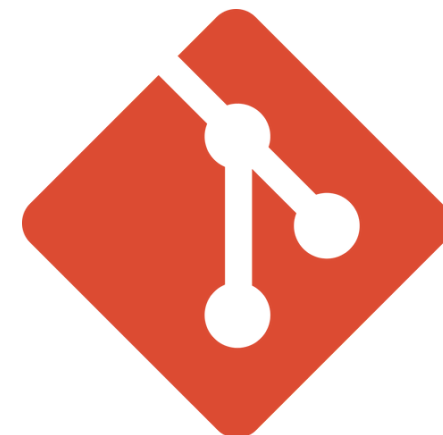


Oficina de Git e Github

Módulo 1: Comandos Básicos



Introdução

Neste módulo, vamos aprender os comandos essenciais do Git para iniciar, registrar, visualizar e versionar arquivos em um projeto.



O que é um repositório?

- Um repositório é onde seu projeto e o histórico de versões são armazenados.
- Cada projeto tem seu próprio repositório.
- Pode ser local (em sua máquina) ou remoto (no GitHub).
- Pode ser clonado por outros desenvolvedores para colaboração.



Criando um repositório

- Comando:
`git init`
- Inicializa um novo repositório Git no diretório atual.
- Cria a pasta oculta .git, onde o histórico será armazenado.
- A partir desse momento, a pasta começa a ser monitorada pelo Git.



Clonando um repositório

- Comando:
`git clone <url-do-repositorio>`
- Baixa (clona) um repositório remoto para sua máquina.
- Ideal para começar a contribuir em um projeto existente.



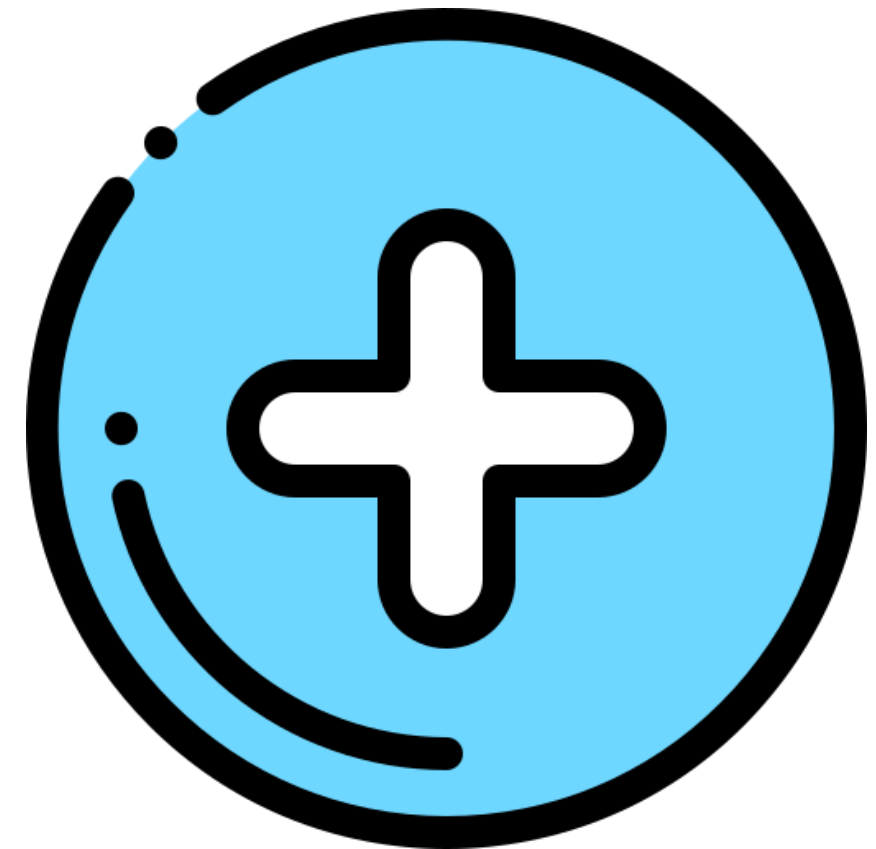
Verificando mudanças no projeto

- Comando:
git status
- Exibe o que foi alterado desde o último commit.
- Mostra arquivos modificados, não monitorados e prontos para commit.



Adicionando arquivos

- Comando:
`git add <nome-do-arquivo>`
`git add .`
- Começa a monitorar os arquivos informados.
- `git add .` adiciona todos os arquivos modificados no diretório atual.



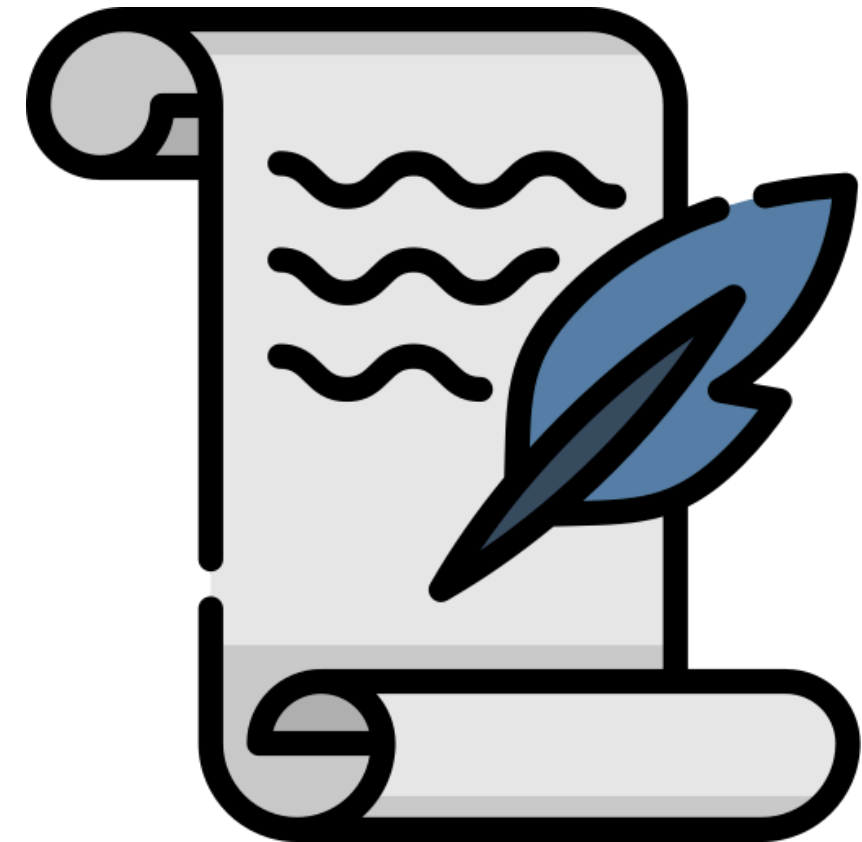
Salvando alterações no repositório

- Comando:
`git commit -a -m "mensagem"`
- Salva as mudanças no histórico de versões.
- Use mensagens claras e objetivas para facilitar a leitura posterior.



Histórico de alterações

- Comando:
git log
- Mostra todos os commits já realizados.
- Inclui autor, data, e hash de identificação de cada commit.



Removendo arquivos

- Comando:
`git rm <nome-do-arquivo>`
- Remove o arquivo do diretório e do controle de versão.
- Precisa de um novo commit para efetivar a remoção.



Renomeando ou movendo arquivos

- Comando:
`git mv <antigo> <novo>`
- Renomeia ou move um arquivo dentro do projeto.
- Git entende essa operação como uma alteração controlada.



Desfazendo alterações em um arquivo

- Comando:
git checkout -- <arquivo>
- Reverte o arquivo ao último commit salvo.
- Útil para descartar mudanças locais ainda não commitadas.



Ignorando arquivos no projeto

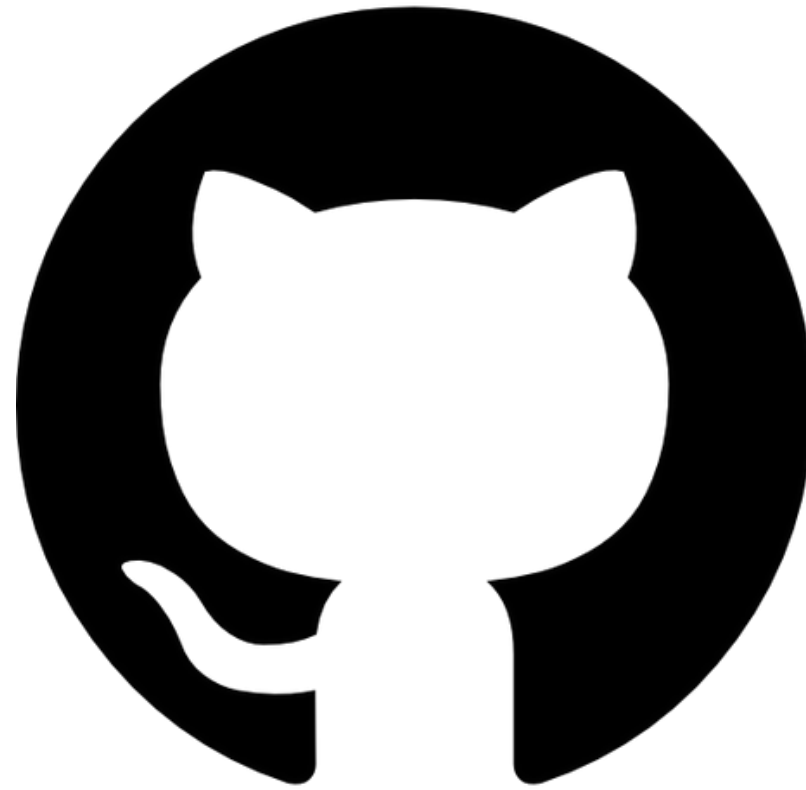
- Arquivo:
 .gitignore
- Adicione arquivos/pastas que você não quer versionar.
- Exemplo: arquivos temporários, configurações locais, .env, node_modules/, etc.



Conclusão do Módulo

- Agora você domina os comandos essenciais para trabalhar com repositórios Git!
- Pratique os comandos no terminal com um projeto próprio ou de exemplo.





No próximo módulo:

Repositórios Remotos

