INTRODUÇÃO AO GIT E GITHUB

1. GIT

* Git – É um software criado em 2005 por Linus Torvalds, software de versionamento de códigos. GIT e GITHUB são duas tecnologias diferentes.

1. NAVEGAÇÃO VIA COMMAND LINE INTERFACE E INSTALAÇÃO

* Comandos básicos para um bom desempenho no terminal - Aprender os comandos: Mudar de pastas; Listar as pastas; criar pastas/arquivos; deletar pastas/arquivo.
* Comandos – Windows: CD; DIR; MKDIR; DEL/RMDIR. Unix: CD; IS; MKDIR; RM -RF
* Usando o CMD – cd ( changing direction) escolher a pasta; cd ..(volta a pasta), limpar a tela cls, listar diretórios dir, criar pastas mkdir, criar arquivos echo hello > hello.txt, deletar arquivos se usa o del ( não a pasta), rmdir /s /q para remover a pasta(repositório)

1. ENTENDENDO COMO O GIT FUNCIONA POR DEBAIXO DOS PANOS

* Tópicos fundamentais para entender o funcionamento do git – SHA1; OBJETOS FUNDAMENTAIS; SISTEMA DISTRIBUIDO; SEGURANÇA.
* Sha1 – É um algoritmo de encriptação, a encriptação gera um conjunto de caracteres identificador de 40 dígitos. (Segurança)
* Objetos internos do GIT – bloobs, trees e commits.
* Bloobs – sha1 do arquivo dos metadados (é um objeto)
* Trees – armazena bloobs
* Commit – objeto mais importante de todos, ele aponta para as trees, para os bloobs, para o autor, mensagem etc... (commit é único para cada autor)
* Git é um sistema distribuído e seguro.
* Chave ssh – É uma forma de estabelecer uma conexão segura e encriptada entre duas máquinas, se conectar com o servidor do github e configurar a nossa máquina para se conectar com o github. E sempre quando formos enviar códigos a nossa máquina já vai estar configurada.
* Token(comandos) para o Git hub - $ eval $(ssh-agent -s)
* Tokens de acesso pessoal – no próprio site é possível geral um token, geralmente utilizado quando irá trabalhar em uma máquina que não te pertence

1. PRIMEIROS COMANDOS COM GIT

* Iniciar o GIT – git init; git add; git commit
* Commandos – ls -a (mostra arquivos ocultos), Configurar o email : git config --global user.email "mafonso.tec@gmail.com", configurar autor: git config –globar user.name.
* Adicionando um arquivo – obs: Markdown é uma forma humanizada de escrever em html, sem necessariamente utilizar os comandos html.
* Adicionando arquivo no git – Após realizar os trabalhos e for mandar para o git, utilizar git add \*, git commit -m “adicionar o texto que quiser (nome do arquivo)”

1. CICLO DE VIDA DOS ARQUIVOS GIT

* Passo a passo no clico da vida – git init ela inicializa para criar o repositório git, e inicializa um repositório git.
* Tracked – subdividi-se: unmodified: arquivo que n foi modificado; modified: já foi modificado; e staged: conceito chave EX: No palco vai ocorrer um show, atrás do palco é o backstage onde ocorre toda a preparação para fazer parte do show (de outro agrupamento), ou seja é tudo aquilo que esta pronto para ser enviado ou utilizado, são os arquivos que estão no estágio já prontos.
* O comando git add \* moveu o arquivo que foi criado para o staged, como se estivesse no modo de espera.
* O que os repositórios significam - Existem dois ambientes, o de desenvolvimento e o servidor. Desenvolvimento: Working Directory; Staging Area; Local Repository ( Seu repositório local tem que ser commits, pois so consegue enviar esse arquivo para o repositório remoto se esses forem commitados.
* Mais comandos : git status é usado para saber em que estagio esta o arquivo, untracked, tracked, etc... Comando para mover o arquivo para outra pasta mv strogonoff.md ./receitas/
* Git add \* - adiciona toda e qualquer modificação feita para o stage, ou seja para um local onde o git o reconhece e que esta pronto para ser modificado. Pode se usar git add “nomedoarquivo” ou git add.
* O git commit -m “” envelopa toda essa mudança para de fato um repositório local, lhe dando informação, no campo “”

1. INTRODUÇÃO AO GITHUB

* git config –global unset – utilizado para apagar uma configuração(modificar), podendo ser user.email, user.name etc..
* git remote add origin – tudo que agente fez no nosso repositório local, agente adiciona a origem do arquivo
* git remote -v – mostra a lista de repositórios remotos cadastrados.
* git push origin master – comando para empurrar de fato o repositório local (commit local) para o remoto.

1. RESOLVENDO CONFLITOS

* Como resolver conflitos – Git pull origin master: usado para puxar do repositório remoto e fazer devidas alterações necessários, para depois empurra-los novamente
* Clonar repositórios remotos de outros usuários e mandar para sua máquina (repositório local) – git clone “e adiciona a URL em https”
* ls -a – mostrar arquivos ocultos