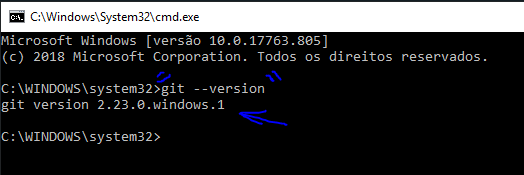
Instalação do GIT para versionamento de Código

* Primeiramente deve ser baixado o programa do GIT no site <https://git-scm.com/>;
* Depois é só executar o arquivo baixado;
* Para verificar a versão do git que foi instalado é necessário abrir primeiramente o CMD ou Prompt de Comando ou o Terminal e executar o comando “ git --version ”



Configurando o GIT

* Sempre usando o CMD
* Colocando o nome de usuário:

git config --global user.name "nome do usuario"

* Colocando o e-mail:

git config --global user.email "nome@email.com"

* Colocando qual o editor está usando:

git config --global core.editor NomeDoEditor

* Para saber as informações cadastradas:

Para nome:

git config user.name

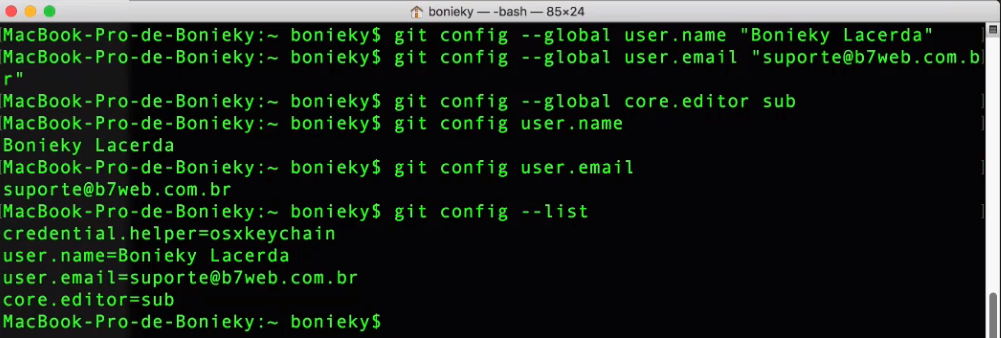
Para email:

git config user.email

Para tudo:

git config --list

* Exemplo:

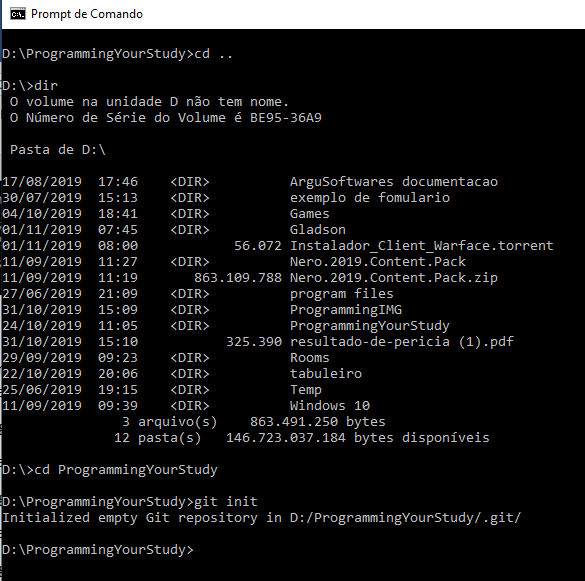


Criando um Repositório GIT

* Coloque o CMD na pasta dos projetos:

Utilizando o comando cd para navegar entre as pastas

* Na pasta que será o repositório git, dê o comando “git init” para inicializar o repositório git
* Exemplo:



Aplicando Alterações “Commit”

* Branch - Branch são versões do sistema, a versão principal é a master.
* README - README.md é um arquivo explicativo para demonstrar em breve com instruções e comentários explicando o que será feito no repositório visitado
* Commit - Commit é aplicar as alterações que foram feitas, é no commit que você coloca uma mensagem explicativa mostrando o que foi alterado.
* Verificando o status dos arquivos do projeto no git:

git status

* Adiciona um arquivo de cada vez:

git add "nome do arquivo"

* Adiciona todos os arquivos para ser commitado:

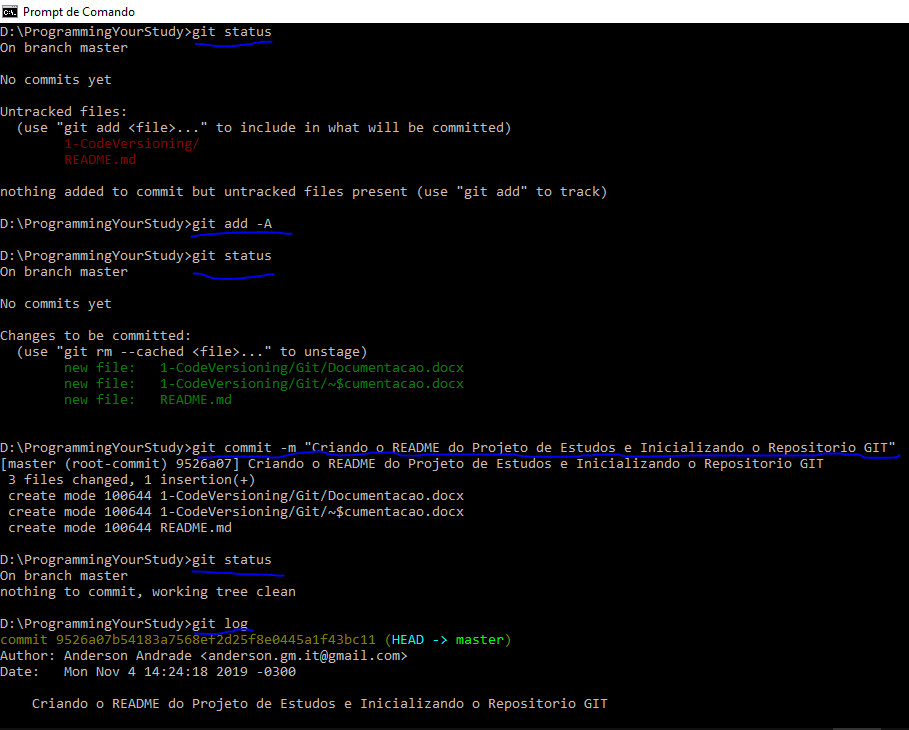
git add -A

* Commitando:

git commit -m "Mensagem do commit"

* Mostra quem fez commit e a mensagem do commit:

git log



Revertendo Modificações

* Lista todos os branches e o (\* identifica o atual):

git branch

* Volta o estado antes do commit:

git reset --soft "id do commit desejado"

* Volta o estado antes do “git add -A”:

git reset --mixed "id do commit desejado"

* Volta o estado para depois do commit anterior ignorando o commit passado:

git reset --hard "id do commit desejado"

* OBS: O ID do commit pode ser visto executando o comando:

git log

Trabalhando com Diferentes Branches

* Criando um novo Branch:

git branch NomeDoBranch

* Movendo para outro Branch:

git checkout NomeDoBranch

* Deletando o Branch:

git branch -D NomeDobranch

Verificando Diferença nos Arquivos

* Ver as alterações feitas dentro dos arquivos:

git diff

* Verifica o nome dos arquivos alterados:

git diff --name-only

* Verifica as alterações do arquivo mencionado:

git diff NomeDoArquivo

* Voltando modificações de apenas um arquivo depois do commit:

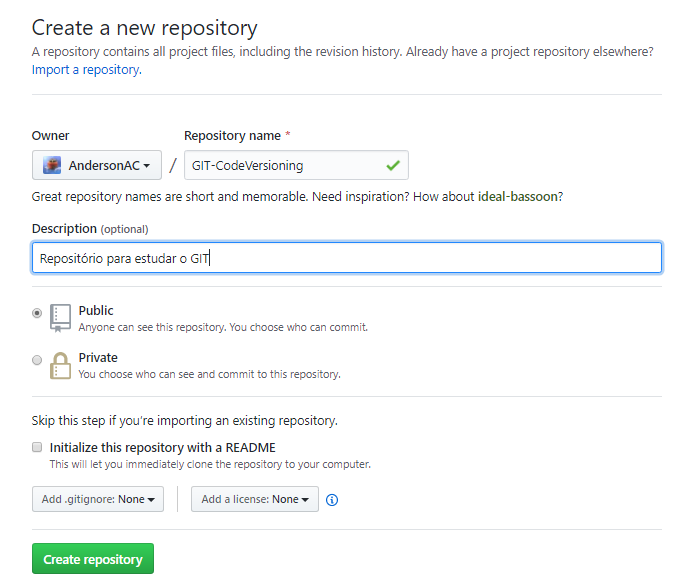
git checkout HEAD -- NomeDoArquivo

* OBS:

HEAD quer dizer o branch atual

Criando um repositório no Github

* Entre no site do <https://github.com/> e crie sua conta
* Crie um novo repositório e descreva:



Vinculando Repositório Local com o Remoto

* Primeiramente deve ser gerado a chave SSH, turorial:

<https://help.github.com/articles/generating-a-new-ssh-key-and-adding-it-to-the-ssh-agent/>

* Passo 1:

Segue o tutorial do link acima;

* Passo 2:

Depois da senha \*\*\*\*

* Passo 3:

Abrir no editor de texto o arquivo id\_rsa.pub que fica localizado no users/.ssh

* Passo 4:

Copie as informações do arquivo

* Passo 5:

No github, clique no usuário->settings

* Passo 6:

No menu SSH and GPG Keys e clica em NEW SSH Keys

* Passo 7:

No campo TITLE coloque uma descrição que descreva de onde vem essa key EX: "DescktopHomeKey"

* Passo 8:

Cole s informações que foram copiadas do arquivo id-rsa.pub na aba key

* passo 9:

Clique em Add SSH Key

Enviando do local para o remoto

* Passo 1:

No repositório lhe indicará o que você deve fazer.

Os códigos são:

git remote add NomeAssociadoPeloRepositorioRemoto URLDoRepositorioRemoto

Exemplo:

git remote add origin https://github.com/AndersonAC/modulogit.git

* Passo 2:

git remote -> Para ver os repositórios remotos

* Passo 3:

git remote -v -> Para mais detalhes

* Passo 4:

git push -u NomeAssociadoPeloRepositorioRemoto NomeDoBranch

Ex:

git push -u origin master

* Passo 5:

Colocar login e senha

* OBS:

fetch -Comando para trazer do remoto para o local

push -Comando para enviar do local para o remoto

Atualizando Repositório Remoto

* Somente para arquivos alterados pois é um commit Local

git commit -am "Mensagem"

* Para novos arquivos deve ser realizado o comando:

git add -A

* Repositório remoto atualizado

git push origin master

Ignorando Arquivos com o Gitignore

* Cria um arquivo com o nome ".gitignore"
* Dentro do arquivo vc coloca o que quer ignorar

exemplo:

backup\_DB.sql

senhas.txt

\*.txt

ou pastas como o:

img/

* comando extra:

git rm -r --cached <nome\_da\_pasta\_ou\_nome\_do\_arquivo>

Revertendo Código

* A diferença entre o reset e o revert é que o reset exclui o commit e volta as alterações, já o revert, ele cria um novo commit com as alterações do commit de antes do bug.

git revert --no-edit LinhaDoCódigoDoCommitAnteriorDoGitLog

* OBS:

Caso queira analizar o commit danificado, pode ser feito um novo branch do problema e depois feito o git revert

Deletando Branches Locais e Remotos

* Deletar remoto

git push origin :NomeDoBranch

* Deletar local

git branch -D NomeDoBranch

Puxando Alterações de Outras Pessoas

* Baixar arquivos remotos:

git pull origin master

* Verificar att remota:

https://git-scm.com/docs/git-diff

* OBS:

<https://stackoverflow.com/questions/10312521/how-to-fetch-all-git-branches>

* Recomendação:

Antes de dar um push dê um pull para juntar primeiro a atualização do remoto com o local e depois enviar as suas alterações do local para o remoto, então primeiramente deve ser dado um pull e depois um push.

Clonando Repositórios Remotos

* Passo 1:

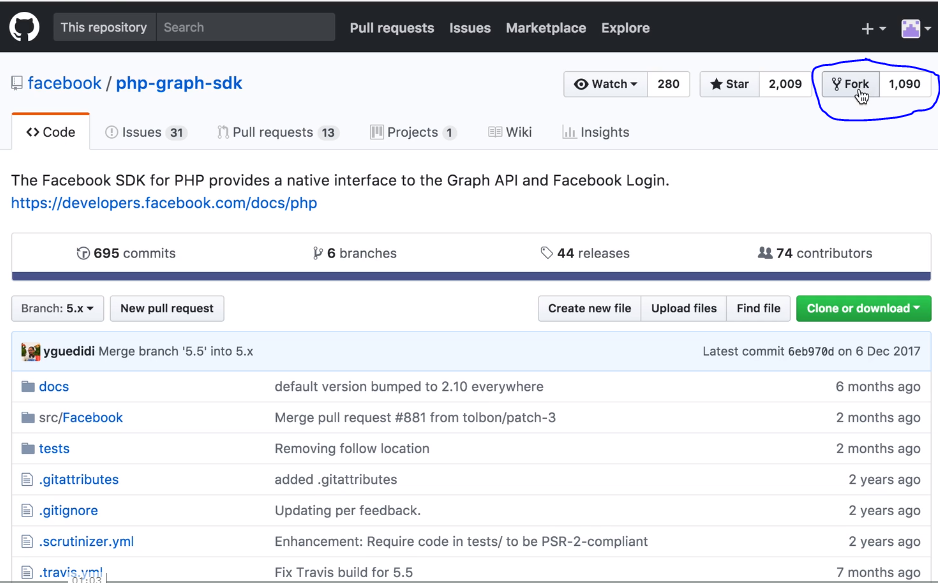
Copia o link do projeto

* Passo 2:

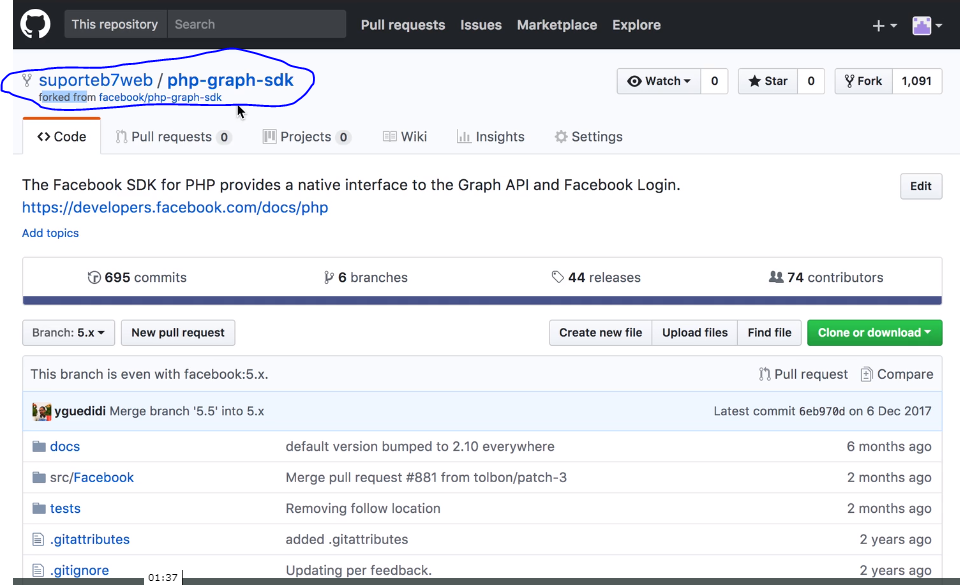
no terminal digite:

git clone URLdoProjeto

Contribuindo com outros repositórios (fork / pull request)



* Depois de dar fork no repositório de um projeto do facebook, podemos ver que foi clonado o repositório e atrelado ao nosso para que possamos contribuir



* Então com o nosso projeto já podemos clonar o repositório para o git local e com isso podemos fazer qualquer alteração
* Depois de fazer qualquer alteração, adicionamos:

git add -A

* Depois um commit

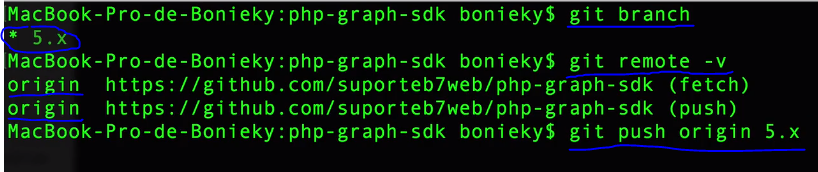
git commit -m “mensagem”

* Agora damos um push, mas primeiramente vemos como foi renomeado o branch remoto:

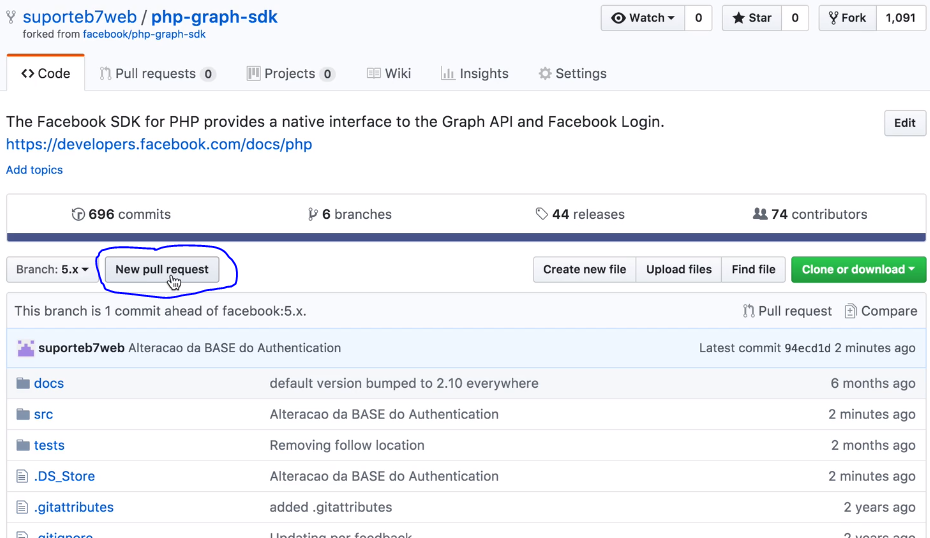
git remote -V

* Por fim aplicamos o push:

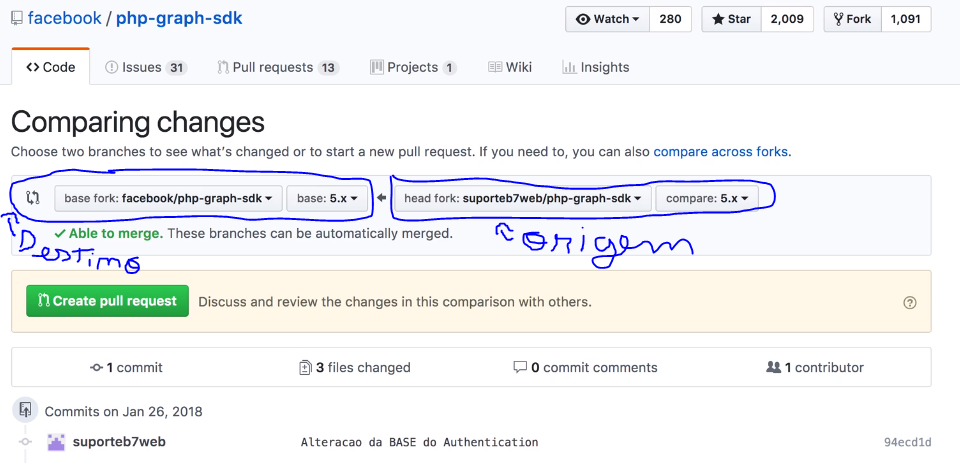
git push origin 5.x



* Pronto agora podemos ver no site do git que as alterações feitas foram aplicadas no nosso projeto, então agora temos que enviar nossa alteração para o diretório original.
* Então damos um new pull request no nosso projeto:



* Depois verificamos a origem e o destino e clicamos em create pull request:



* Depois aparecerá um campo para digitar uma mensagem que será enviada para o dono do repositorio original que irá avaliar as alterações e verificar se aceitara ou não e depois de descrever a alteração é só clicar em create pull request novamente.