

Fluxo de Trabalho do Git

1. Inicialização do Repositório
 - Clone o repositório remoto (caso já exista).
 - Configure suas credenciais Git.
2. Criação de uma Nova Branch
 - Trabalhe sempre em branches específicas para cada tarefa/feature.
3. Realização de Alterações
 - Faça commits pequenos e descritivos.
4. Sincronização com o Repositório Remoto
 - Puxe alterações da branch principal antes de realizar um merge.
5. Merge e Pull Request
 - Realize o merge via Pull Request para manter o histórico organizado.
6. Deploy (se aplicável)
 - Após aprovação, atualize o ambiente de produção.

Lista de Comandos Git

1. Configuração Inicial

Configure o Git com seu nome e email:

- `git config --global user.name "Seu Nome"`
- `git config --global user.email "seuemail@exemplo.com"`

Inicialize o repositório local (se necessário):

- `git init`

Clone o repositório remoto:

- `git clone https://github.com/edmilsonpmfilho/RenovacaoAPI`

2. Gerenciamento de Branches

Liste todas as branches:

- `git branch`

Crie e mude para uma nova branch:

- `git checkout -b nome-da-branch`

Troque para uma branch existente:

- `git checkout nome-da-branch`

Delete uma branch (após merge):

- `git branch -d nome-da-branch`

3. Adicionando e Comitando Alterações

Verifique o status do repositório:

- `git status`

Adicione arquivos ao estágio:

- `git add <arquivo>`

ou todos os arquivos

- `git add .`

Faça um commit com uma mensagem:

- `git commit -m "Descrição das alterações"`

4. Sincronização com o Repositório Remoto

Puxe alterações da branch remota:

- `git pull origin nome-da-branch`

Envie alterações para o repositório remoto:

- `git push origin nome-da-branch`

5. Resolvendo Conflitos

Ao encontrar um conflito, edite os arquivos conflitantes conforme necessário.

Marque como resolvido:

- `git add <arquivo-conflito>`

Finalize o commit de merge:

- `git commit`

6. Revisão e Pull Request

Visualize o histórico de commits:

- `git log --oneline`

Crie um Pull Request no repositório remoto (via interface do GitHub/GitLab).

7. Limpeza e Organização

Limpe branches não utilizadas localmente:

- `git branch -d nome-da-branch`

Sincronize branches remotas:

- `git fetch -p`