**Usando o Git**

Versionando com o GIT

OBS: Existe um material na biblioteca do meu SENAI que te ajudará a realizar esta atividade.

# 1. INSTALAÇÃO DO GIT

Para Windows:

• Acesse: https://git-scm.com/download/win

• Execute o instalador e aceite as opções padrão.

Para Linux (Ubuntu/Debian):

• Execute: sudo apt update && sudo apt install git

Para macOS:

• Execute: brew install git

# 2. CONFIGURAÇÃO INICIAL DO USUÁRIO

git config --global user.name "Seu Nome"

git config --global user.email "seuemail@exemplo.com"

Verificar configurações: git config --list

# 3. CRIAR OU CLONAR UM REPOSITÓRIO

A) Criar repositório local:

mkdir nome-do-projeto

cd nome-do-projeto

git init

B) Clonar um repositório remoto:

git clone https://github.com/empresa/projeto.git

cd projeto

# 4. PRIMEIROS COMANDOS ESSENCIAIS

git status

git add .

git commit -m "Mensagem do commit"

git push origin main

# 5. CONFIGURANDO O REPOSITÓRIO REMOTO

git remote add origin https://github.com/empresa/projeto.git

git branch -M main

git push -u origin main

# 6. FLUXO DE TRABALHO COLABORATIVO

Modelo de branches:

• main

• feature/nome-da-feature

• bugfix/nome-do-bug

Criar branch: git checkout -b feature/login

Enviar branch: git push -u origin feature/login

Mesclar com main:

git checkout main

git pull origin main

git merge feature/login

git push origin main

# 7. BOAS PRÁTICAS DE GIT PARA EQUIPE

• Criar branches por tarefa

• Usar Pull Requests para revisão

• Mensagens de commit claras

• Sempre fazer git pull antes de começar

• Evitar commits diretos na main

# 8. FERRAMENTAS ÚTEIS

• GitHub/GitLab/Bitbucket

• VS Code com Git

• Git Kraken ou Sourcetree

• Husky + Lint-Staged

# 9. EXEMPLO DE FLUXO COMPLETO (Desenvolvedor)

git pull origin main

git checkout -b feature/cadastro

... (editar arquivos) ...

git add .

git commit -m "Implementa tela de cadastro"

git push origin feature/cadastro

# 10. PASSO A PASSO PARA CONFIGURAR EM TODAS AS MÁQUINAS

1. Instalar o Git

2. Configurar nome e email

3. Clonar o repositório do projeto

4. Criar sua branch por tarefa

5. Usar git add, commit, push para enviar alterações

6. Criar Pull Request no repositório remoto

7. Aguardar revisão

8. Iniciar nova tarefa em outra branch