**Git**, um sistema de controle de versão distribuído (DVCS)

Git – mecanismo de controle de versão;

GitHub / Gitbucket / Gitlab – plataformas que hospedam repositórios do GIT;

Branches – versão independente e editável que se cria de um código;

Branche Master / Main – Branch principal do código, ela será upada; É editável.

Pode-se criar commit e agregar a master.

Merge – Une alterações feitas em duas branches os mais

GIT INIT – toda vez que vai iniciar um novo projeto;

GIT STATUS – mostra status dos arquivos

GIT ADD – Criar algo para enviar [pegando uma cx vazia e adc coisas para enviar, criando o que vai enviar]

GIT COMMIT – [embalando, colocando as informações que eu preciso para enviar]

Vários adds para 1 commit ou 1 pra 1 .

GIT LOG – Mostra os commits

GIT -AM – add + commit

GIT COMMIT –-AMEND – EDITAR COMMIT

git commit --amend -m "an updated commit message" – atualiza mensagem do último commit

GIT BRANCH “nome branch” – Criar branch

GIT CHECKOUT “nome branch” – Mudar de branch

GIT MERGE “Nome branch” – Une branchs

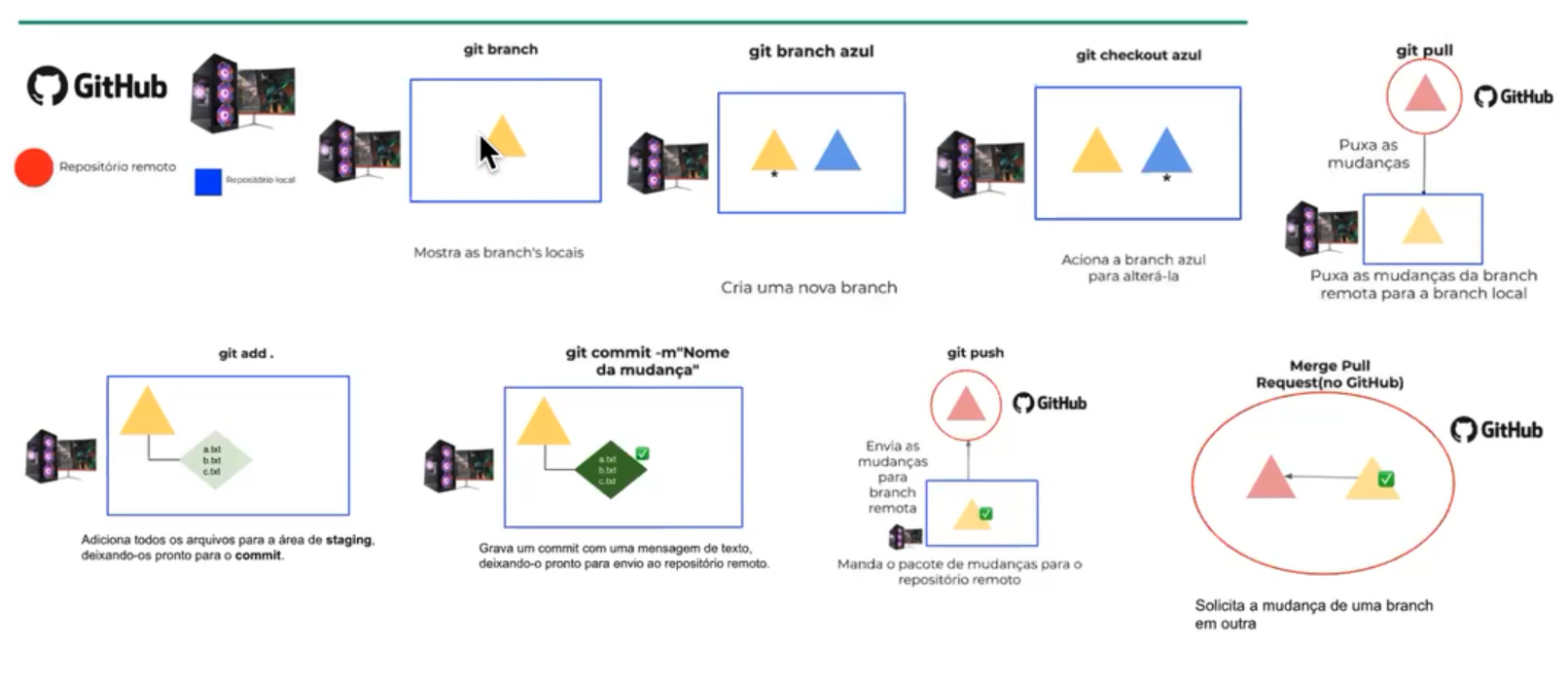
GIT BRANCH (nome branch) -D – Deleta uma branch

GIT CHECKOUT -B (nome branch) – Cria branch e vai até ela

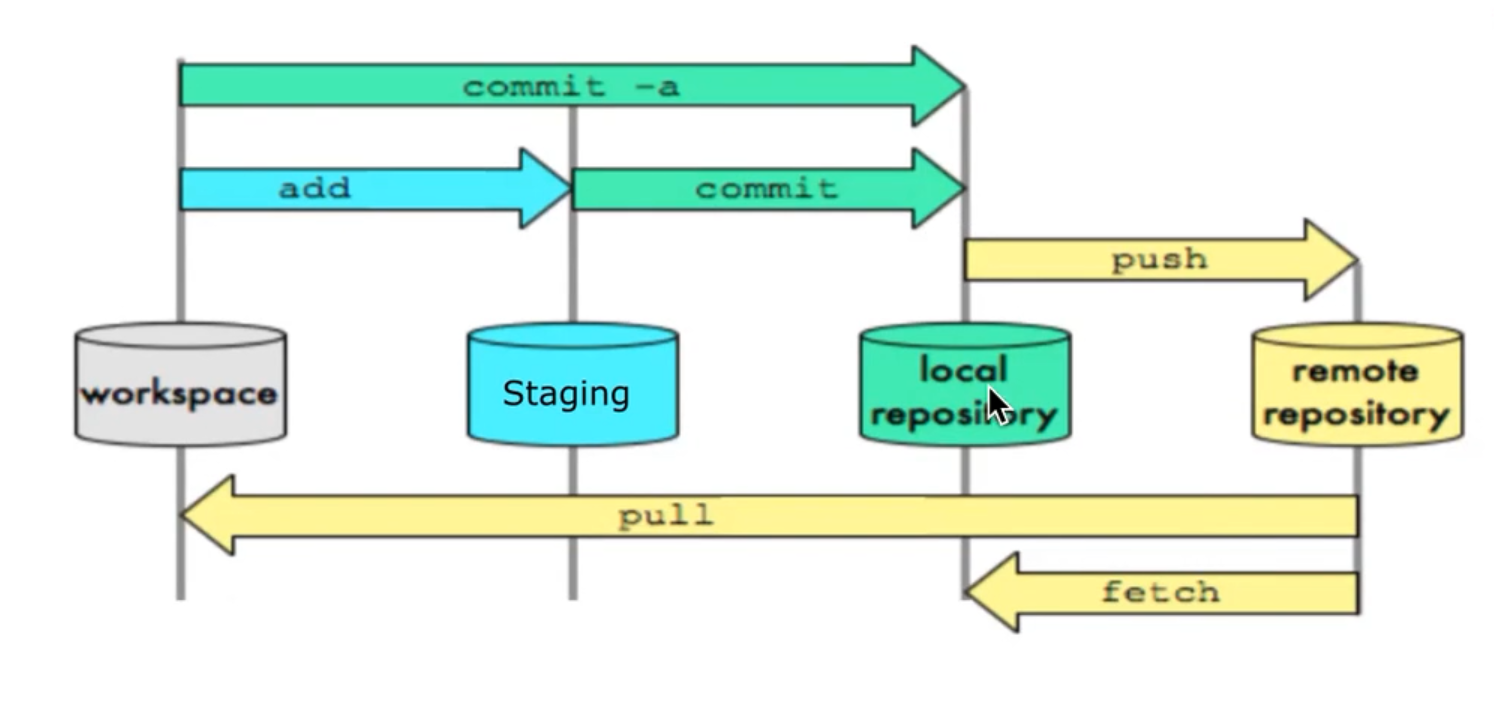
GIT PULL – PUXA AS MUDANÇAS DA BRANCH REMOTA PARA A LOCAL

GIT PUSH -U ORIGIN (NOME BRUNCH) – ENVIA AS MUDANÇAS FEITAS LOCAL PARA REMOTO

GIT FETCH – MOSTRA SE TEVE ALTERAÇÕES NO REMOTO







Recapitulando

Agora você já pode esquecer que um dia versionou os arquivos adicionando o nome da versão no final! 😉

O Git é um sistema de versionamento distribuído (DVCS);

A maior vantagem do Git é que ele é distribuído, ou seja, o repositório fica na sua máquina e você não precisa se preocupar em fazer "lock" de arquivos assim como em outros VCS's ;

Vimos que a instalação do Git e sua configuração são simples. Basta baixar o instalador e configurá-lo usando o comando git config ;

Para iniciar um repositório local, vimos que você tem duas opções:

git init (para criar um novo repositório Git ) ;

git clone (para copiar um repositório Git existente para sua máquina e, assim, poder criar branches e Pull Requests) .

O GitHub é uma plataforma de hospedagem de código-fonte com controle de versão usando o Git;

Vimos também que, para subir seu código para o GitHub , basta você verificar qual das opções abaixo deseja:

Criar um repositório do zero localmente, na sua máquina, onde utilizará o comando git init ;

Trabalhar num repositório já existente, onde utilizará o comando git clone ;

Esses dois comandos são excludentes , ou seja, você usa um, ou usa o outro, nunca os dois ao mesmo tempo. Após isso, siga um dos dois fluxos abaixo

Fluxo 1 - Após git init :

Criar seu primeiro commit ;

Criar o repositório remoto na sua conta do GitHub ;

Adicionar a URL do repositório remoto;

Executar o famoso push .

Fluxo 2 - Após git clone :

Criar sua branch dentro do repositório clonado;

Fazer checkout na sua branch;

Criar seu primeiro commit ;

Executar o famoso push .