Git

Monitoria - Engenharia de Software

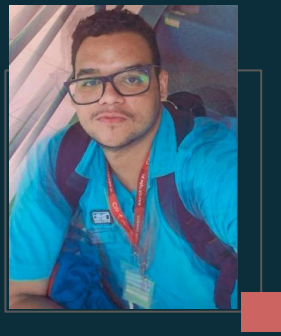
Nosso Time

Diego Rafael Gomes (drgf)

7° Período de SI

Pauline Duarte (pvad)

7° Período de SI



Alexandre Cabral (asc5)

7° Período de SI

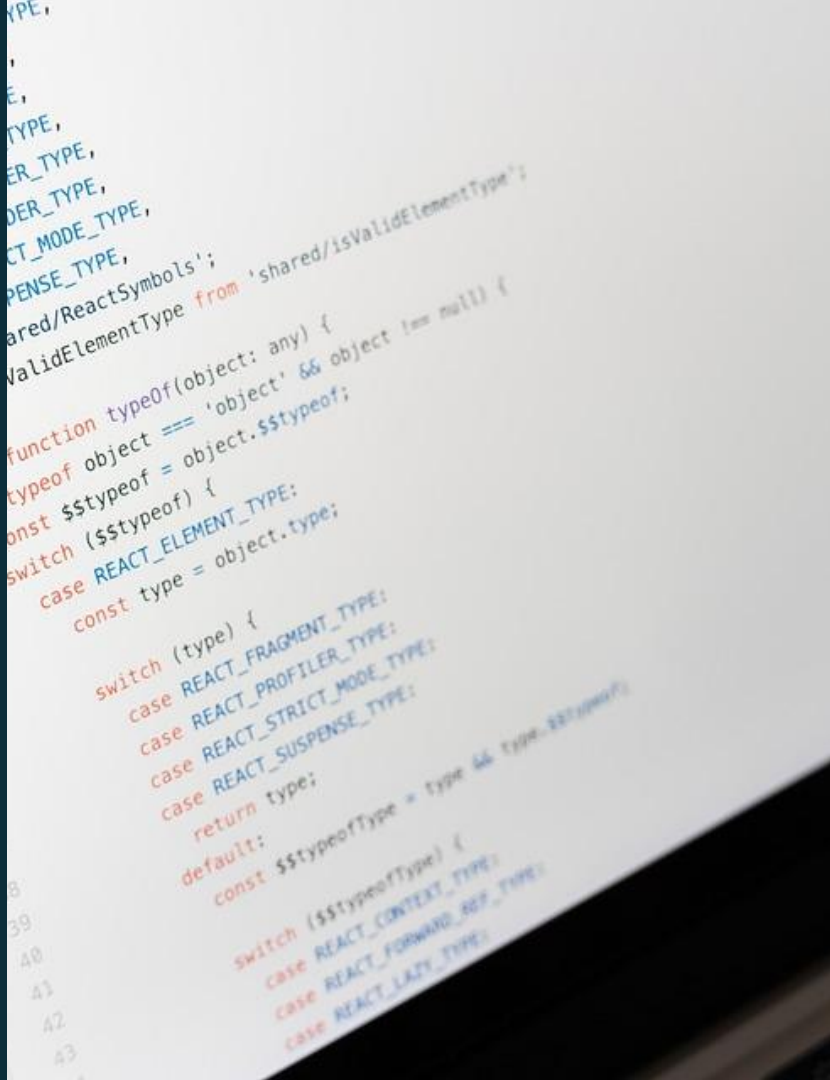
Roteiro da monitoria

Topicos abordados durante a monitoria:

1. O que é o Git?
2. Gitflow
3. Comandos
4. Conflitos

1

O que é Git?


function typeOf(object: any) {
typeof object === 'object' && object !== null) {
const $$typeof = object.$$typeof;
switch ($$typeof) {
case REACT_ELEMENT_TYPE:
const type = object.type;
switch (type) {
case REACT_FRAGMENT_TYPE:
case REACT_PROFILER_TYPE:
case REACT_STRICT_MODE_TYPE:
case REACT_SUSPENSE_TYPE:
return type;
default:
const $$typeofType = type && type.
switch ($$typeofType) {
case REACT_CONTEXT_TYPE:
case REACT_FORWARD_REF_TYPE:
case REACT_LAZY_TYPE:

“Git é um sistema de controle de versão distribuído gratuito e de código aberto projetado para lidar com tudo, desde projetos pequenos a muito grandes com velocidade e eficiência.”

Versionamento

Rede social de fotos

Feature de vídeo

É insta ou t1k t0k?

Versionamento

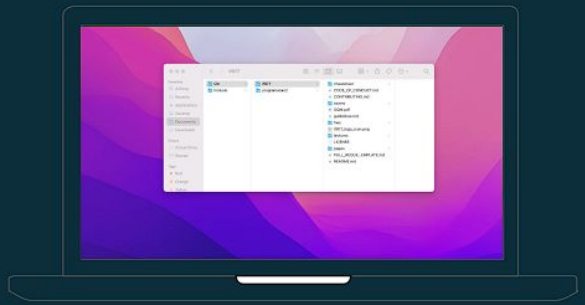
O versionamento evita perdas!

Permite consultas de versões antigas e até mesmo voltar atrás de uma má decisão.

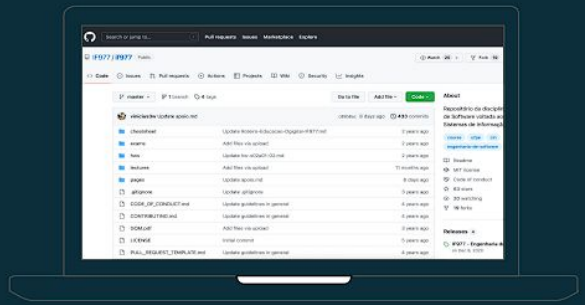
Como o GIT ajuda no controle de versões?

Repositórios

“Um repositório de software é um local de armazenamento de onde pacotes de software podem ser recuperados e instalados em um computador.” – Primeiro resultado do google



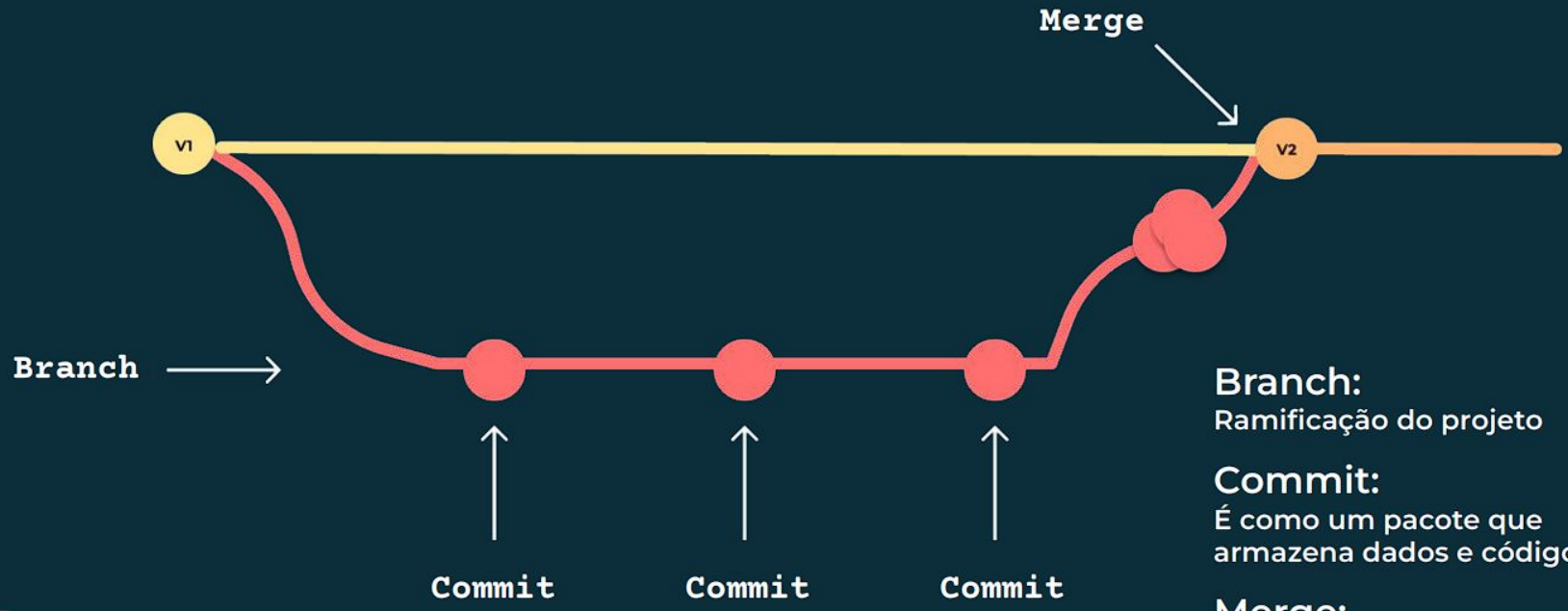
Locais



Remotos


git
Local
Local
Local
Remoto
ATLASSIAN
Bitbucket
GitLab
GitHub

Gerenciamento de código fonte

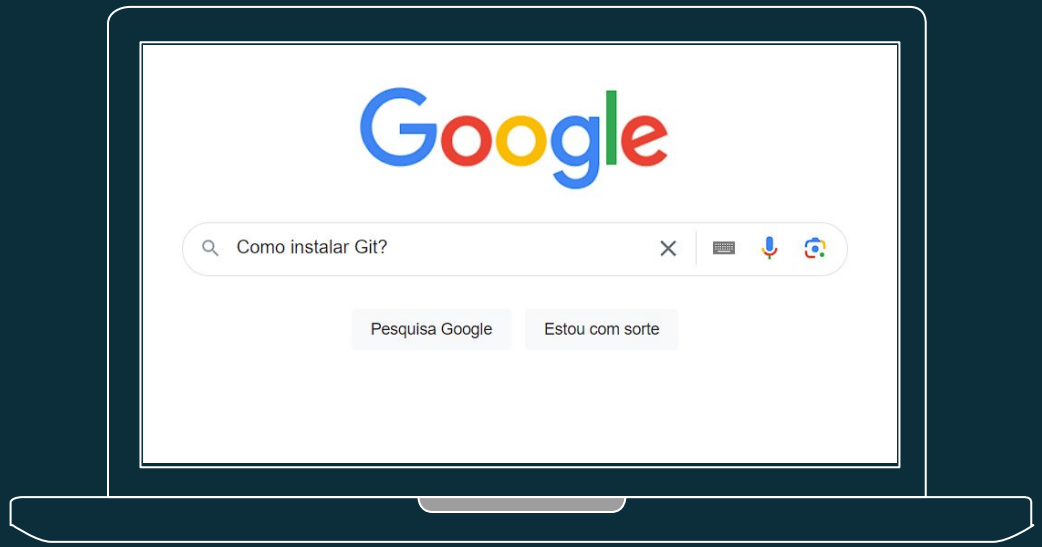


Branch:
Ramificação do projeto

Commit:
É como um pacote que armazena dados e códigos

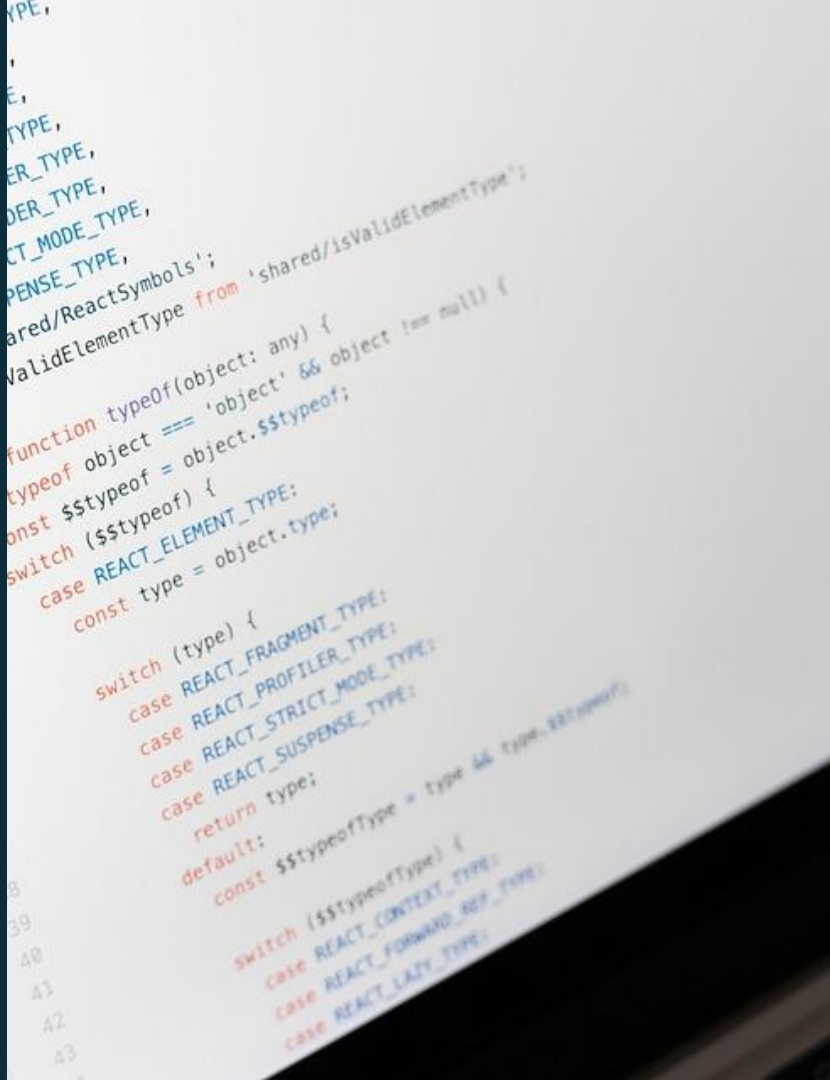
Merge:
Mesclagem de duas branches

Instalando Git



2

Gitflow


function typeOf(object: any) {
const $$typeof = object.$$typeof;
switch ($$typeof) {
case REACT_ELEMENT_TYPE:
const type = object.type;
switch (type) {
case REACT_FRAGMENT_TYPE:
case REACT_PROFILER_TYPE:
case REACT_STRICT_MODE_TYPE:
case REACT_SUSPENSE_TYPE:
return type;
default:

O que é o gitflow?

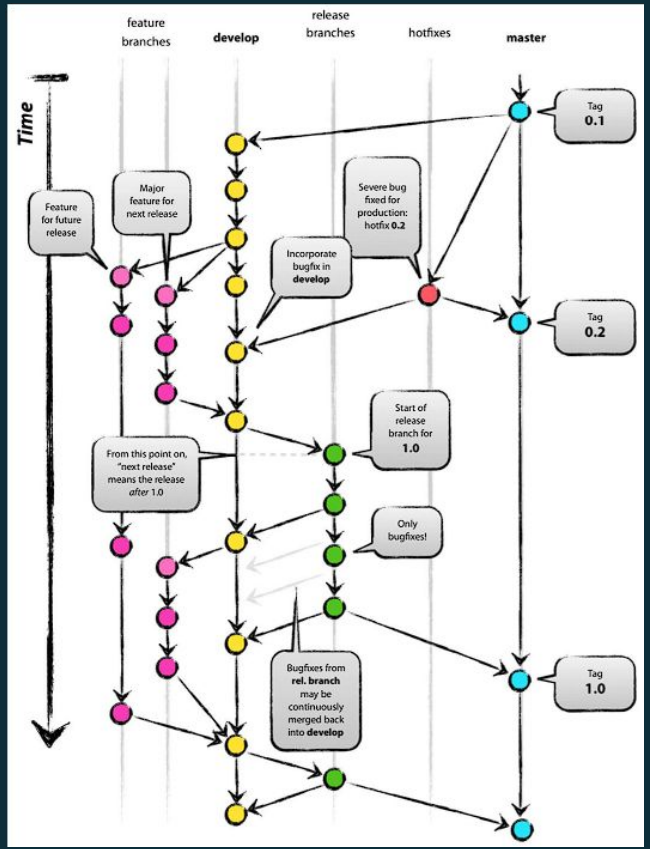
Um* modelo, ou workflow, que tem o objetivo de auxiliar na organização do versionamento de código

*Um dos inúmeros. Muitos se baseiam no gitflow e melhoram para sua realidade

Vincent Driessen

O que foi proposto

- Master/Main
- Develop
- Release
- Hotfixes
- Feature

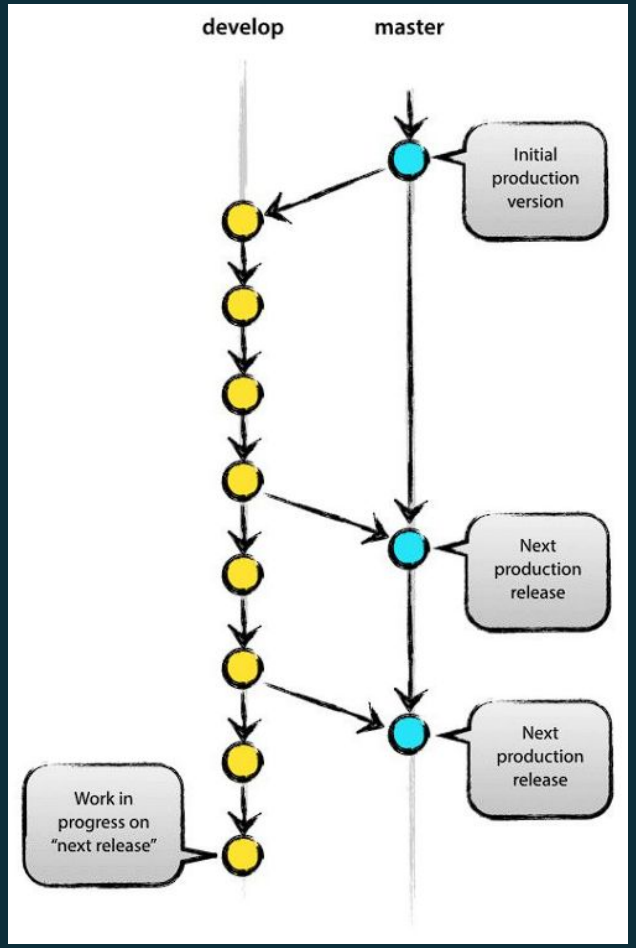


Master/Main e Develop

Principais

```
Nomenclatura proposta

master
develop
```

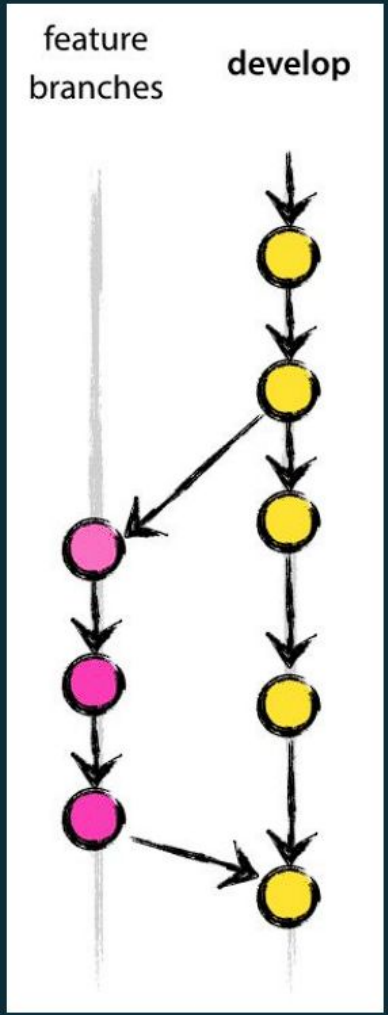


Feature

Nomenclatura proposta

Qualquer nome exceto:

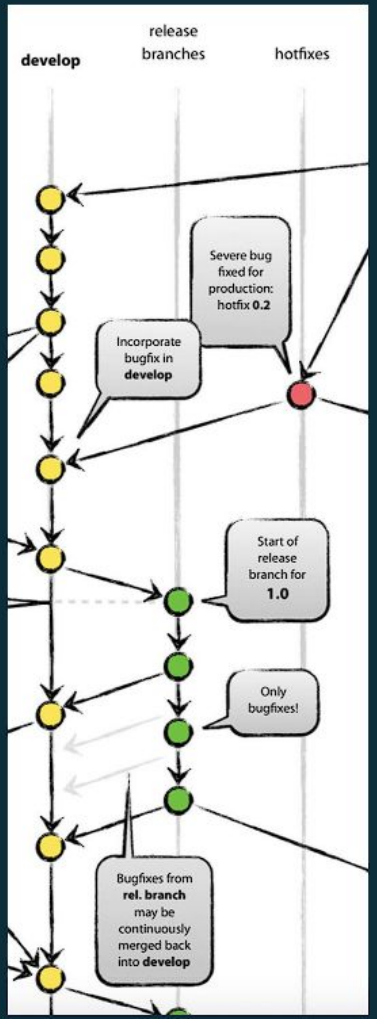
master
develop
release-*
hotfix-*



Release

Nomenclatura proposta

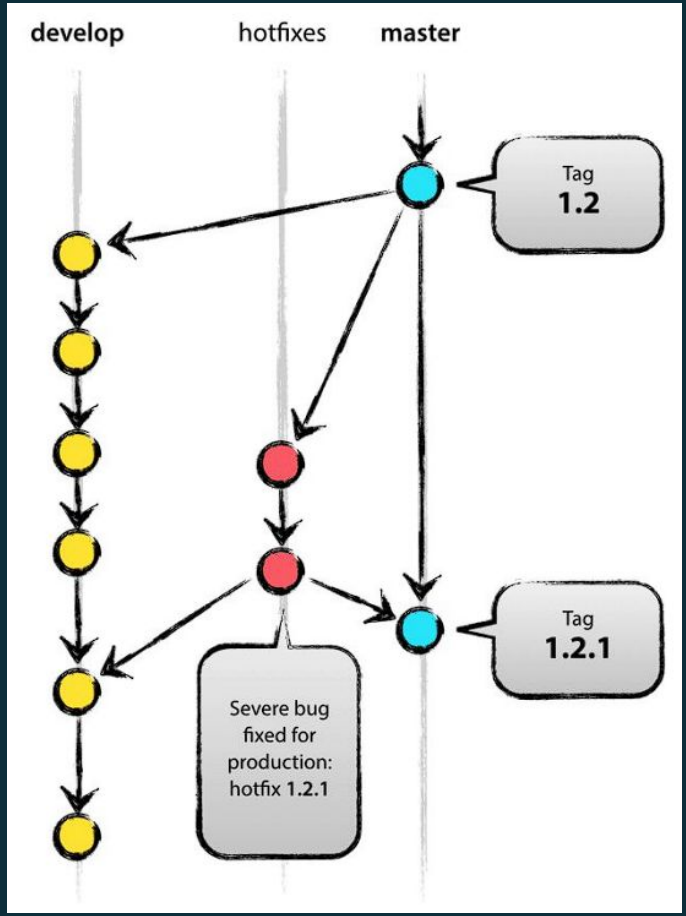
release-*



Hotfixes

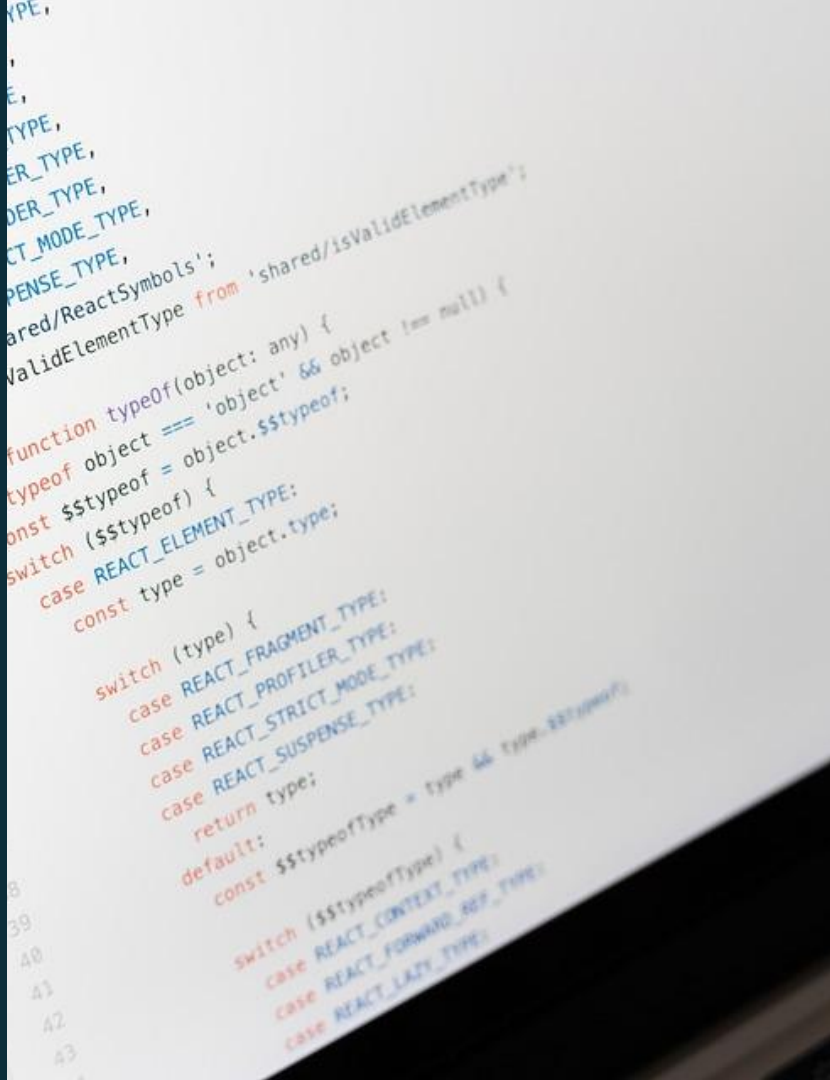
Nomenclatura proposta

hotfix-*



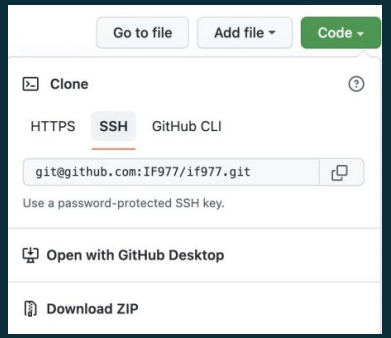
3

Comandos


function typeOf(object: any) {
const $$typeof = object.$$typeof;
switch ($$typeof) {
case REACT_ELEMENT_TYPE:
const type = object.type;
switch (type) {
case REACT_FRAGMENT_TYPE:
case REACT_PROFILER_TYPE:
case REACT_STRICT_MODE_TYPE:
case REACT_SUSPENSE_TYPE:
return type;
default:

git init
- criar um novo repositório no git

git status
- exibir possíveis alterações no seu repo



git clone <url-repo-remoto>
- clonar/baixar um repo remoto

git checkout -b <nome-branch>
- criar uma nova branch

git checkout <nome-branch>
- mudar para uma branch já existente

git add <nome-do-arquivo>

- Adiciona as mudanças desse arquivo à “staging area”, que significa que aquelas mudanças serão incluídas no próximo commit

git commit -m “<mensagem-de-commit>”

- Guarda o estado do seu repositório, incluindo as mudanças que estavam na “staging area”, com uma mensagem

git push <alias> <nome-da-branch>

- Envia sua branch para o repositório remoto contendo todos os commits realizados nela

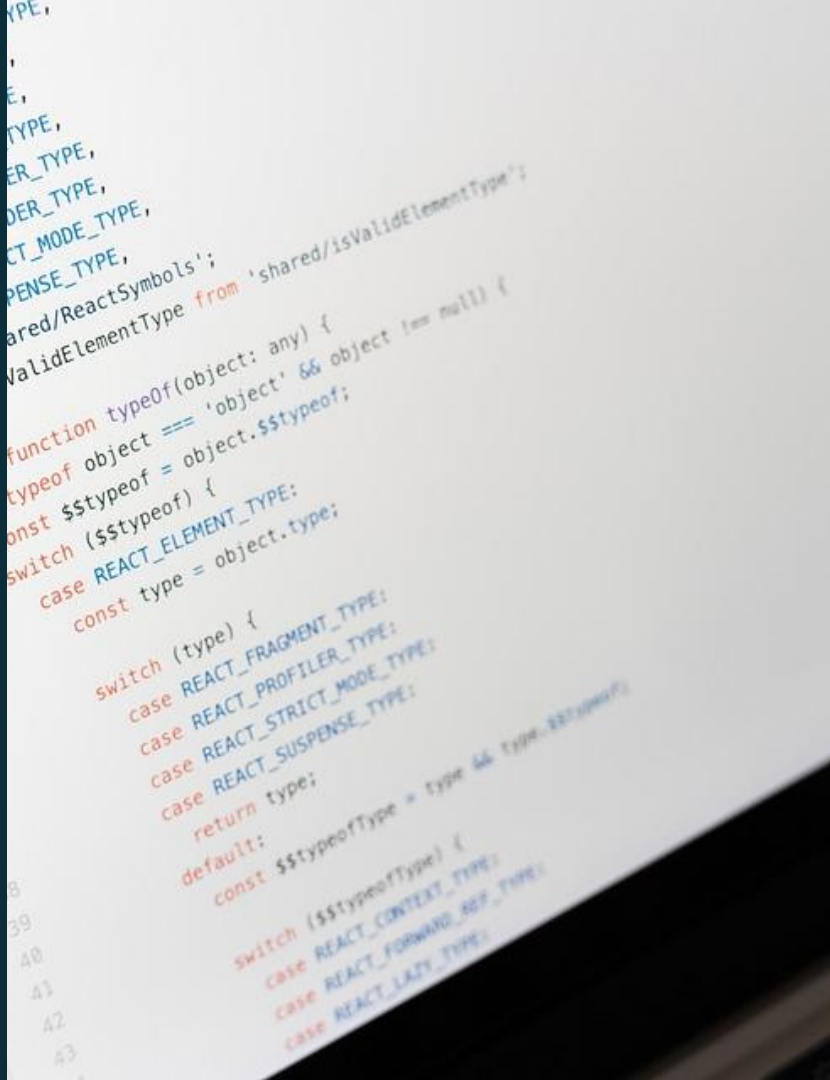
git pull
- “Puxar” informações de uma branch que estão no repositório remoto

git branch
- Listar todas as branches que estão no seu repositório local

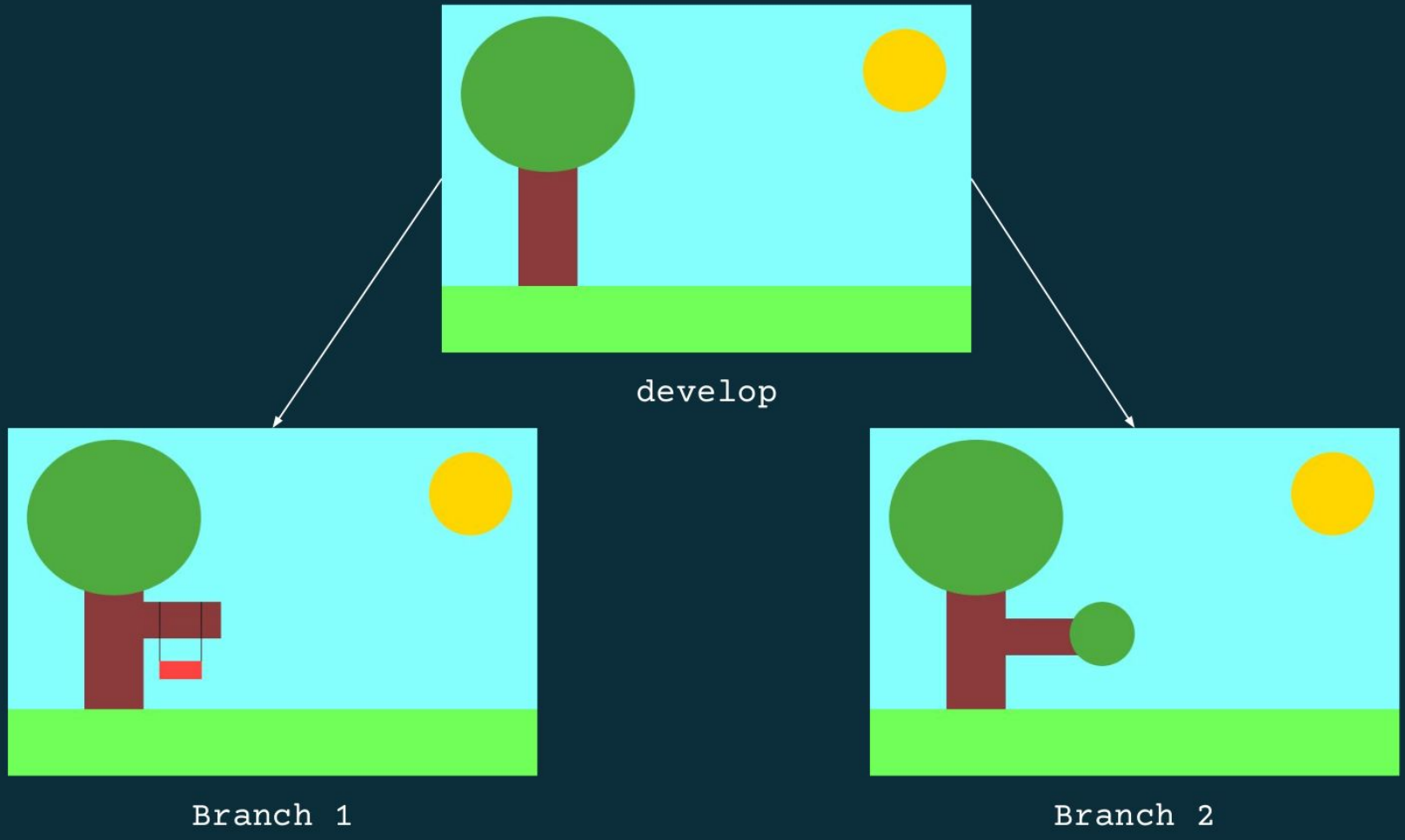
git branch -d <nome-branch>
- Deleta uma branch do seu repositório local

4

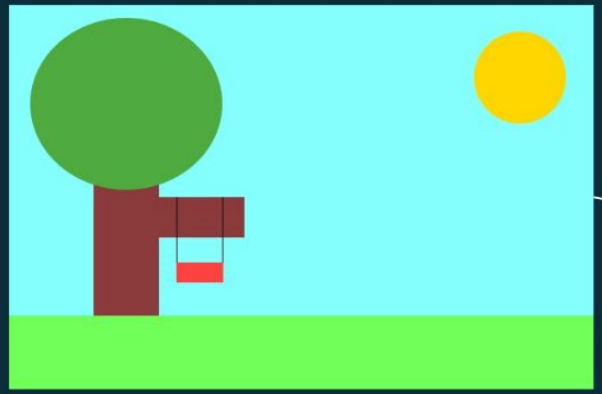
Conflitos


TYPE,
ER_TYPE,
DER_TYPE,
CT_MODE_TYPE,
PENSE_TYPE,
ared/ReactSymbols';
ValidElementType from 'shared/isValidElementType';
function typeOf(object: any) {
typeof object === 'object' && object !== null) {
const $$typeof = object.$$typeof;
switch ($$typeof) {
case REACT_ELEMENT_TYPE:
const type = object.type;
switch (type) {
case REACT_FRAGMENT_TYPE:
case REACT_PROFILER_TYPE:
case REACT_STRICT_MODE_TYPE:
case REACT_SUSPENSE_TYPE:
return type;
default:
const $$typeofType = type && type.
switch ($$typeofType) {
case REACT_CONTEXT_TYPE:
case REACT_FORWARD_REF_TYPE:
case REACT_LAZY_TYPE:

Um conflito acontece quando duas ramificações diferentes editam a mesma linha de código.

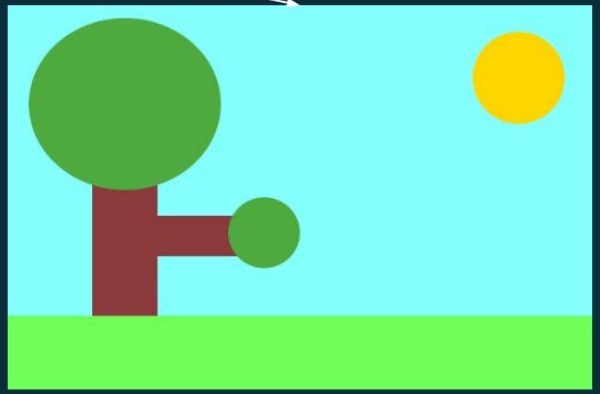

develop
Branch 1
Branch 2

Branch 1 foi mesclada na develop



develop

Conflito!!!

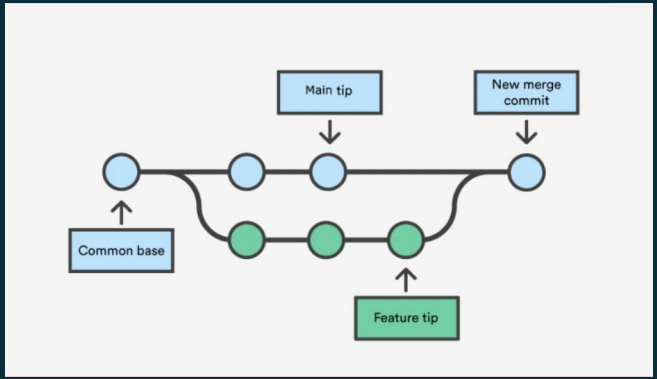


Branch 2

Criada a partir da versão antiga da develop

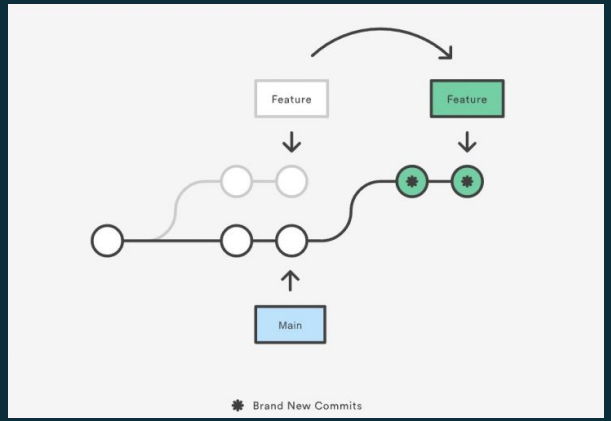
git merge <branch>

- Comando de resolução de conflitos que vai combinar várias sequências de commits em um histórico unificado

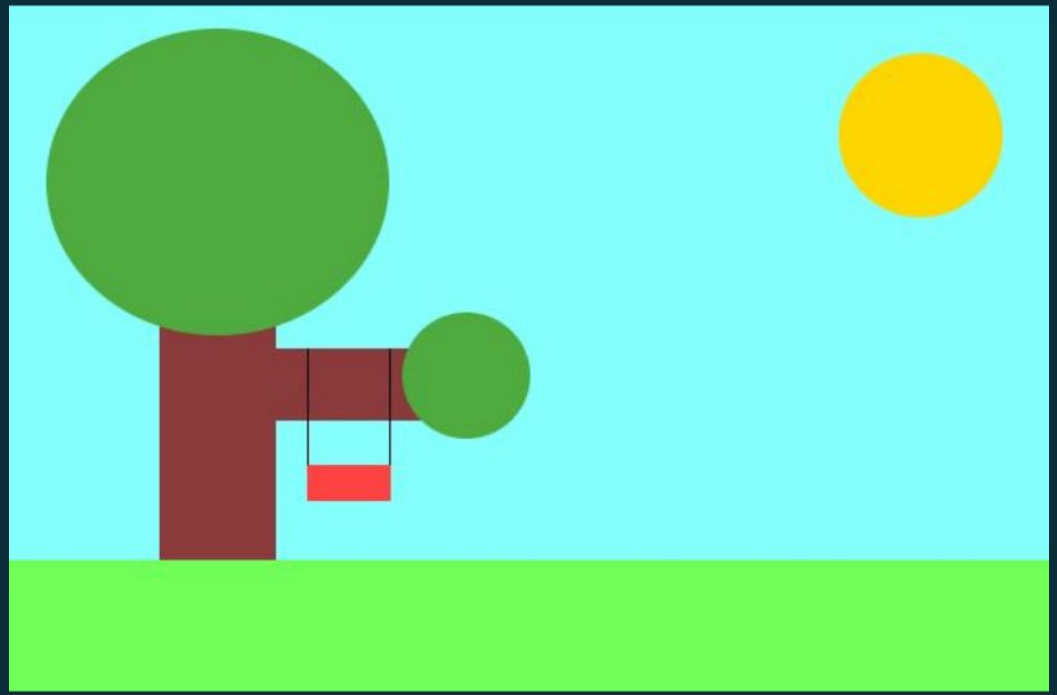


git rebase <branch>

- comando de resolução de conflitos que vai mover ou combinar uma sequência de commits para um novo commit base.



Conflito ✨ magicamente ✨ resolvido 😡#$!@ na força do ódio 😡#$!@

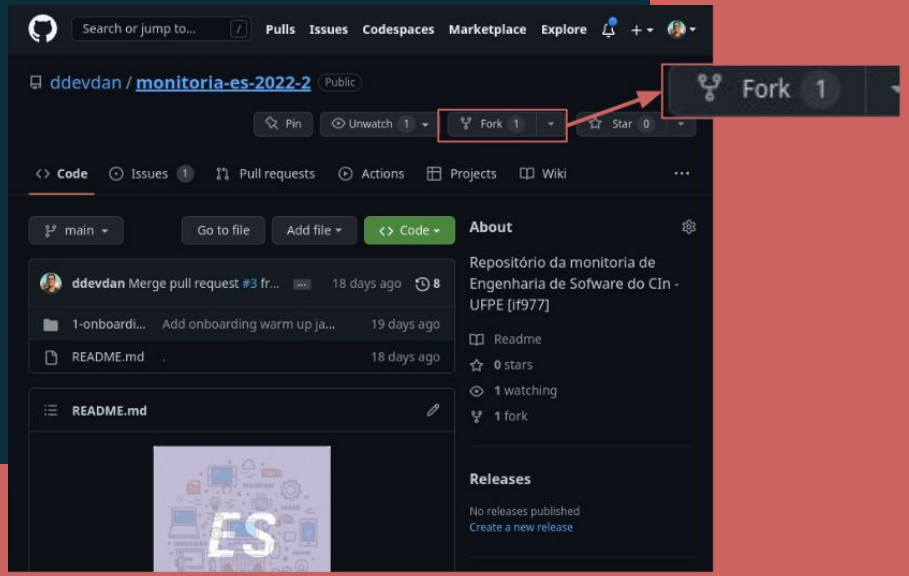


Fork

- É uma forma de você copiar um repositório e fazer alterações sem afetar o repositório original.

👀

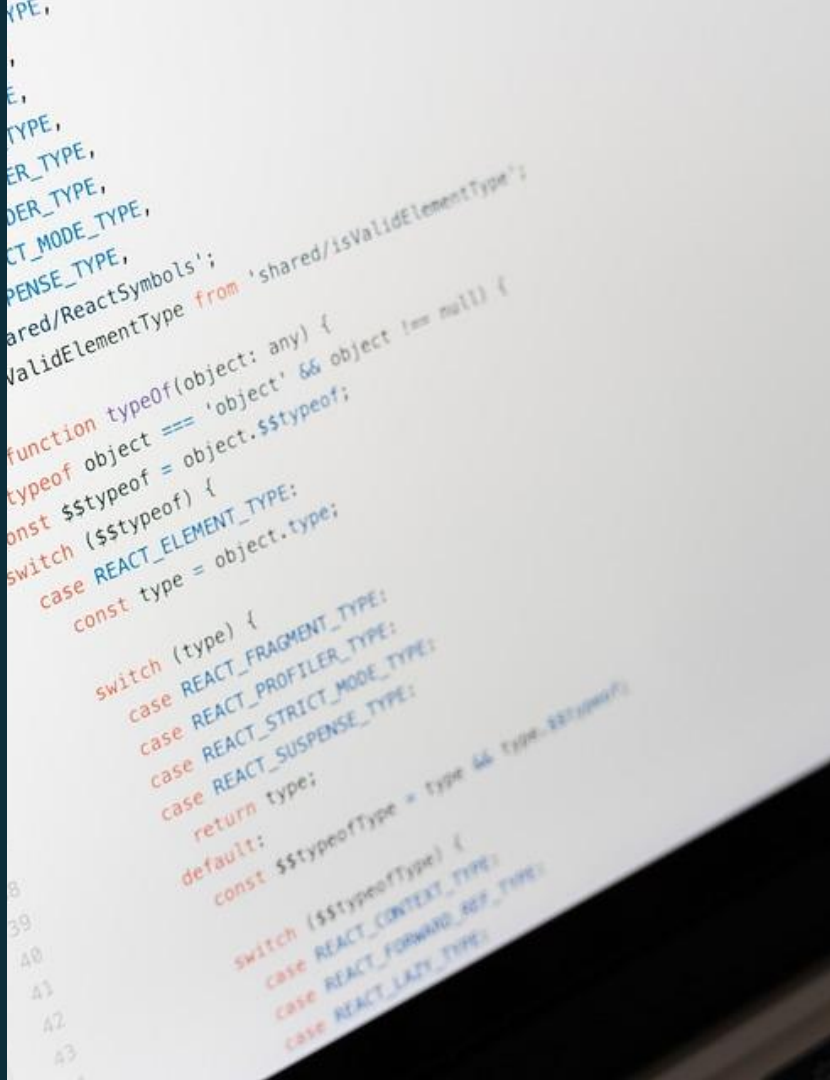
Vamos usar bastante Fork no projeto da monitoria.



How to resolve merge conflicts in Git

5

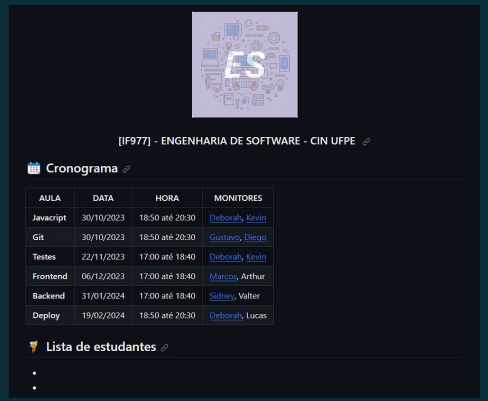
Desafio


function typeOf(object: any) {
typeof object === 'object' && object !== null) {
const $$typeof = object.$$typeof;
switch ($$typeof) {
case REACT_ELEMENT_TYPE:
const type = object.type;
switch (type) {
case REACT_FRAGMENT_TYPE:
case REACT_PROFILER_TYPE:
case REACT_STRICT_MODE_TYPE:
case REACT_SUSPENSE_TYPE:
return type;
default:
const $$typeofType = type && type.
switch ($$typeofType) {
case REACT_CONTEXT_TYPE:
case REACT_FORWARD_REF_TYPE:
case REACT_LAZY_TYPE:

Desafio: Adicionar nome ao readme do repositório da monitoria

> Fazer um Fork do Repositório
> Clonar o Repositório Forkado
> Configurar o Upstream
> Criar uma Nova Branch
> Fazer Alterações no README
> Manter a Sua Branch Atualizada
> Commitar > Dar push
> Abrir um Pull Request

Desafio: Adicionar nome ao readme do repositório da monitoria



[IF977] - ENGENHARIA DE SOFTWARE - CIN UFPE

Cronograma

AULA	DATA	HORA	MONITORES
Javacript	30/10/2023	18:50 até 20:30	Deborah, Kevin
Git	30/10/2023	18:50 até 20:30	Gustavo, Diego
Testes	22/11/2023	17:00 até 18:40	Deborah, Kevin
Frontend	06/12/2023	17:00 até 18:40	Marcos, Arthur
Backend	31/01/2024	17:00 até 18:40	Sidney, Valter
Deploy	19/02/2024	18:50 até 20:30	Deborah, Lucas

Lista de estudantes

-
-

Conteúdos e referências

Links:

- Site do Git - Documentação, Instalação & Outros Recursos
- O que é Git e Github? - Curso em Vídeo
- Artigo AWS - Gerenciamento de Código Fonte
- Artigo Alura - Gitflow
- Git Flow - Lean Pub (en)
- Artigo Original: Gitflow por Vincent Driessen (en)
- Lista Completa de Todos os Comandos do Git
- Principais Comandos com Suas Categorias
- Artigo sobre Merge do Bitbucket
- Artigo sobre Rebase do Bitbucket
- Diferença Entre Git Rebase e Git Merge - @kevbeltrao
- Jogo 'Oh My Git!'

Obrigado!

Perguntas?