



git



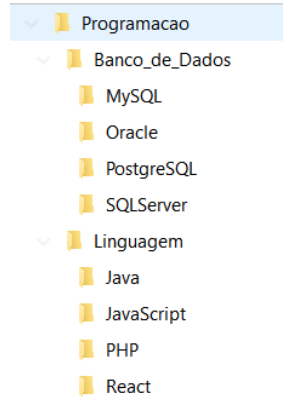
GitHub

Tópicos Essenciais à Programação

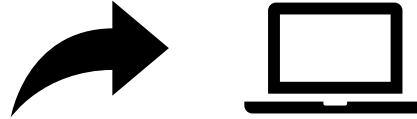
Git e GitHub

- O **Git** é o sistema de controle de versão e o **GitHub** é uma plataforma baseada na web que utiliza o Git para hospedar, colaborar e compartilhar projetos de desenvolvimento de software.
 - Git é um sistema de controle de versão distribuído amplamente utilizado no desenvolvimento de software. Ele permite que você acompanhe e gerencie as alterações feitas em arquivos ao longo do tempo, facilitando o trabalho colaborativo em projetos.
 - GitHub é uma plataforma de hospedagem de código-fonte baseada na web que utiliza o Git como sistema de controle de versão. Ela fornece um local centralizado para armazenar, colaborar e compartilhar projetos Git.

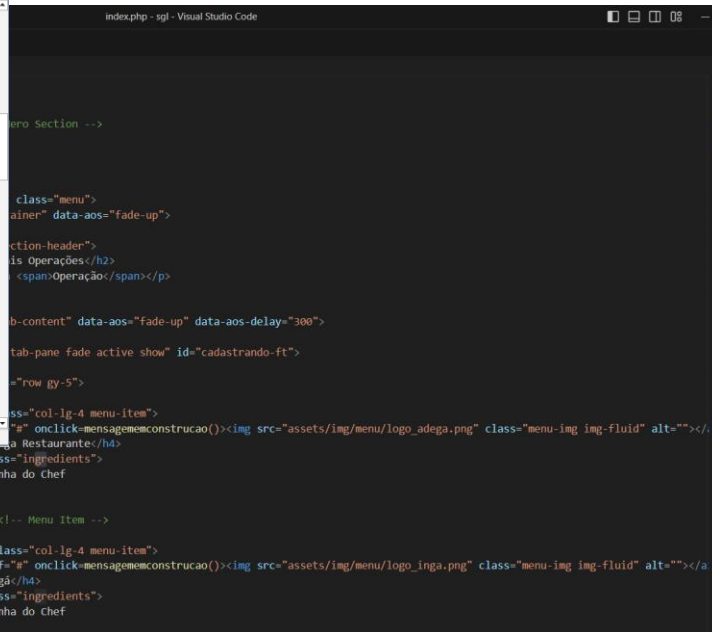
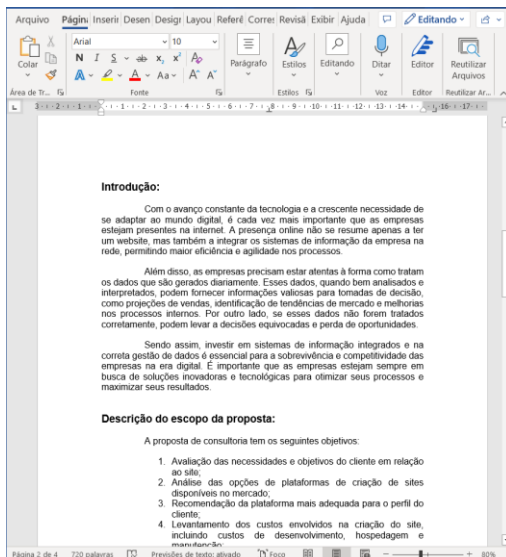
Você e o
seu sistema operacional
(Conjunto de Arquivos)



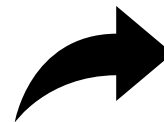
Repositório local



Arquivos



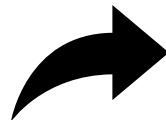
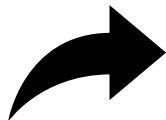
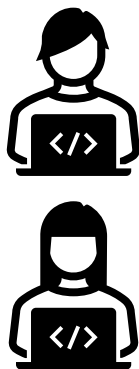
Repositório local



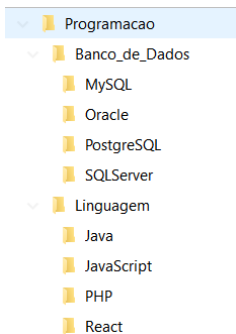
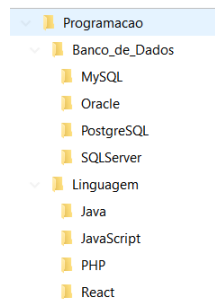


Repositório local

Repositório remoto



Trabalhando com equipes



Repositório local



Repositório local



Repositório local



Repositório remoto



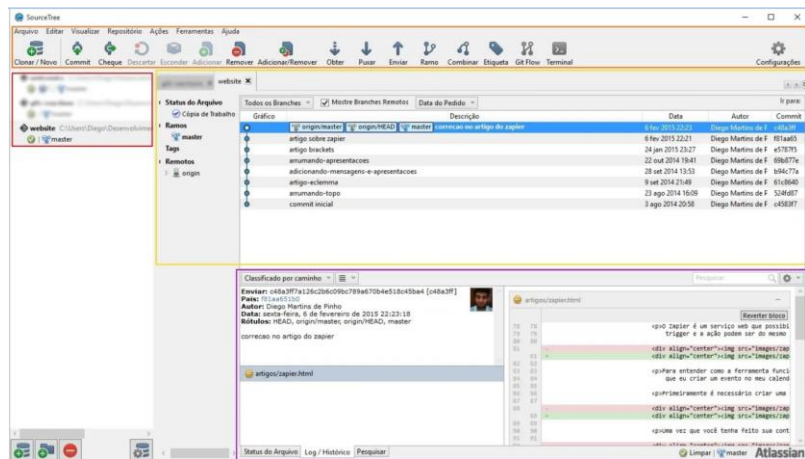
git

Como trabalhar com o Git e o GitHub?

Shell

```
MINGW64~/Users/nisst/Documents/projetogit
$ ls
arquit01.txt  arquito2.txt
nisst@LAPTOP-B0A00460 MINGW64 ~/Documents/projetogit
$
```

Interface Gráfica



Interface do GitHub

nisston / disciplinasistemainformacao (Public)

<> Code Issues Pull requests Actions Projects Wiki Security Insights Settings

main 1 branch 0 tags Go to file Add file <> Code About

nisston	Minha terceira versao	649b43b 49 minutes ago	4 commits
LICENSE	Initial commit	1 hour ago	
README.md	Initial commit	1 hour ago	
arquivo1.txt	Minha terceira versao	49 minutes ago	
arquivo2.txt	Minha segunda versao	1 hour ago	

README.md

disciplinasistemainformacao

Repositório de material para as disciplinas da Uniesp.

Readme MIT license 0 stars 2 watching 0 forks

Releases

No releases published
[Create a new release](#)

Packages

No packages published
[Publish your first package](#)

Comandos GIT {



```
< git init -> Iniciar o Git
git clone -> Cópia a o repositório a ser trabalhado
git pull origin
git add . ou nome do diretório do arquivo --> Adiciona um
arquivo novo ou alterado
git rm . ou nome do diretório do arquivo --> Remove um
arquivo novo ou alterado
git commit -m "texto que vai aparecer nas obs" --> conjunto
de mudanças permanentes
git branch -M main -> area de trabalho
git remote add origin https://github.com/urldorepositorio -
git push -u origin main --> "Empurra" os arquivos para o
repositório
git status -> verifica o status atual da fila
>
```



Vamos praticar!

- Clonando um repositório criado no GitHub
- Recuperando uma versão anterior de um arquivo

Clonando um repositório

- Comandos Básicos:
 - `git clone [url]`
 - `git add .`
 - `git commit -m "Descricao..."`
 - `git push`

Obs.: O comando abaixo irá verificar o status dos arquivos alterados:

- `git status`

Restaurando um arquivo anterior

- Comandos Básicos:
 - `git log`
 - `git checkout [hash] -- [nome_do_arquivo]`
 - `git status`
 - `git add .`
 - `git commit -m "Descricao..."`
 - `git push`

Comandos GIT, acesso via chave SSH

< Guia Rápido de Comandos:



1. Instale o SSH.
2. Instale o Git: `$ sudo apt install git -y`
3. Configure o Git (Com suas credenciais do github ou gitlab):
 - a. `$ git config --global user.name "Fulano de Tal"`
 - b. `$ git config --global user.email "meu_email@dominio.com"`
4. Gere uma chave publica/privada:
 - a. `$ ssh-keygen`
5. (Enter para gerar com nome padrão 'id_rsa.pub')
6. (Vai perguntar duas vezes, se gostaria de colocar senha, neste tutorial deixarei em branco, tecle Enter para gerar, mas é sempre bom colocar senha).
7. Copie sua chave pública gerada para o seu painel no gitlab ou hub:
 - a. `cat ~/.ssh/id_rsa.pub`
8. Copie todo conteúdo mostrado após o comando
9. vá para o site gitlab ou github > SSH Keys > Add Key
10. Cole no campo key



Utilizando o Git no dia-a-dia:



```
# troca para branch de prd
```

```
$ git checkout prd
```

```
# atualiza a branch local com o repositório remoto
```

```
$ git pull origin
```

```
# cria uma nova branch com <tipo>_<descricao>
```

```
$ git checkout -b nome_da_nova_branch
```

```
# envia a branch recém criada para o repositório remoto
```

```
$ git push origin nome_da_nova_branch
```



Utilizando o Git no dia-a-dia:



```
}
```

```
# verifica se há arquivos alterados a serem commitados
```

```
$ git status
```

```
# adiciona-os para staging area
```

```
$ git add <nome-do-arquivo>
```

```
# Adiciona a mensagem do commit
```

```
$ git commit -m '<tipo>: Comentário breve da alteração.'
```

```
}
```





uniesp

Centro Universitário