

# Aula 4 - GIT

Rafaela de Moraes Papale

# O que é Git? E Git Hub?

- ▶ **O Git é um sistema de controle de versão distribuído e amplamente adotado.**
- ▶ Permite manter o histórico de arquivos, acessar o código de outros colaboradores, recuperar o estado de um arquivo anterior, comparar mudanças e encontrar bugs.
- ▶ **GitHub é uma plataforma para gerenciar seu código e criar um ambiente de colaboração entre devs, utilizando o Git como sistema de controle.** Ele vai facilitar o uso do Git, escondendo alguns detalhes mais complicados de setup. É lá que você provavelmente vai ter seu repositório e usar no dia a dia.
- ▶ O Bitbucket e o Gitlab são as duas principais alternativas de serviço Git além do Github.

# Como usar o Git?

- ▶ Você vai usar o Git para guardar o versionamento de todo seu sistema, de maneira segura e distribuída.
- ▶ Deve sempre fazer pequenos commits e 'pushar' suas modificações em grupos que fazem sentido.
- ▶ Do outro lado, deve fazer os 'pulls' para sincronizar com as modificações de todo time.
- ▶ O Git é uma ferramenta, você vai precisar de um servidor principal (mesmo ele sendo distribuído) para facilitar seu trabalho. É aí que entra o Github como sendo uma das opções já prontas.

# O que é um commit?

- ▶ Cada commit registra um momento de vida do seu projeto. É uma foto (snapshot) daquele instante dos arquivos. Com isso você consegue se referenciar àquele momento do projeto, podendo voltar a ele, analisá-lo, ver as diferenças com um outro momento.
- ▶ Os commits são feitos no seu repositório local e você precisa 'pushar' (empurrá-los) de volta para a origem, para que outras pessoas tenham acesso a essas fotos e atualizem-se no tempo de vida.

# Como instalar o Git no computador?

- ▶ Consulte o seguinte site e siga o passo a passo:

<https://dicasdeprogramacao.com.br/como-instalar-o-git-no-windows/>

# Comandos importantes

- ▶ **git init**: Inicializar um novo repositório
- ▶ **git status**: Verificar estado dos arquivos daquele diretório
- ▶ **git add meuArquivo.txt**: Adiciona um arquivo específico.
- ▶ **git add .** : Adiciona todos os arquivos modificados naquele repositório local.
- ▶ **git commit -m “Mensagem de commit”**: Commita os arquivos após adicioná-los.
- ▶ **git push**: Enviar os arquivos para o repositório remoto.
- ▶ **git pull**: Atualiza os arquivos do repositório.
- ▶ **git clone ‘link do repositório remoto’**: Faz um “clone” de um repositório remoto na sua máquina local.