Controle de versão Distribuído

Introdução

emerson@paduan.dev.br

Softwares

Git

https://ahttps://git-scm.com

GitHub

https://github.com/

emerson@paduan.dev.br

Contexto

Já passou por isso?

trabalho.doc
trabalho-v02.doc
trabalho-v03.doc
...etc..
trabalho-vfinal.doc
trabalho-vfinal-ultima.doc
trabalho-vfinal-ultima-mesmo.doc
trabalho-vfinal-ultima-mesmo-entregue.doc

O que é?

O que é controle de versão?

- Um sistema que mantém um registro das alterações realizadas, permitindo saber:
  - quais foram as alterações realizadas (histórico).
  - quando foram feitas as alterações
  - quem fez
- Permite REVERTER as alterações feitas
- Permite o desenvolvimento colaborativo

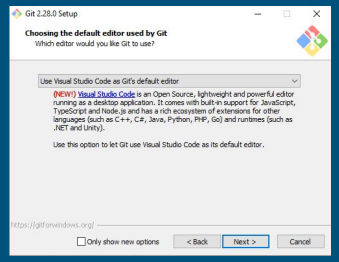
Por que Git?

Vantagens?

- Faz o controle de versão distribuída
- Usuários mantêm uma cópia do código completo em sua máquina local
- Desempenho
- Uma das mais utilizadas pelos desenvolvedores

Git

https://git-scm.com/downloads

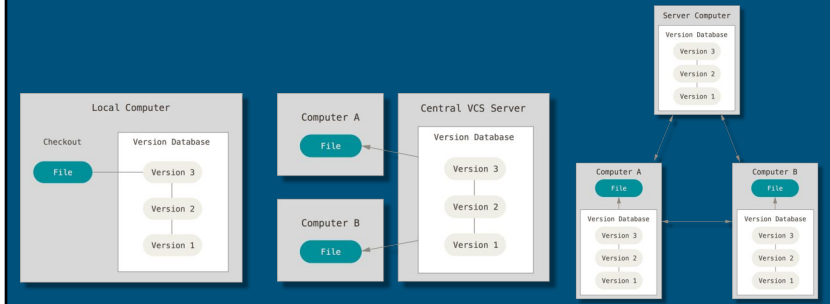


Controle de versão

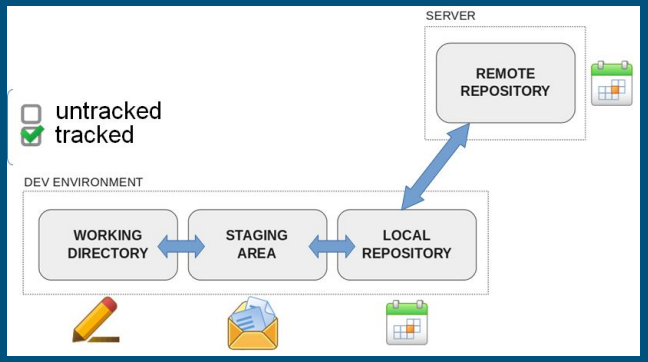
Repositórios

Conjunto de arquivos e o histórico de alterações daqueles arquivos.

Normalmente trata-se de todos os Snapshots de um projeto.



Resumo Git



Iniciando...

$> git init [diretório]

Cria um repositório GIT no diretório indicado, ou no diretório atual.

Configurações

Caso não estejam aparecendo as cores no terminal

$> git config –global color.ui auto

Configurando o usuário:

$> git config [–global] user.name seuNomeDeUsuário

$> git config [–global] user.email seuE-mail

Status

$> git status

Lista como estão os arquivos untracked, staged, unstaged.

Adicionando ao stage

$> git add <arquivo(s)>

$> git add * / git add .

Adiciona dos arquivos *modificados* para o stage.

Commit

$> git commit –m "mensagem do commit"

Cria um commit (snapshot) dos arquivos que estão no stage.

Um commit é identificado por um texto informado pelo programador e um ***hash code***.

Log

$> git log

Mostra o histórico dos commits.

Repositórios Git

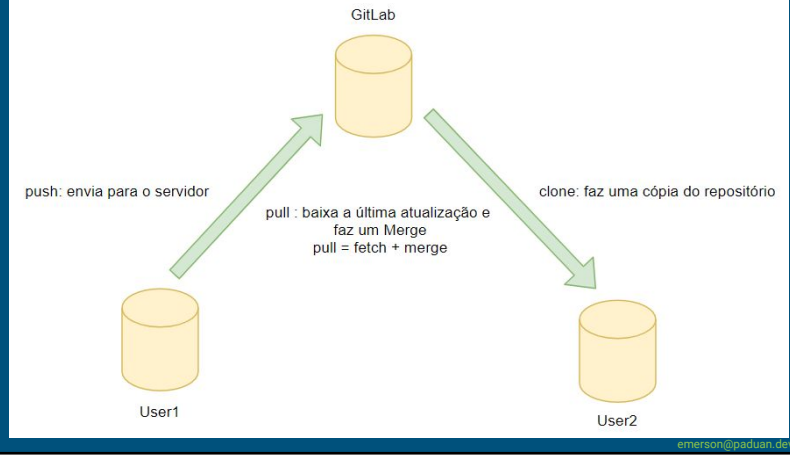
GitHub

https://github.com/

GitLab

https://gitlab.com/

emerson@paduan.dev.br



emerson@paduan.dev.br

Remote

$> git clone <url do repositório remoto>
Clona na máquina local um repositório remoto

$> git remote add origin <url remoto>
Adiciona (configura) um repositório remoto para executar o push.

$> git push [origin] [branch]
Envia o commit branch para o origin remoto.

$ git pull <remote>
Faz uma cópia da branch atual do repositório remoto, para o repositório local fazendo um merge na cópia local.