

Controle de Versão

Versionamento de código para se ter mais controle sobre o que está programando

O Controle de versão

- O Que é ? - é um software que gerencia mudanças nos códigos fontes, gerando uma nova versão a cada alteração.
- Controla o histórico, facilitando desfazer uma alteração, proporciona o trabalho em equipe.
- Facilita instalações.
- Diminui o risco de perder códigos.

GIT

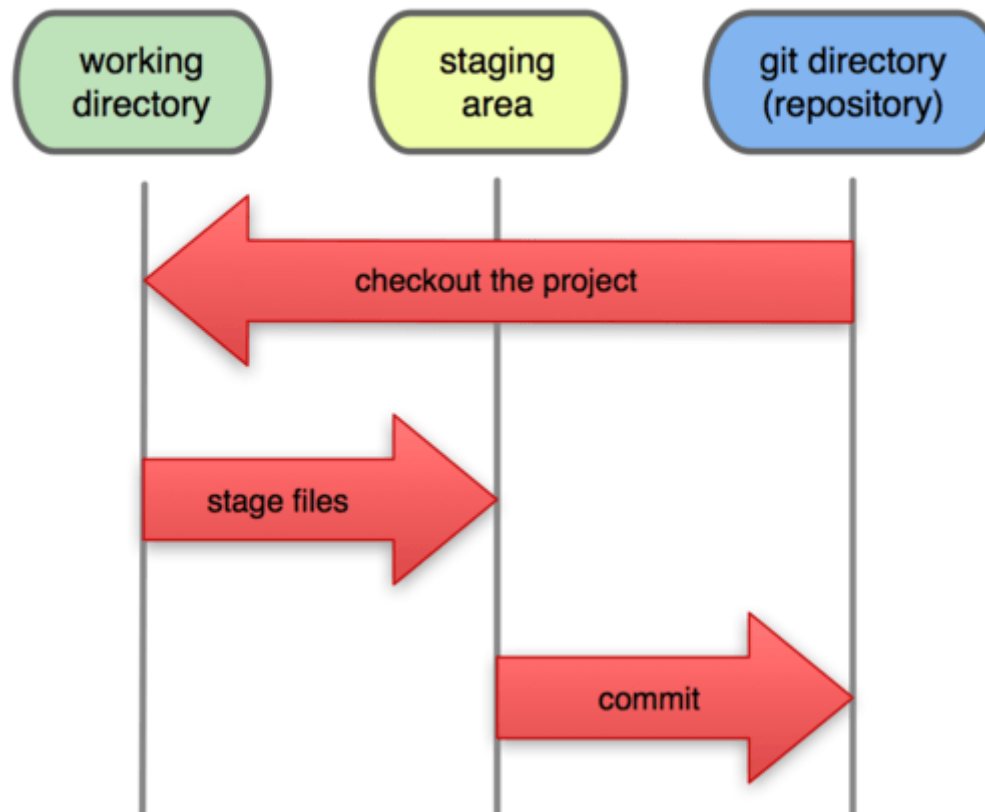
- Snapshots – Cria “fotos” do estado dos arquivos do projeto a cada Commit.
- Checksum – Verificação dos arquivos (SHA1).

3 Estados

- As operações com o GIT ficam entre 3 estágios, sendo:
 - Working directory – o estado atual do trabalho, arquivos reais onde se está trabalhando.
 - Staging area – o meio termo entre o working directory e repositório, os arquivos que estão modificados, criados, ou alterados, mas ainda não foram confirmados.
 - Repository – Local onde os arquivos confirmados e versionados estão.

3 Estados

Local Operations



Fluxo básico do trabalho

- O fluxo de trabalho, ou workflow no GIT, pode ser descrito da seguinte maneira:
 - Você modifica arquivos no seu diretório de trabalho.
 - Você seleciona os arquivos, adicionando snapshots deles para sua área de preparação.
 - Você faz um commit, que leva os arquivos como eles estão na sua área de preparação e os armazena permanentemente no seu diretório Git.

Repositórios

- Locais, quando o repositório está na própria máquina onde o programador trabalha e faz o versionamento dos códigos.
- Remoto, é um repositório igual ao local, todavia centralizado em algum servidor.

Trabalhando com GIT

- O git é um programa que apesar de poder ser integrado as diversas IDEs que utilizamos, pode ser operado individualmente, através de linhas de comando em um ambiente shell.
- Para trabalhar com programas que operam por linhas de comando (shell). Normalmente enviamos um comando e aguardamos a resposta, assim sucessivamente.

Principais comandos

- Ajuda – o primeiro comando a se saber, é o de ajuda, pois assim pode-se obter auxílio com os demais comandos quando necessário.

usuari@pc ~/ \$ Terminal

- \$ git help <comando> (Enter)

Iniciando um Repositório

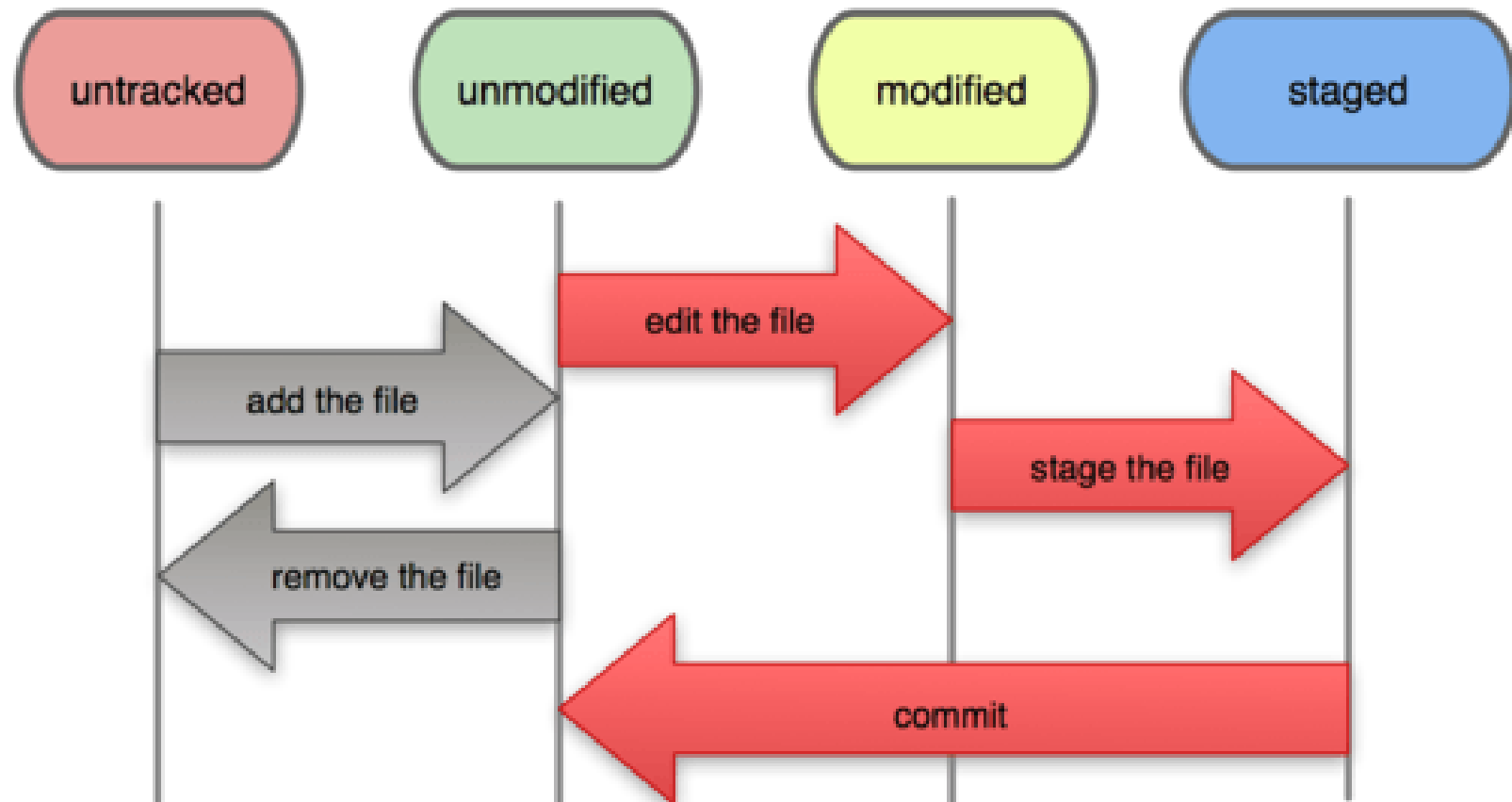
- `$ git init` - Inicializa um repositório, dentro do diretório que estiver.
- `$ git clone <repositório>` – clona um repositório criando uma pasta com o nome do repositório no diretório atual.
 - Se adicionar um quarto parâmetro, pode se alterar o nome do diretório onde o repositório será clonado.

GIT Ignore

- O ignore, como o próprio nome sugere, serve para ignorar (excluir) um item do commit (confirmação), fazendo com que este item mesmo que alterado no work directory, não seja enviado para o repository.
- Para ignorar, basta criar um arquivo chamado `.gitignore` na raiz do projeto, e em cada linha do arquivo especificar o que deseje-se ignorar.

Ciclo de vida dos arquivos

File Status Lifecycle



Verificando o estado dos arquivos

- `$ git status`
- Retorna o status dos arquivos.

Adicionando arquivos

- `$ git add <arquivo>` - Adiciona na stage area os arquivos desejados (pode-se utilizar coringas)

Confirmando arquivos

- `$ git commit -m "mensagem"`
- Realiza a confirmação, jogando os arquivos do staging area para o repository.

Movendo ou Removendo

- \$ git mv <arquivo origem> <arquivo destino> - move um arquivo da origem para o destino.
- \$ git rm <arquivo> - remove um arquivo.

Remotos

- Adicionando repositórios remotos ao repositório local.
- `$ git remote add <nome> <caminho>`

Enviando

- `$ git push <opções> <remoto> <ramificação>`
- Envia dados do repositório local para o remoto conforme informado nos parâmetros

Recebendo (atualizando)

- `$ git fetch <nome-remoto>`
- `$ git pull <nome-remoto> <ramificação>`
- Atualizado o work directory conforme o repositório remoto especificado.

Git Tag

- `$ git tag -a <nome> -m "comentario"`
- Adiciona um rótulo, uma marcação da versão.

Git Brach

- `$ git checkout -b <nome do branch>` - Cria um branch e muda para ele automaticamente.
- `$ git checkout <nome do branch>` - muda para um branch existente.
- `$ git branch -D <nome do branch>` remove um branch.

Mais sobre git

- <https://git-scm.com/>
- Repositórios remotos:
 - Github
 - BitBucket