# Exercício Avaliativo – Projeto de Login com Git e GitHub

## Objetivo

Desenvolver uma página de login com tema escuro, com validação de campos via JavaScript, aplicar controle de versão com Git, utilizar branches específicas e hospedar o site no GitHub Pages.

## Etapa 1 – Configuração Inicial

1. Instalar e configurar o Git:  
 git --version  
 git config --global user.name "SeuNome"  
 git config --global user.email "seuemail@etec.sp.gov.br"  
  
2. Criar pasta do projeto:  
 mkdir login-projeto  
 cd login-projeto  
  
3. Iniciar o repositório Git:  
 git init

## Etapa 2 – Criar os Arquivos do Projeto

Crie os arquivos: index.html, style.css e script.js com o conteúdo conforme especificado no exercício.

## Etapa 3 – Versionamento com Git e Branches

1. Adicionar arquivos e primeiro commit:  
 git add .  
 git commit -m "Estrutura inicial do projeto"  
  
2. Criar repositório no GitHub e conectar:  
 git remote add origin https://github.com/seuusuario/login-projeto.git  
 git branch -M main  
 git push -u origin main  
  
3. Criar branch develop:  
 git checkout -b develop  
 git push -u origin develop  
  
4. Criar branch feature/validacao:  
 git checkout -b feature/validacao develop  
 # editar script.js com a validação  
 git add script.js  
 git commit -m "Validação de campos implementada"  
 git push -u origin feature/validacao  
 git checkout develop  
 git merge feature/validacao  
 git push origin develop  
  
5. Criar branch feature/dashboard:  
 git checkout -b feature/dashboard develop  
 # criar dashboard.html com menu e boas-vindas  
 git add dashboard.html  
 git commit -m "Criação do painel principal"  
 git push -u origin feature/dashboard  
 git checkout develop  
 git merge feature/dashboard  
 git push origin develop  
  
6. Criar branch feature/cadastro-usuario:  
 git checkout -b feature/cadastro-usuario develop  
 # criar cadastro.html com formulário  
 git add cadastro.html  
 git commit -m "Formulário de cadastro de usuários"  
 git push -u origin feature/cadastro-usuario  
 git checkout develop  
 git merge feature/cadastro-usuario  
 git push origin develop  
  
7. Criar branch hotfix/erro-html:  
 git checkout -b hotfix/erro-html main  
 # corrigir botão mal fechado no index.html  
 git add index.html  
 git commit -m "Correção de estrutura HTML"  
 git push -u origin hotfix/erro-html  
 git checkout main  
 git merge hotfix/erro-html  
 git push origin main  
  
8. Criar branch release/v1.1.0:  
 git checkout -b release/v1.1.0 develop  
 # revisar código, ajustes visuais, criar README.md  
 git add .  
 git commit -m "Preparação para versão 1.1.0"  
 git push -u origin release/v1.1.0  
 git checkout main  
 git merge release/v1.1.0  
 git tag -a v1.1.0 -m "Lançamento versão 1.1.0"  
 git push origin main --tags

## Etapa 4 – Hospedagem no GitHub Pages

1. Acesse o repositório no GitHub  
2. Vá em Settings > Pages  
3. Em Source, selecione:  
 - Branch: main  
 - Pasta: /root  
4. Clique em Save  
5. O GitHub gerará um link como:  
 https://seuusuario.github.io/login-projeto/

## Etapa 5 – Entrega Final

Você deve entregar:  
- Link do repositório no GitHub  
- Link do site hospedado  
- PDF ou DOC com todos os comandos utilizados  
- Responder o questionário: https://forms.office.com/r/He5qzQ3ESX

git --version

git config --global user.name "SeuNome"

git config --global user.email "seuemail@etec.sp.gov.br"

mkdir login-projeto

cd login-projeto

git init

git add .

git commit -m "Estrutura inicial do projeto"

git remote add origin https://github.com/seuusuario/login-projeto.git

git branch -M main

git push -u origin main

git checkout -b develop

git push -u origin develop

git checkout -b feature/validacao develop

# editar script.js com a validação

git add script.js

git commit -m "Validação de campos implementada"

git push -u origin feature/validacao

git checkout develop

git merge feature/validacao

git push origin develop

git checkout -b feature/dashboard develop

# criar dashboard.html com menu e boas-vindas

git add dashboard.html

git commit -m "Criação do painel principal"

git push -u origin feature/dashboard

git checkout develop

git merge feature/dashboard

git push origin develop

git checkout -b feature/cadastro-usuario develop

# criar cadastro.html com formulário

git add cadastro.html

git commit -m "Formulário de cadastro de usuários"

git push -u origin feature/cadastro-usuario

git checkout develop

git merge feature/cadastro-usuario

git push origin develop

git checkout -b hotfix/erro-html main

# corrigir botão mal fechado no index.html

git add index.html

git commit -m "Correção de estrutura HTML"

git push -u origin hotfix/erro-html

git checkout main

git merge hotfix/erro-html

git push origin main

git checkout -b release/v1.1.0 develop

# revisar código, ajustes visuais, criar README.md

git add .

git commit -m "Preparação para versão 1.1.0"

git push -u origin release/v1.1.0

git checkout main

git merge release/v1.1.0

git tag -a v1.1.0 -m "Lançamento versão 1.1.0"

git push origin main --tags