

O QUE É
**SISTEMA DE
VERSIONAMENTO**



QUEM SOU EU ?



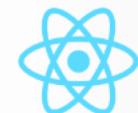
**Leonardo Brito
Bittencourt**
leobittencourt

Web Developer

20 anos de idade
Ribeirão Preto - SP

Atualmente sou Front-End no
Time de Inteligência Artificial do
Grupo SEB.

Trabalho atualmente com ReactJS,
ReactNative, NodeJS.



O QUE É CONTROLE DE VERSÃO ?

O controle de versão é um sistema que registra as mudanças feitas em um arquivo ou um conjunto de arquivos ao longo do tempo de forma que você possa recuperar versões específicas.

QUEM É GIT ?

É um sistema de controle de versões, usado principalmente no desenvolvimento de software.

O Git foi inicialmente projetado e desenvolvido por Linus Torvalds para o desenvolvimento do kernel Linux.

QUEM É GIT ?

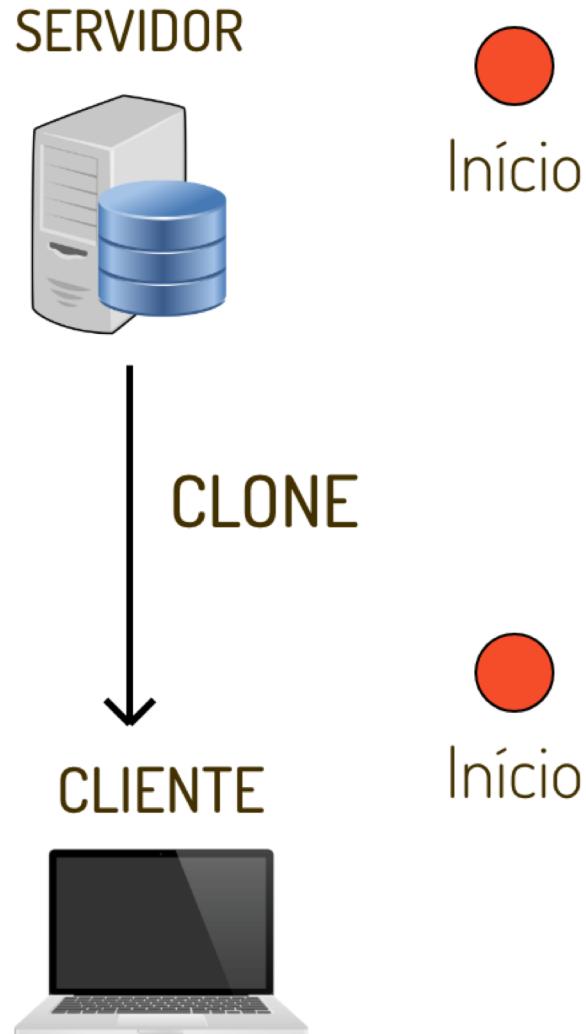
O Git é um software livre.

Sua manutenção é atualmente
supervisionada por Junio Hamano.



Engenheiro de Software e
Hacker Japonês

COMO O GIT FUNCIONA ?



COMO O GIT FUNCIONA ?

SERVIDOR

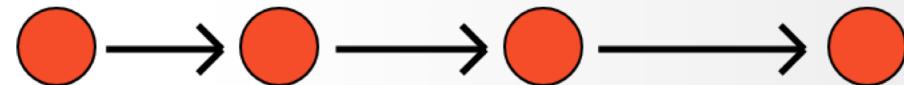


Início

CLONE



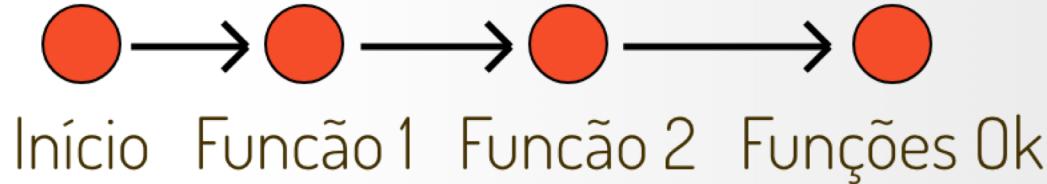
CLIENTE



Início Funcão 1 Funcão 2 Funcões 0k

COMO O GIT FUNCIONA ?

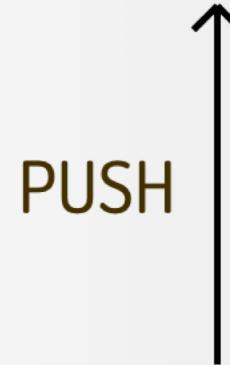
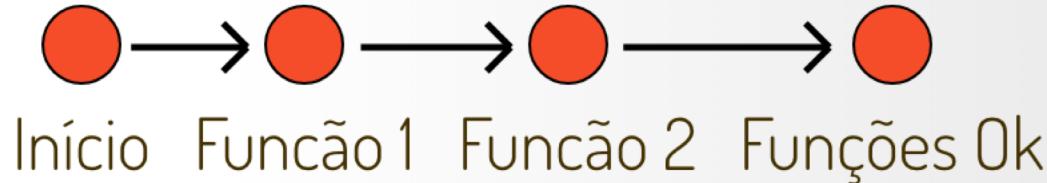
SERVIDOR



CLONE

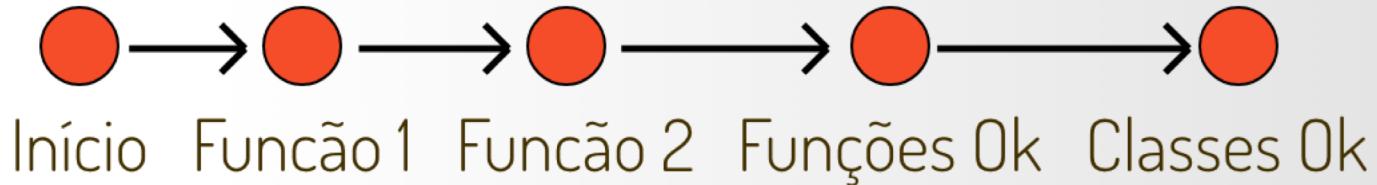


CLIENTE



COMO O GIT FUNCIONA ?

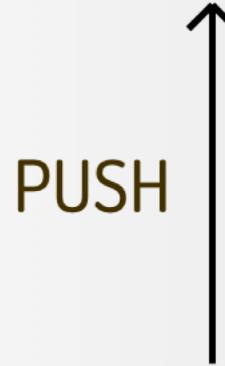
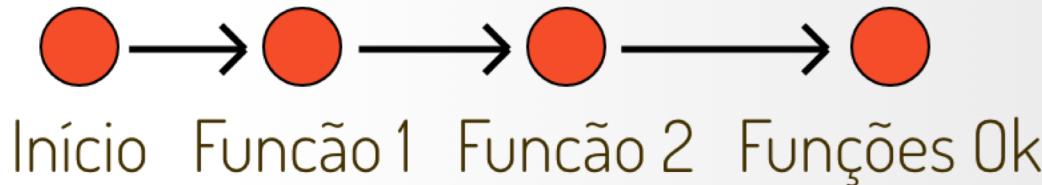
SERVIDOR



CLONE

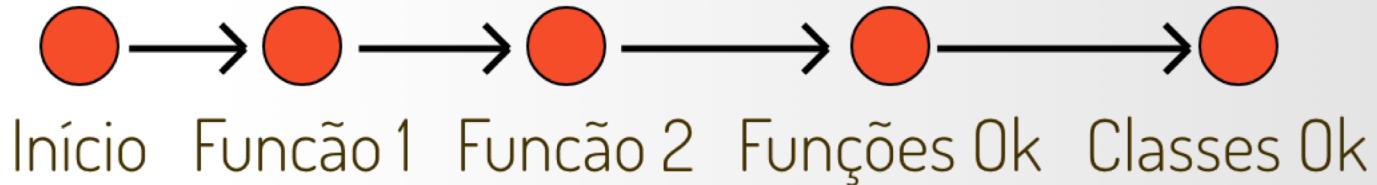


CLIENTE



COMO O GIT FUNCIONA ?

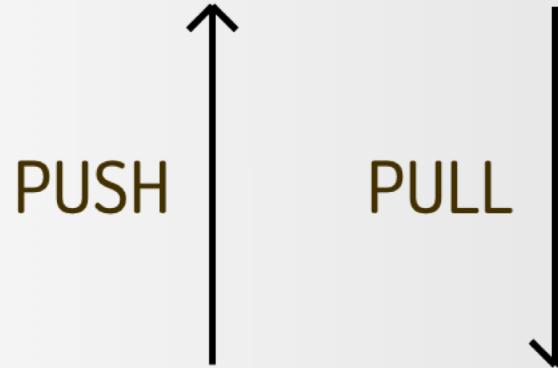
SERVIDOR



CLONE



CLIENTE



TIPOS DE ESTADOS DE UM ARQUIVO

UNTRACKED

NÃO RASTREADO

UNMODIFIED

INALTERADO

STAGED

SELECIONADO

MODIFIED

MODIFICADO

NÃO RASTREADO

```
t ➤ ~/Projects/PresentationGIT ➤ git init
Initialized empty Git repository in /Users/leonardo/Projects/PresentationGIT/.git/
t ➤ ~/Projects/PresentationGIT ➤ master ➤ ls
t ➤ ~/Projects/PresentationGIT ➤ master ➤ touch index.js
t ➤ ~/Projects/PresentationGIT ➤ master± ➤ git status
On branch master

No commits yet

Untracked files:
  (use "git add <file>..." to include in what will be committed)

    index.js

nothing added to commit but untracked files present (use "git add" to track)
t ➤ ~/Projects/PresentationGIT ➤ master± ➤
```

Novo arquivo que não foi rastreado pelo repositório

SELECIONADO

```
+ ~Projects/PresentationGIT ✘ master± git add index.js  
+ ~Projects/PresentationGIT ✘ master± git status  
On branch master  
  
No commits yet  
  
Changes to be committed:  
(use "git rm --cached <file>..." to unstage)  
  
    new file:   index.js  
  
+ ~Projects/PresentationGIT ✘ master±
```

Arquivo selecionado

COMMIT

```
t ➤ ~/Projects/PresentationGIT ➤ master ➤ git commit -m "arquivo index criado"
[master (root-commit) 0d17c0d] arquivo index criado
 1 file changed, 0 insertions(+), 0 deletions(-)
  create mode 100644 index.js
t ➤ ~/Projects/PresentationGIT ➤ master ➤ git status
On branch master
nothing to commit, working tree clean
t ➤ ~/Projects/PresentationGIT ➤ master ➤
```

Executado um commit para enviar o novo arquivo criado

ARQUIVO ALTERADO

```
t ~Projects/PresentationGIT > master nano index.js
t ~Projects/PresentationGIT > master± git status
On branch master
Changes not staged for commit:
  (use "git add <file>..." to update what will be committed)
  (use "git checkout -- <file>..." to discard changes in working directory)

    modified:   index.js

no changes added to commit (use "git add" and/or "git commit -a")
t ~Projects/PresentationGIT > master±
```

Arquivo alterado detectado

COMMIT DA ALTERAÇÃO

```
t ~Projects/PresentationGIT master± git add .
t ~Projects/PresentationGIT master± git status
On branch master
Changes to be committed:
  (use "git reset HEAD <file>..." to unstage)

    modified:   index.js

t ~Projects/PresentationGIT master± git commit -m "alteração feita no index"
[master 6066348] alteração feita no index
  1 file changed, 1 insertion(+)
t ~Projects/PresentationGIT master
```

Executado commit para enviar alteração do arquivo

CLONAR REPOSITORIO

```
git clone <repo>
```

Clona o repositório <repo> para a máquina local.

```
+ ➤ ~/Projects/PresentationGIT ➤ git clone https://github.com/leobittencourt/GerenciamentoDeVeiculos.git
Cloning into 'GerenciamentoDeVeiculos'...
remote: Enumerating objects: 62, done.
remote: Counting objects: 100% (62/62), done.
remote: Compressing objects: 100% (40/40), done.
remote: Total 62 (delta 21), reused 53 (delta 16), pack-reused 0
Unpacking objects: 100% (62/62), done.
+ ➤ ~/Projects/PresentationGIT ➤ ls
GerenciamentoDeVeiculos
+ ➤ ~/Projects/PresentationGIT ➤ cd GerenciamentoDeVeiculos
+ ➤ Projects/PresentationGIT/GerenciamentoDeVeiculos ➤ ↵ master ➤
```

INICIALIZAR REPOSITÓRIO

```
git init
```

Transforma o diretório atual em um repositório git, criando o subdiretório “.git”

```
t ➤ ~/Projects/PresentationGIT ➤ git init
Initialized empty Git repository in /Users/leonardo/Projects/PresentationGIT/.git/
t ➤ ~/Projects/PresentationGIT ➤ ↵ master ➤ l
total 0
drwxr-xr-x  3 leonardo  staff   96B Oct 15 23:06 .
drwxr-xr-x 17 leonardo  staff  544B Oct 15 23:05 ..
drwxr-xr-x 10 leonardo  staff  320B Oct 15 23:06 .git
t ➤ ~/Projects/PresentationGIT ➤ ↵ master ➤
```

SELECIONAR ARQUIVOS

```
git add .
```

Seleciona todos os arquivos que ainda não foram rastreados pelo repositório

```
+ ~/Projects/PresentationGIT ✘ master ➔ touch routes.js
+ ~/Projects/PresentationGIT ✘ master± ➔ git status
On branch master
Untracked files:
  (use "git add <file>..." to include in what will be committed)

    routes.js

nothing added to commit but untracked files present (use "git add" to track)
+ ~/Projects/PresentationGIT ✘ master± ➔ git add .
+ ~/Projects/PresentationGIT ✘ master± ➔ git status
On branch master
Changes to be committed:
  (use "git reset HEAD <file>..." to unstage)

    new file:   routes.js

+ ~/Projects/PresentationGIT ✘ master± ➔
```

COMMIT

```
git commit -m "outro commit"
```

Realiza o commit com a mensagem “outro commit”

```
t ➤ ~/Projects/PresentationGIT ➤ master± ➤ git status
On branch master
Changes to be committed:
  (use "git reset HEAD <file>..." to unstage)

    new file:   routes.js

t ➤ ~/Projects/PresentationGIT ➤ master± ➤ git commit -m "outro commit"
[master 5d782e0] outro commit
 1 file changed, 0 insertions(+), 0 deletions(-)
 create mode 100644 routes.js
t ➤ ~/Projects/PresentationGIT ➤ master ➤
```

REPOSITÓRIO GITHUB

leobittencourt / Presentation-GIT

Watch 0 Star 0 Fork 0

Code Issues 0 Pull requests 0 Projects 0 Wiki Insights Settings

Presentation-GIT Edit

Manage topics

1 commit 1 branch 0 releases 1 contributor

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

leobittencourt Initial commit Latest commit 1f6d1ad just now

LICENSE Initial commit just now

README.md Initial commit just now

README.md

Presentation-GIT

Presentation-GIT

“LINKAR” REPOSITÓRIO LOCAL COM REMOTO

```
git remote add origin <repo>
```

Adiciona um repositório remoto como origin no meu repositório local.

```
t ➤ ~/Projects/PresentationGIT ✘ master ➤ git remote add origin https://github.com/leobittencourt/Presentation-GIT.git
t ➤ ~/Projects/PresentationGIT ✘ master ➤ git remote -v
origin  https://github.com/leobittencourt/Presentation-GIT.git (fetch)
origin  https://github.com/leobittencourt/Presentation-GIT.git (push)
t ➤ ~/Projects/PresentationGIT ✘ master
```

BAIXAR ATUALIZAÇÕES

```
git pull origin <branch>
```

Baixar atualizações do repositório remoto

```
x + ➤ ~/Projects/PresentationGIT ➤ ↵ master ➤ git pull origin master
From https://github.com/leobittencourt/Presentation-GIT
 * branch           master      -> FETCH_HEAD
fatal: refusing to merge unrelated histories
x + ➤ ~/Projects/PresentationGIT ➤ ↵ master ➤
```

LISTAR BRANCHS

```
git branch [-a]
```

Lista todas as branch que você possui em sua máquina
[-a] Lista todas as branch disponíveis no repositório

```
x + ~/Projects/PresentationGIT ✘ master ➔ git branch
* master
+ ~/Projects/PresentationGIT ✘ master ➔ git branch -a
* master
  remotes/origin/master
+ ~/Projects/PresentationGIT ✘ master ➔
```

“SUBIR” ATUALIZAÇÕES

```
git push origin <branch>
```

Envia as atualizações para o seu repositório remoto

```
+ ➤ ~/Projects/Presentation-GIT ➤ master ➤ touch index.js
+ ➤ ~/Projects/Presentation-GIT ➤ master± ➤ git add .
+ ➤ ~/Projects/Presentation-GIT ➤ master± ➤ git commit -m "add index.js"
[master 96c5fa3] add index.js
 1 file changed, 0 insertions(+), 0 deletions(-)
 create mode 100644 index.js
+ ➤ ~/Projects/Presentation-GIT ➤ master ➤ git push origin master
Counting objects: 3, done.
Delta compression using up to 4 threads.
Compressing objects: 100% (2/2), done.
Writing objects: 100% (3/3), 324 bytes | 324.00 KiB/s, done.
Total 3 (delta 0), reused 0 (delta 0)
To https://github.com/leobittencourt/Presentation-GIT.git
 1f6d1ad..96c5fa3  master -> master
+ ➤ ~/Projects/Presentation-GIT ➤ master ➤
```

CRIAR NOVO BRANCH

```
git checkout -b <nome>
```

Criar um novo branch a partir do branch atual realizando uma cópia idêntica preservando o branch anterior

```
t ~ /Projects/Presentation-GIT ↵ master git branch
* master
t ~ /Projects/Presentation-GIT ↵ master git checkout -b funcao-soma
Switched to a new branch 'funcao-soma'
t ~ /Projects/Presentation-GIT ↵ funcao-soma ls
LICENSE README.md index.js
t ~ /Projects/Presentation-GIT ↵ funcao-soma
```

MUDAR DE BRANCH

```
git checkout master
```

Mudar da branch funcho-soma para o branch master,
com isso todos os respectivos arquivos são carregados

```
+ ➤ ~/Projects/Presentation-GIT ➤ ↵ funcao-soma ➤ git checkout master
Switched to branch 'master'
Your branch is up to date with 'origin/master'.
+ ➤ ~/Projects/Presentation-GIT ➤ ↵ master ➤
```

APAGAR BRANCH

```
git branch -d [-D] <nome>
```

Apaga o branch especificado que já foi mesclado

[-D] Força remover o branch mesmo que não tenha sido
mesclado

```
t ➤ ~/Projects/Presentation-GIT ➤ ↵ master ➤ git branch
  funcao-soma
* master
t ➤ ~/Projects/Presentation-GIT ➤ ↵ master ➤ git branch -d funcao-soma
Deleted branch funcao-soma (was 96c5fa3).
t ➤ ~/Projects/Presentation-GIT ➤ ↵ master ➤ git branch
* master
t ➤ ~/Projects/Presentation-GIT ➤ ↵ master ➤
```

VERIFICAR DIFERENÇAS ENTRE BRANCH

```
git diff <branch1> <branch2>
```

Verificar diferenças entre brancos

```
diff --git a/index.js b/index.js
index e69de29..e121465 100644
--- a/index.js
+++ b/index.js
@@ -0,0 +1 @@
+const soma = (a, b) => a +b;
(END)
```

MESCLANDO BRANCH

```
git merge <branch>
```

Realizará a mesclagem do branch especificado para o branch atual

[–no-ff] Obrigatoriamente adiciona uma mensagem de mesclagem para melhorar visualização no log

```
t ~/Projects/Presentation-GIT ✚ master ➔ git merge --no-ff funcao-soma
```

MESCLANDO BRANCH

Mensagem para especificar mesclagem

```
Merge branch 'funcao-soma'
```

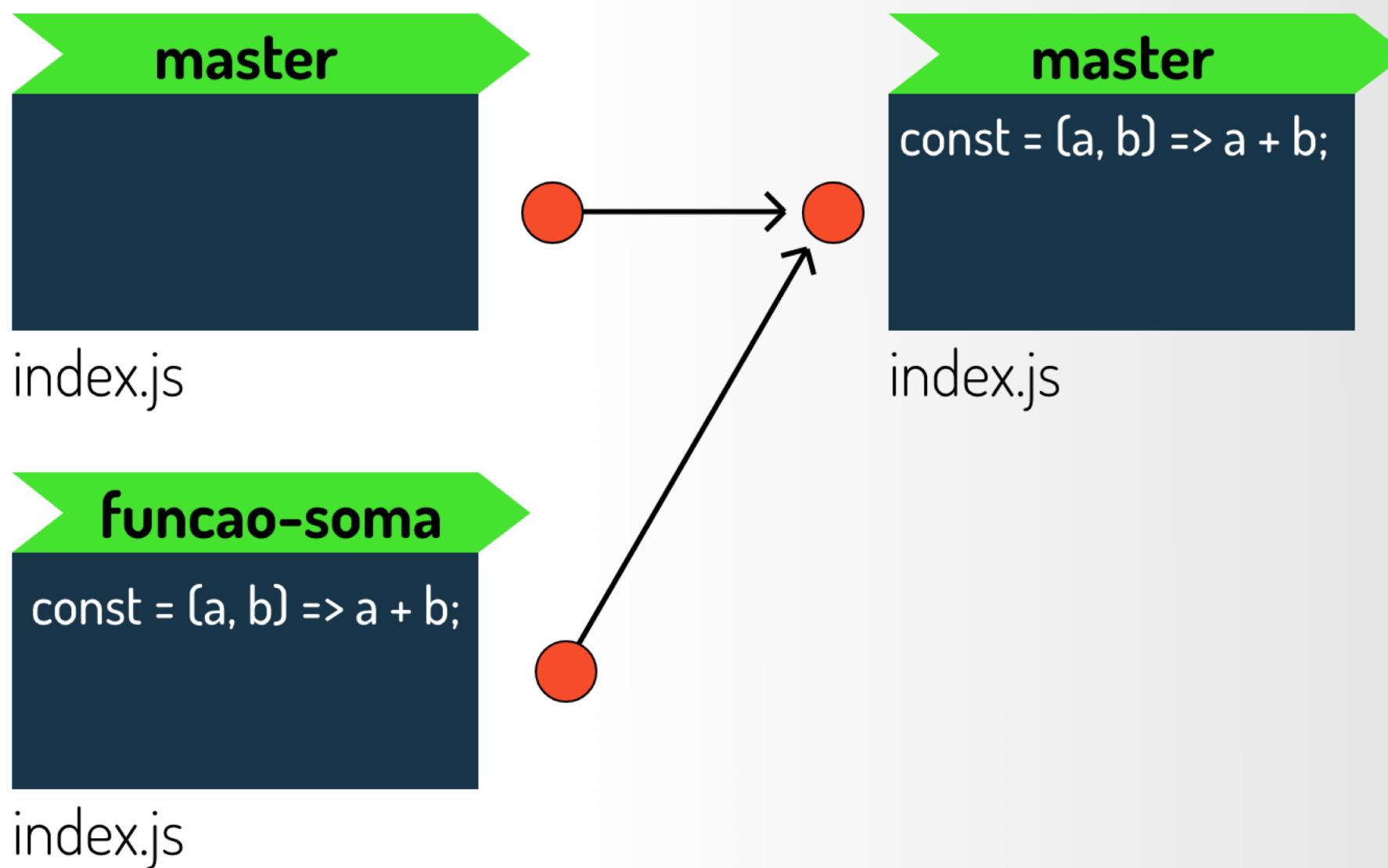
```
# Please enter a commit message to explain why this merge is necessary,  
# especially if it merges an updated upstream into a topic branch.  
#  
# Lines starting with '#' will be ignored, and an empty message aborts  
# the commit.  
~  
~
```

MESCLANDO BRANCH

Verificando se a mesclagem foi realizada com sucesso

```
t ➤ ~/Projects/Presentation-GIT ➤ master ➤ git merge --no-ff funcao-soma
Merge made by the 'recursive' strategy.
index.js | 1 +
1 file changed, 1 insertion(+)
t ➤ ~/Projects/Presentation-GIT ➤ master ➤ cat index.js
const soma = (a, b) => a +b;
t ➤ ~/Projects/Presentation-GIT ➤ master ➤
```

MESCLANDO BRANCH



MESCLANDO BRANCH

funcao-sub

```
const func = (a, b) => a - b;
```

index.js

funcao-soma

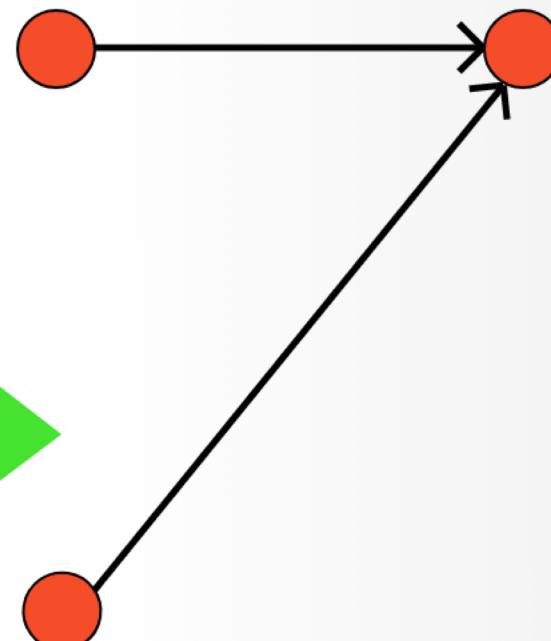
```
const func = (a, b) => a + b;
```

index.js

funcao-sub

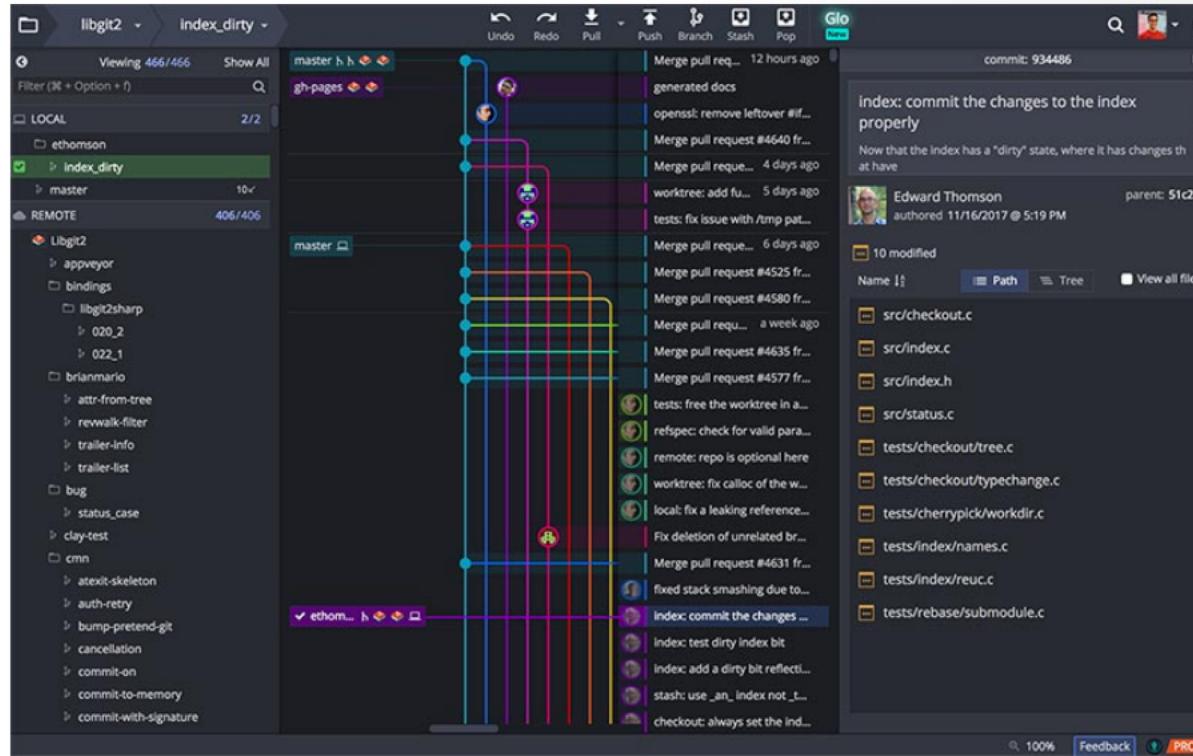
```
<<<<< HEAD  
const func = (a, b) => a - b;  
=====  
const func = (a, b) => a + b;  
>>>>> funcao-soma
```

index.js



RESOLVENDO CONFLITOS

- Alterando o arquivo manualmente
- IDEs (Visual Studio, WebStorm), Editores de Textos
- Ferramentas gráficas (GitKraken, por exemplo)



axosoft
GitKraken

OBRIGADO!

Contato ?

leonardobritobittencourt@gmail.com