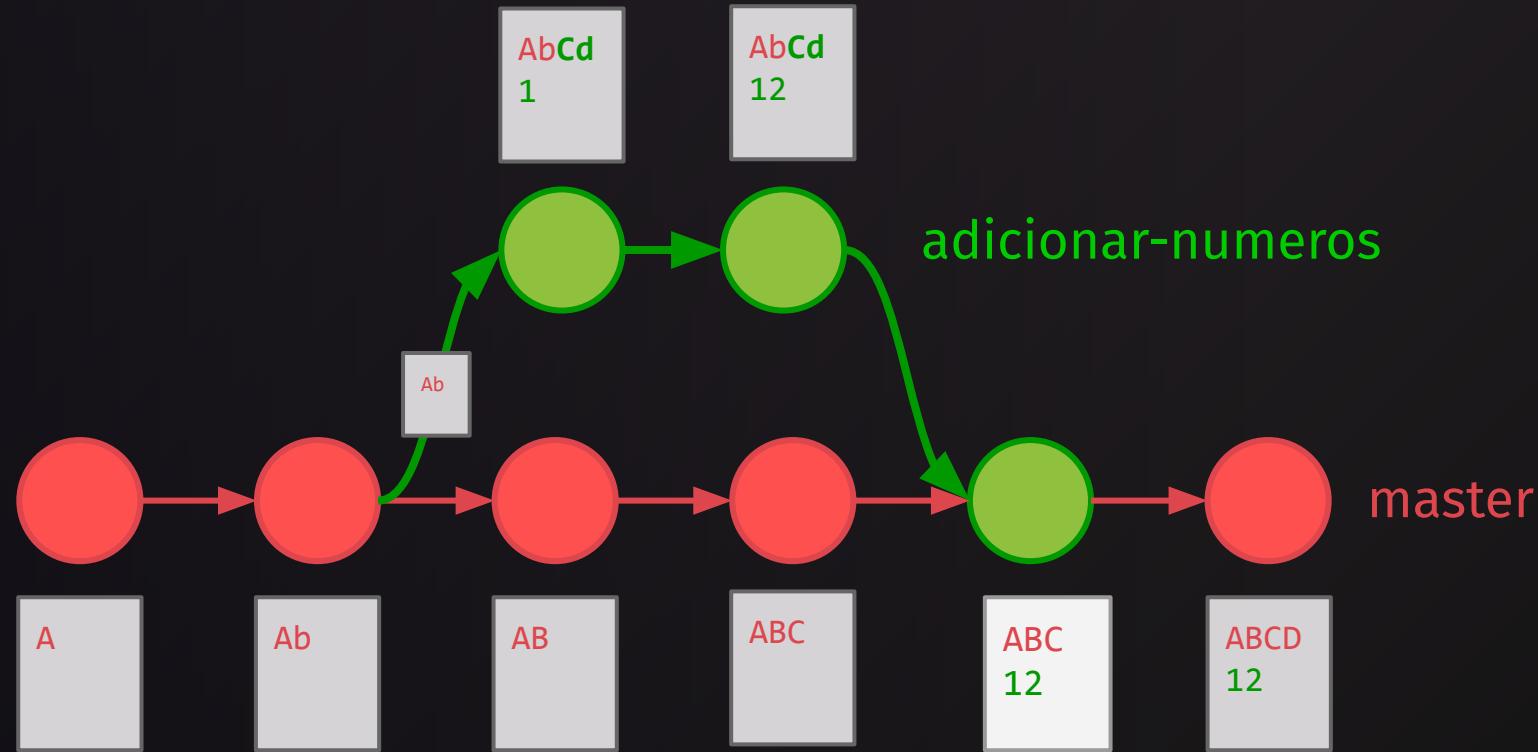
```
function add(a, b){  
  console.log(a, b)  
}
```

>>>>> deadbeef0123 (long): commit message (Incoming Change)

Tempo



Tempo

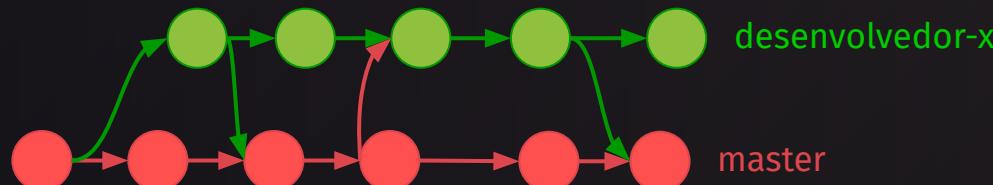


- ▼ organização
- ▼ complexidade
- ▲ problemas



master

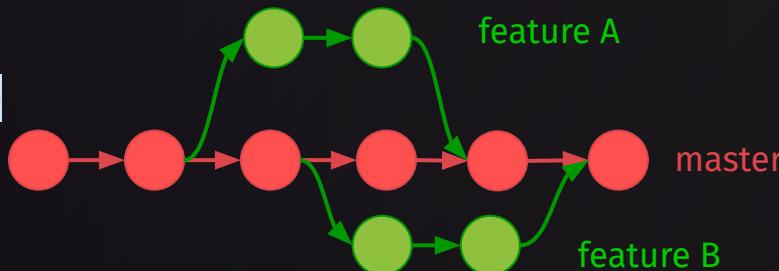
Tudo na **master**



desenvolvedor-x

master

Orientado por
desenvolvedor



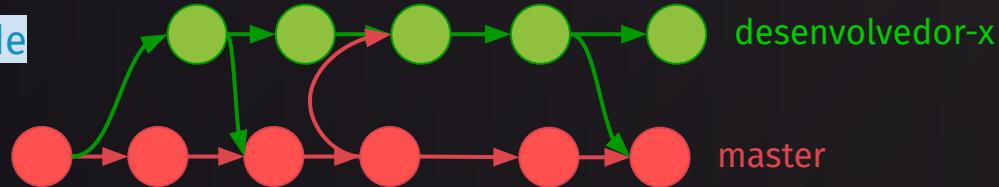
feature A

master

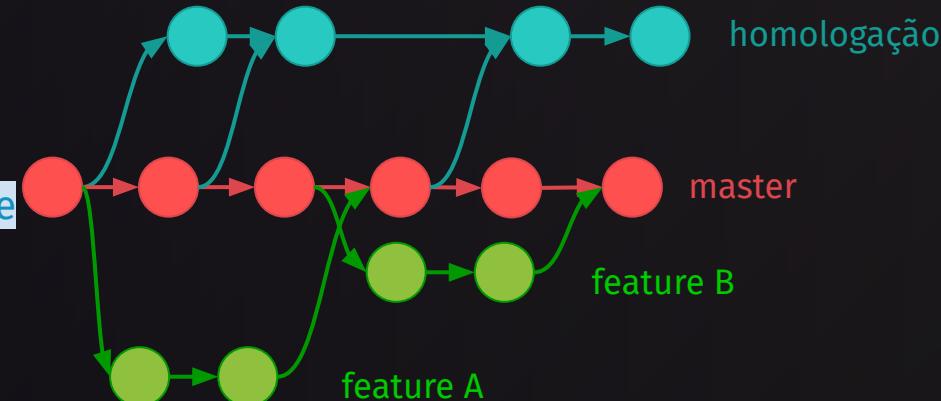
Orientado por
funcionalidade

- ▲ organização
- ▲ complexidade
- ▼ problemas





Orientado por
desenvolvedor



Orientado por
funcionalidade e
responsabilidade

- ▼ organização
- ▼ complexidade
- ▲ problemas

- ▲ organização
- ▲ complexidade
- ▼ problemas

*Pull-request,
fork e issues*

- ajudando
projetos da
comunidade

Pull request e fork

- Enviar seus commits para um projeto em que você não faz parte;
- Faça com que o projeto seja “seu”;
- Altere e...
 - guarde para você
 - envie suas alterações como um pedido de merge

forked from nasa/apod-api

Code Pull requests 0 Actions Projects 0 Wiki Security Insights Settings

Forking nasa/apod-api

It should only take a few seconds.

Refresh



Y seu-usuário / **apod-api**

forked from [nasa/apod-api](#)

Watch 0 Star 0 Fork 62

Pull requests 0 Actions Projects 0 Wiki Security Insights Settings

Astronomy Picture of the Day API service Edit

Manage topics

108 commits 3 branches 0 packages 0 releases 6 contributors Apache-2.0

Branch: master ▾ New pull request Create new file Upload files Find file Clone or download ▾

This branch is even with nasa:master. Pull request Compare

Author	Commit Message	Latest commit
brianthomas	remove mention of Google app engine	Latest commit d56a775 on Apr 4, 2019
apod	changed 'FileNotFoundException' to 'IOError'	11 months ago
tests	fix tests for http->https fix	3 years ago
.gitignore	Update gitignore to include pycharm project files and pycache files	3 years ago
.playground	Initial stub checkin of google flask-based python app code	4 years ago
LICENSE	Initial stub checkin of google flask-based python app code	4 years ago



Remote
Repositório
compartilhado do
GitHub





Remote
Repositório
compartilhado do
GitHub

Fork



Remote
Repositório
compartilhado do
GitHub



Remote
Repositório
compartilhado do
GitHub

Fork



Remote
Repositório
compartilhado do
GitHub

Alterações

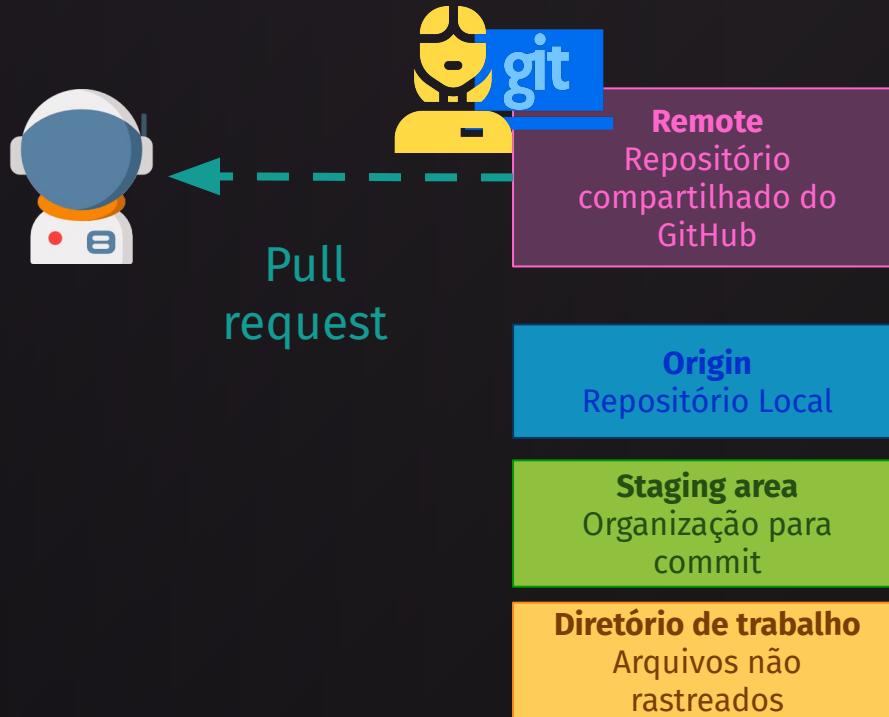
Origin
Repositório Local

Staging area
Organização para
commit

Diretório de trabalho
Arquivos não
rastreados



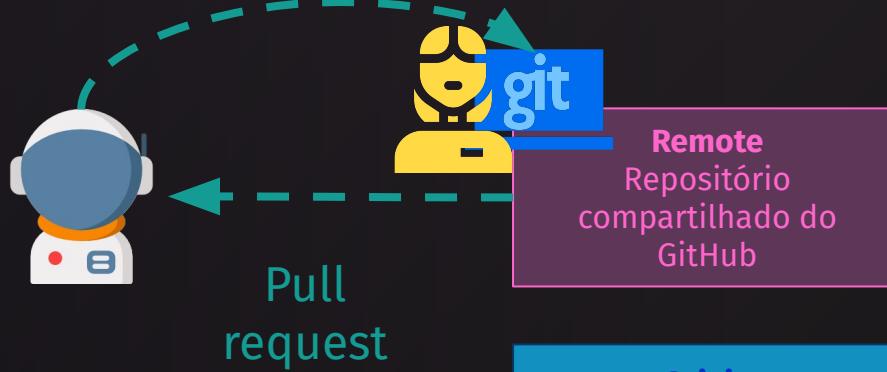
Remote
Repositório
compartilhado do
GitHub



Pedidos de alteração



Remote
Repositório
compartilhado do
GitHub

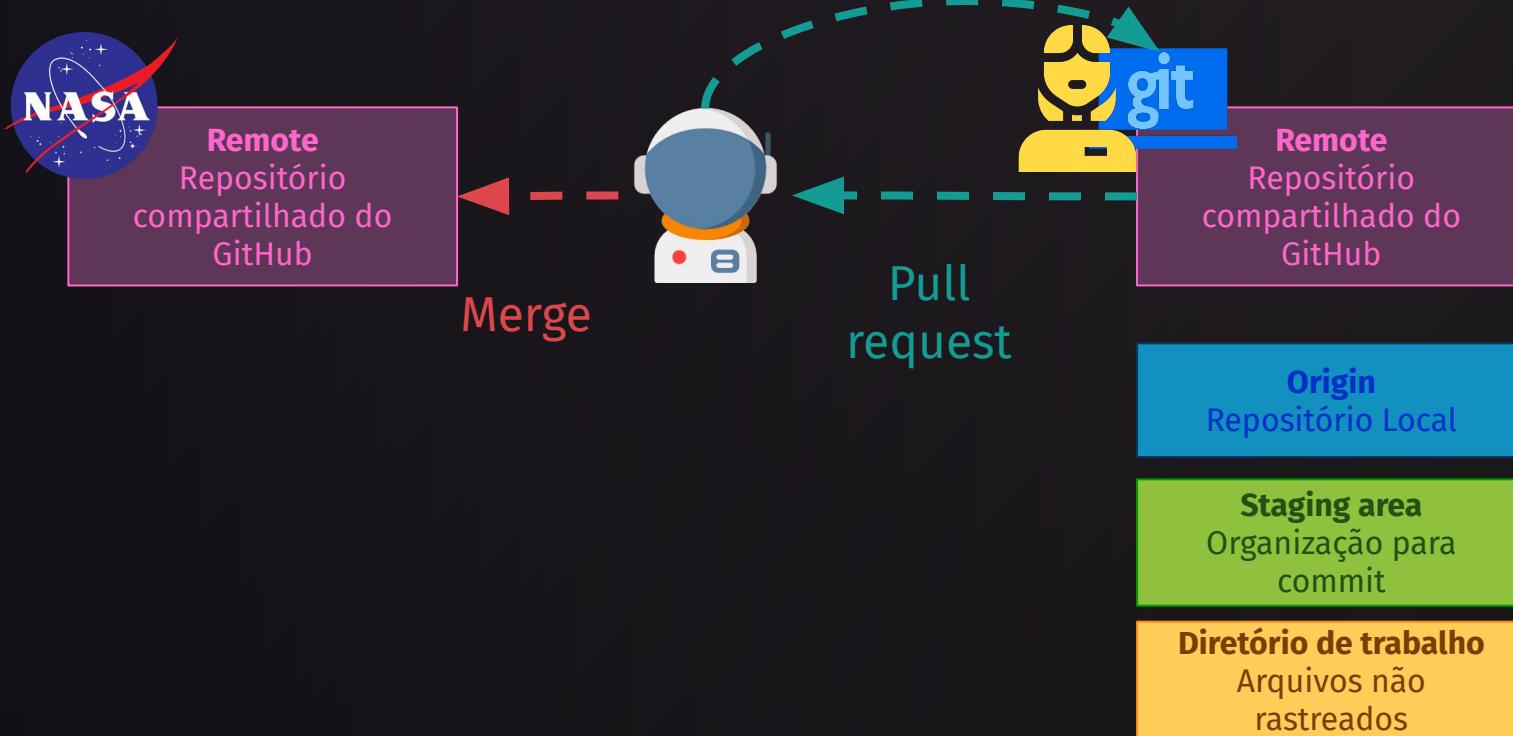


Origin
Repositório Local

Staging area
Organização para
commit

Diretório de trabalho
Arquivos não
rastreados

Pedidos de alteração



seu-usuário / apod-api

forked from nasa/apod-api

Watch 0

Star 0

Fork 62

Code

Pull requests 0

Actions

Projects 0

Wiki

Security

Insights

Settings

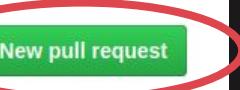
Filters ▾

is:open

Labels 9

Milestones

New pull request



Welcome to Pull Requests!

Pull requests help you collaborate on code with other people. As pull requests are created, they'll appear here in a searchable and filterable list. To get started, you should [create a pull request](#).

💡 ProTip! Type [g](#) [i](#) on any issue or pull request to go back to the issue listing page.

```
75  ````bash  
76  pip install -r requirements.txt  
77
```



Commit changes

Atualizando o README.md como um exemplo

Este commit faz parte de um exemplo onde algumas linhas foram alteradas, inseridas ou deletadas

⚙ Commit directly to the `master` branch.

🛡 Create a **new branch** for this commit and start a pull request. [Learn more about pull requests.](#)

`alreest-sec-patch-1`

[Propose file change](#)

[Cancel](#)

Open a pull request

The change you just made was written to a new branch named `alreest-sec-patch-1`. Create a pull request below to propose these changes.

base: `master` ▾ compare: `alreest-sec-patch-1` ▾ **X Can't automatically merge.** Don't worry, you can still create the pull request.

 Atualizando o README.md como um exemplo

[Write](#) [Preview](#) AA B i “ “ ↶ ↷ ≡ ☰ ↶ ↷ ↶ ↷

Este commit faz parte de um exemplo onde algumas linhas foram alteradas, inseridas ou deletadas

Attach files by dragging & dropping, selecting or pasting them.

Create pull request ▾

Remember, contributions to this repository should follow our [GitHub Community Guidelines](#).

Reviewers
No reviews

Assignees
No one—assign yourself

Labels
None yet

Projects
None yet

Milestone
No milestone

Linked issues
Use [Closing keywords](#) in the description to automatically close issues

Helpful resources
[GitHub Community Guidelines](#)

• **1 commit** ⌚ **1 file changed** ✍ **0 commit comments** 👤 **1 contributor**

15 README.md

... ... @@ -1,17 +1,8 @@

1 - # Astronomy Picture of the Day (APOD) microservice
1 + # Outro título
2 2
3 - just an example
3 + Algum texto
4 4
5 - A microservice written in Python with the [Flask micro framework](http://flask.pocoo.org).
6 -
7 - # Table of contents
8 - 1. [Getting Started](#getting_started)
9 - 1. [Standard environment](#standard_env)
10 - 2. [`virtualenv` environment](#virtualenv)
11 - 3. [`conda` environment](#conda)
12 - 2. [Docs](#docs)
13 - 3. [Feedback](#feedback)
14 - 4. [Author](#author)
15 5 + Deletei coisas
16 6
17 7
17 8 ## Getting started

No commit comments for this range

Como o Git pode me ajudar?

1. Maior organização nos trabalhos;
2. Ctrl + z “infinito”;
3. Maior facilidade na **divisão de responsabilidades**;
4. Histórico centralizado dos projetos em que você já fez
 - Se usou o README.md
 - Explicação de o que o projeto faz;
 - Instruções para rodar o programa;
 - Prints do programa funcionando.

- Documente esta divisão (ou a que utilizar) no GitHub;
- Usar mais arquivos diminui a chance de conflito e melhora a organização;
- **Leve a divisão ao nível do código:** separe o programa em funções, arquivos e módulos quando for preciso.

```

1  /** main.c ***/
2
3  #include <stdio.h>
4  #include "funcoes_adicionais.h"
5
6  int main(){
7      // ...
8      printQuadradoDe(3)
9      return 0;
10 }
11
12 float printQuadradoDe(float valor){
13     return print("%d² = %d", valor, quadradoDe(valor))
14 }
```

Módulo separado

<pre> 1 /** funcoes_adicionais.h ***/ 2 #ifndef FUNCOES_ADICIONAIS_ /* Include guard */ 3 #define FUNCOES_ADICIONAIS_ 4 5 int quadradoDe(int valor){ 6 return valor * valor 7 } 8 9 #endif //FUNCOES_ADICIONAIS_</pre>	<pre> 1 /** funcoes_adicionais.c ***/ 2 3 # include "funcoes_adicionais.h" 4 5 int quadradoDe(int valor){ 6 return valor * valor 7 }</pre>
---	---

Uma ideia para divisão de tarefas

(~Planning Poker + ~Pontos de Função)

1. Levante, divida e liste as tarefas a realizar
 - Funcionalidades ou funções em um programa, tópicos de uma pesquisa, etc.
2. Peça para cada pessoa do time dar uma nota de dificuldade de cada tarefa usando a escala de fibonacci - tente fazer com que todos revelem as notas ao mesmo tempo (evite o efeito manada)
 - 0, .5, 1, 2, 3, 5, 8, ...
3. Discuta os motivos das notas mais alta e mais baixa - encontre um consenso para a dificuldade da tarefa
4. Separe as tarefas de acordo com a vontade dos integrantes
5. Separe as restantes de forma que todos tenham uma soma de pontos o mais parecida possível

Possível forma de usar a escala

- 0 - não deveria nem ser uma tarefa (ex.: já está feito)
- 1/2 - muito fácil, sei de cabeça
- 1 - fácil, sei exatamente onde encontrar a solução
- 2 - fácil, mas preciso pesquisar algumas coisas
- 3 - não é difícil, mas vai levar tempo
- 5 - médio, sei resolver algumas coisas, outras não
- 8 - difícil, mas consigo resolver
- 13 - não faço ideia do que fazer
- 21 (ou mais) - procure subdividir a tarefa em tarefas menores

Planning poker e desenvolvimento ágil: [Scrum: Uma Metodologia Ágil para \[...\] de Projetos de Software](#)

Pontos de função: <http://www.gestao.mt.gov.br/images/files/Advogado teste22112012154044.pdf>

... pessoas reconhecem meu trabalho

Sou...

Trabalho
com...

Meu contato
é...

Participação
em
organizações

Daniel Shiffman's GitHub profile overview:

- Pinned Repositories:**
 - shiffman.net (shiffman.net personal site)
 - ml5js/ml5-library (Friendly machine learning for the web!)
 - A2Z-F18 (ITP A2Z Course for Fall 2018)
 - nature-of-code/noc-book (The Nature of Code book (archived repo, see README for new repo / build system!))
 - CodingTrain/website (Coding Train website)
 - Open-Source-Studio-at-ITP/Syllabus (Syllabus for ITP course about Open Source)
- Contributions Heatmap:** Shows 1,079 contributions in the last year, with a color scale from light green (Less) to dark green (More).

Linguagens que
conheço

Minha ajuda em
projetos da
comunidade

Minha atividade
ao longo do
tempo

Recomendação de podcast



Hipsters.tech 184

[Guia do Iniciante
em Github -
Hipsters #184](#)



Hipsters.tech 184

[Git e Github -
Hipsters #109](#)