

Título do Trabalho: Introdução Prática ao Git e GitHub: Versionando e Colaborando em Projetos

Objetivo: Este trabalho tem como objetivo aplicar os conhecimentos teóricos sobre Git e GitHub na prática, demonstrando a compreensão dos comandos básicos de versionamento e simulando um fluxo de trabalho colaborativo. Ao final deste trabalho, espera-se que vocês sejam capazes de:

- Inicializar um repositório Git local.
- Adicionar e commitar arquivos, registrando o histórico de alterações.
- Criar um repositório remoto no GitHub.
- Conectar um repositório local a um remoto.
- Enviar (push) o projeto para o GitHub.
- Criar e fazer merge de branches para isolar funcionalidades ou correções.
- (Opcional) Colaborar com um colega utilizando o fluxo de Pull Requests no GitHub.

Instruções:

1. **Escolha do Projeto:** Seleccionem um projeto simples para versionar. Pode ser um site estático (HTML, CSS, JavaScript básico), um script em alguma linguagem de programação (Python, Java, etc.) que realize uma tarefa simples, um conjunto de arquivos de texto organizados, ou qualquer outro tipo de projeto que permita a criação e modificação de arquivos.

2. Inicialização do Repositório Git Local:

- Naveguem até a pasta do projeto utilizando o terminal ou prompt de comando.
- Inicializem um repositório Git utilizando o comando: `git init`.

3. Adicionando e Commitando:

- Adicionem todos os arquivos do projeto à área de staging: `git add ..`
- Realizem o primeiro commit com uma mensagem clara: `git commit -m "Initial commit: Adicionando arquivos iniciais do projeto"`.
- Façam algumas modificações nos arquivos do projeto.
- Adicionem as novas alterações ao staging: `git add ..`
- Realizem um novo commit descrevendo as modificações: `git commit -m "Descrição das alterações realizadas"`.

4. Criação do Repositório Remoto no GitHub:

- Criem uma conta no GitHub (caso ainda não possuam).
- Criem um novo repositório **público** no GitHub, dando um nome relevante ao projeto. **Não** inicializem o repositório com README, .gitignore ou licença.

5. Conexão Local-Remoto:

- No terminal, dentro da pasta do projeto local, conectem o repositório local ao remoto utilizando o comando (substituindo `<seu_usuario>` pelo seu nome de usuário

no GitHub e `<nome_do_repositorio>` pelo nome do repositório que você criou):

- Bash

```
git remote add origin
```

```
https://github.com/<seu_usuario>/<nome_do_repositorio>.git
```

-
-

6. Envio para o GitHub (Push):

- Enviem os commits do repositório local para o GitHub:
- Bash

```
git push -u origin main
```

-
- (Lembrem-se de usar `master` no lugar de `main` caso essa seja a branch principal do seu repositório).

7. Trabalhando com Branches (Obrigatório):

- Criem uma nova branch para implementar uma nova funcionalidade ou realizar uma correção: `git checkout -b nova-funcionalidade`.
- Realizem as modificações necessárias na nova branch e façam commits descrevendo as alterações.
- Voltem para a branch principal: `git checkout main`.
- Façam o merge da branch de funcionalidade na branch principal: `git merge nova-funcionalidade`.

- Enviem as alterações da branch principal para o GitHub: `git push origin main`.

8. Colaboração (Opcional, mas altamente recomendado):

- Trabalhem em duplas ou pequenos grupos. Um membro do grupo cria o repositório no GitHub e convida os outros como colaboradores.
- Um membro cria uma nova branch, faz alterações e envia um "Pull Request" para a branch principal.
- Outro membro revisa o código no Pull Request, pode adicionar comentários e, finalmente, faz o merge das alterações.

Formato de Entrega:

- Vocês deverão entregar o **link para o repositório público do projeto no GitHub**.
- (Opcional) Um breve relatório (em formato de texto ou PDF) descrevendo os passos que realizaram, os comandos Git que utilizaram e quaisquer desafios que encontraram durante o processo (e como os superaram).

Critérios de Avaliação:

- Presença e organização dos arquivos do projeto no repositório do GitHub.
- Qualidade e clareza das mensagens de commit.
- Demonstração da correta utilização de branches (criação e merge).
- Conexão bem-sucedida do repositório local com o remoto e envio das alterações para o GitHub.

- (Se aplicável) Utilização correta do fluxo de Pull Requests e interação na colaboração.
- (Se solicitado) Clareza e completude do relatório.

Prazo de Entrega: [Inserir a data de entrega aqui]

Observações:

- Não hesitem em consultar a documentação do Git e do GitHub, tutoriais online e outros recursos para auxiliar na realização deste trabalho.
- Em caso de dúvidas, procurem o professor durante o horário de aula ou plantão.