

Treinamento Git e GitHub

COEN UFSJ - Setembro de 2023

Cronograma

- 19h às 20h30: introdução, download dos programas necessários, criação de uma conta no Github e criação do primeiro repositório;
- 20h30 às 20h50: *Coffee Break* ☕ - Sala 3.05/3.06;
- 20h50 às 23h (talvez 🤔): configuração do ambiente local, "conectar" o repositório local ao remoto (do Github) e contribuição nesta apresentação.

Motivação (Por que e para que esse treinamento?)

- No contexto da faculdade: utilizá-lo para trabalhos em grupo ou para simplesmente ter um controle das alterações que fazemos em nosso código/arquivo;
- Sobre o treinamento: a ideia principal é dar uma introdução bem inicial sobre o que é versionamento de código e como fazê-lo.
 - Portanto, o objetivo é que vocês saiam com o **conceito** bem definido. A parte do uso é aprimorada ao passar do tempo.

Tá, mas o que é Git e GitHub?

- Git: "é um sistema de controle de versões distribuído, usado principalmente no desenvolvimento de software, mas pode ser usado para registrar o histórico de edições de qualquer tipo de arquivo".
- GitHub: "é uma plataforma de hospedagem de código-fonte e arquivos com controle de versão usando o Git. Ele permite que programadores, utilitários ou qualquer usuário cadastrado na plataforma contribuam em projetos privados e/ou Open Source de qualquer lugar do mundo."

Na prática

- Podemos **versionar** os nossos arquivos (sejam códigos ou não) de maneira mais inteligente: chega de criar várias cópias de um mesmo arquivo
 - versao_final.docx
 - versao_final1.docx
 - versao_final_agora_vai.docx
 - versao_final_FINAL.docx
- Além disso, podemos compartilhá-los com outras pessoas (GitHub)

Analogias sobre a diferença entre o Git e Github

- Git como um Fotógrafo e GitHub como um Álbum Online
- Git como um Livro de Anotações Pessoais e GitHub como uma Biblioteca Pública
- Git como uma Receita de Cozinha e GitHub como um Livro de Receitas Compartilhado

(créditos ao ChatGPT)

Vamos à prática!

Antes de começar, alguns links importantes:

👉 [VSCode - Editor de Textos](#) - Lembrar de instalar a extensão "Marp for VSCode"

👉 [Windows Terminal](#)

👉 [Git](#)

👉 [Lista de Comandos úteis do Git](#) - vamos usar isso em breve

GitHub:

1. Criar uma conta no [GitHub](#)
2. Depois, vamos [criar nosso primeiro repositório](#)

Configuração do Git

1. Fazer o download do [Git](#)

1.1. Vamos seguir o passo a passo de instalação juntos

2. Configurar o Git localmente

3. "Conectar o Git ao GitHub" ([Link](#))

Próximos passos:

- Criar um repositório local;
- Brincar com alguns comandos do Git;
- Conectar o repositório local ao repositório do GitHub;

Criando um repositório:

- Criar uma pasta na Área de Trabalho (ou em outro local);
- Entrar na pasta, clicar com o botão direito do mouse e ir na opção "Abrir no Windows Terminal";
- No Windows Terminal, digitar `code .`
- No VSCode, criar um arquivo .md e escrever algo nele;
- Seguir os comandos mostrados no Github 📝

Os comandos:

👉 Veja esses comandos também no site do Github

```
git init
git add .
git commit -m "first commit"
git branch -M main
git remote add origin <link para o repo aqui>
git push -u origin main
```

Vamos criar uma nova branch:

- Rode o seguinte comando: `git checkout -b adicionar-nova-linha`
 - O "adicionar-nova-linha" é o nome da nossa nova branch;
 - O comando `checkout` é utilizado para navegar entre branches, e a flag `-b` serve para criar e pular para a nova branch que criamos
- Vamos escrever mais uma linha no nosso arquivo;
- Depois, executar os comandos:

```
git add .  
git commit -m "adicionando uma nova linha"  
git push origin HEAD
```

Agora, vamos colaborar com esta apresentação:

- Crie um fork desse repositório no Github;
- Clone na sua máquina: `git clone <endereço do repo aqui>`
- Abra esta apresentação no seu VSCode;
- Crie uma nova branch;
- Depois, abra um Pull Request para esta apresentação

Coloque seu nome abaixo 👉 :

Links Importantes!

👉 [Lista de Comandos úteis do Git](#)

👉 [Documentação do Github](#)

Muito obrigado! 🤝

Se possível, responda a [pesquisa de satisfação!](#)