



Fullstack - Git e Github

Prof. Renan Ponick



Agenda

22/07 - Introdução + Git configuração

23/07 - Git comandos e Github SSH

29/07 - Avaliação Git/GitHub

30/07 - CLAUDIA

Review

Os Problemas

- Preciso voltar o arquivo na versão que estava antes - rollback - Ctrl+Z;
- Deleção de arquivo sem a possibilidade de recuperação;
- Cópias e mais copias de um arquivos:
 - tcc.docx;
 - tcc-final.docx;
 - tcc-agora-vai.docx;

A solução

Controle de Versão

Os sistemas de controle de versão são ferramentas de software que ajudam as equipes de software a gerenciar as alterações ao código-fonte ao longo do tempo;





E o GitHub?

Os Problemas

- Como trabalhar em equipe, utilizando o controle do Git, para que tenhamos um controle de versão?
- Como compartilhar o conhecimento(o código fonte)?



GitHub

Serviço web compartilhado para projetos que utilizam o Git para versionamento. É um local de armazenamento de código.



git

Controle de Versão **Local**

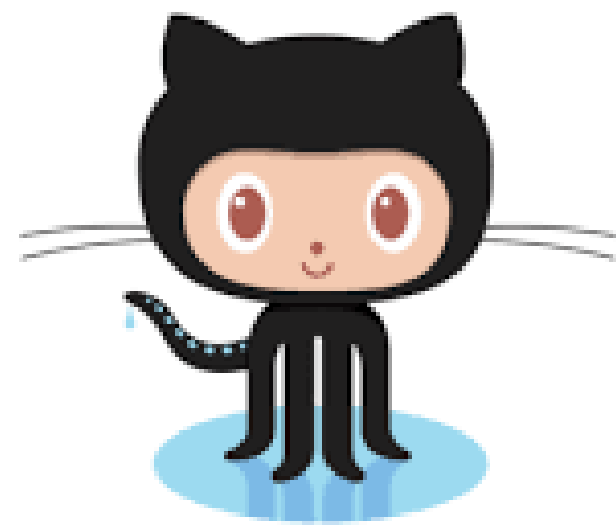
!=



GitHub

Repositório **remoto**

Check Point



GitHub



git

Quem veio primeiro?
Um poderia existir sem o outro?
Eles são os únicos?

Comandos Git

Abra o GitBash

git init

git status

git add <filename ou . >

git rm --cached <file> / git restore --staged <filename ou . >

git branch

git checkout <branchname>

git checkout -b <branchname>

git commit -m "<description>"

git push

git branch -D <branchname>

git fetch

git pull



git init

Inicializa um novo repositório Git em um diretório existente.

Cria um subdiretório .git que contém todos os arquivos necessários para o repositório.

Para desfazer a inicialização, exclua o diretório .git.



git status

Exibe o estado atual do repositório.

Mostra arquivos modificados, novos arquivos e arquivos preparados para commit.

Não há necessidade de reverter, pois apenas exibe informações.



`git add <filename ou . >`

Adiciona arquivos ao índice (staging area).

Prepara os arquivos especificados para o próximo commit.

Use `git rm --cached <file>` para remover os arquivos da staging area.



```
git rm --cached <file>
```

Remove arquivos do índice (staging area).

Os arquivos são removidos da staging area, mas não do diretório de trabalho.

Adicione novamente os arquivos com `git add <filename ou . >`.



git branch

Lista todas as branches locais existentes

git branch -r Lista todas as branches remotas

git branch -a Lista todas as branches remotas e locais




git checkout <branchname>

Muda para a branch especificada.

O diretório de trabalho é atualizado para refletir o estado da branch.

Para voltar à branch anterior, use git checkout <previous-branchname>.



`git checkout -b <branchname>`

Cria uma nova branch e muda para ela.

Uma nova branch é criada e você é movido para essa nova branch.

Use `git branch -D <branchname>` para excluir a branch e `git checkout <previous-branchname>` para voltar à branch anterior.

`git commit -m "<description>"`

Faz um commit das mudanças no índice com uma mensagem descritiva.

As mudanças na staging area são salvas no repositório com a mensagem fornecida.

Use `git reset --soft HEAD~1` para desfazer o último commit mantendo as mudanças no índice, ou `git reset --hard HEAD~1` para desfazer o commit e as mudanças.



git merge <branch>

Realiza o merge da branch ATUAL na Branch informada



git push

Envia commits do repositório local para o repositório remoto.

Os commits locais são enviados para a branch correspondente no repositório remoto.

Use `git revert <commit-hash>` para criar um novo commit que desfaz as mudanças do commit específico.



git branch -D <branchname>

Exclui a branch especificada.

A branch é removida do repositório local.

Não há como reverter diretamente. Se necessário, crie a branch novamente a partir de um commit específico.



git fetch

Baixa objetos e referências do repositório remoto.

Atualiza o repositório local com informações do repositório remoto sem mesclar.

Não há necessidade de reverter, pois apenas atualiza as referências locais.



git pull

Baixa objetos e referências do repositório remoto e mescla com a branch atual.

Mescla as mudanças do repositório remoto na branch local.

Use `git reset --hard <commit-hash>` para reverter o estado da branch local ao commit anterior ao pull.

GitHub - Prática

Recomendação



Agora tem ambos, Git e Github, configurados recomendo criar um repositório no GitHub para suas anotações de estudo (independente a disciplina ou instituição).

Dessa forma conseguirá realizar anotações durante as aulas, inclusive deixar exemplos práticos no mesmo, sem se preocupar em perder os arquivos ou em ter muitos arquivos em locais diferentes

Adicione uma pasta nova de Git/GitHub e um Arquivo .md para colocar todos os comandos e regras de commit, assim fica tudo centralizado em um unico local.

Atividade

Para que seu perfil como dev seja notado, é interessante ter um portfólio montado. Utilizaremos o git e github para versionamento do mesmo.

De maneira simples, vamos criar um arquivo index.html contendo suas informações igual um currículo.

Criaremos também uma branch dev e protegeremos ela contra exclusão.