

# Unidade 0 - Nivelamento - Introdução ao GitHub



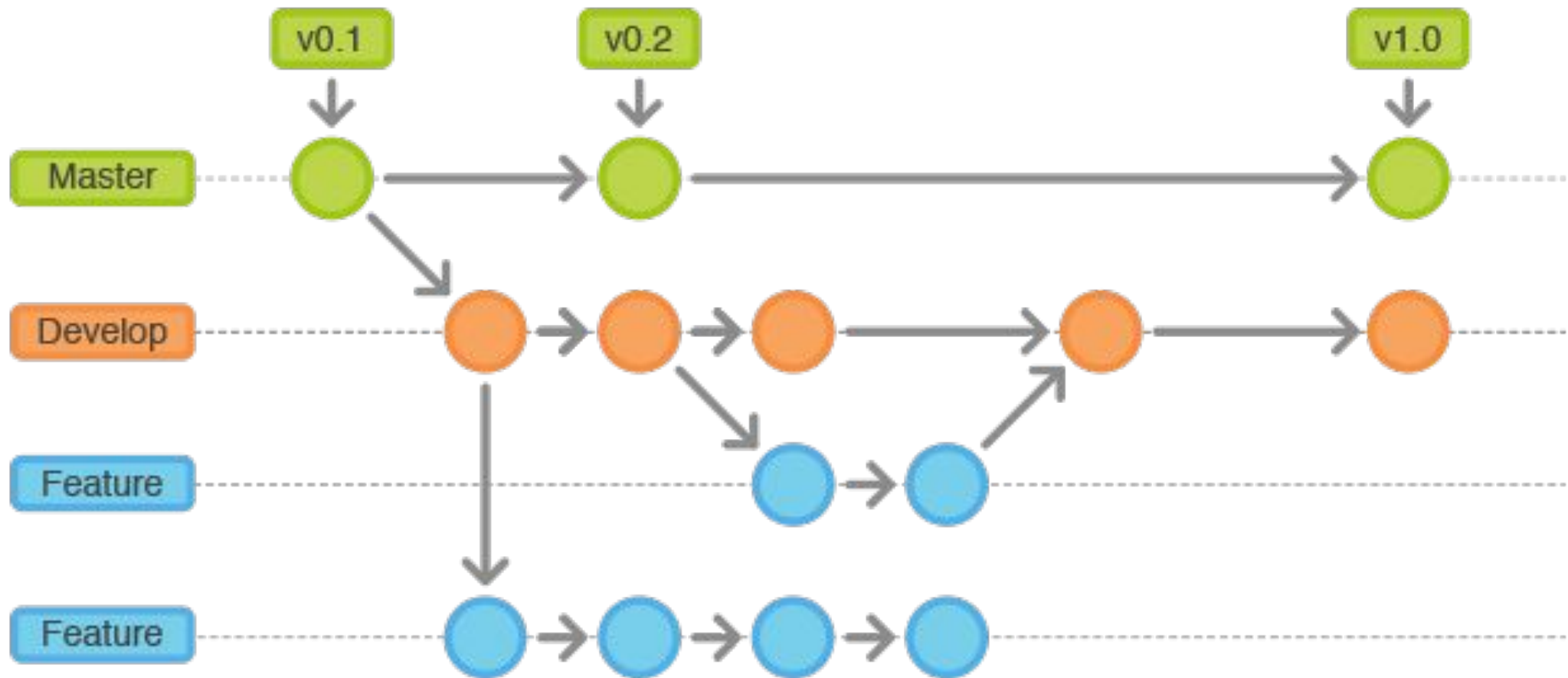
**PUC Minas**

Instituto de Ciências Exatas e Informática  
Departamento de Ciência da Computação

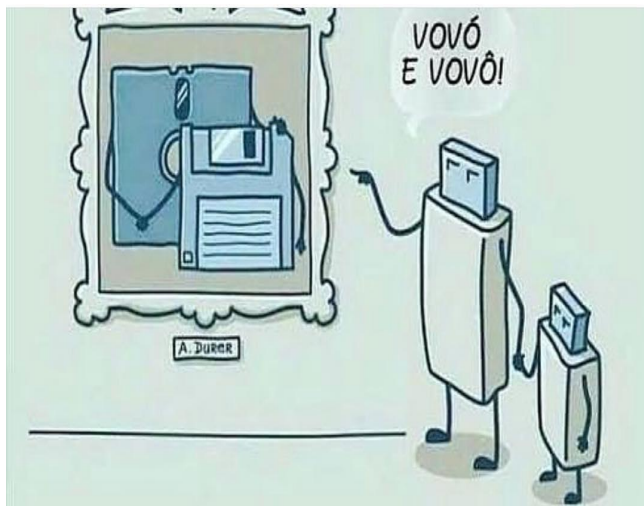
# Motivação: Trabalho em Equipe



# Motivação: Versionamento de Código



# Compartilhamento de Código no Passado



# GIT vs. GitHub

- **GIT**: programa que versiona códigos
- **GitHub**: servidor na nuvem que armazena seus códigos e possui uma interface amigável. Outras opções são o GitLab e BitBucket

# Primeiro Passo: Instalar GIT e Conta no GitHub

- Baixar e instalar o GIT
  - Endereço: [git-scm.com](https://git-scm.com)
  - No Linux, basta usar o `atp-get`

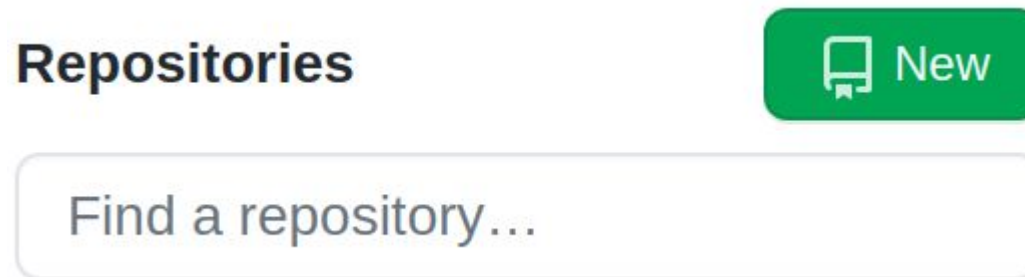


- Criar sua conta no GitHub
  - Endereço: [github.com](https://github.com)

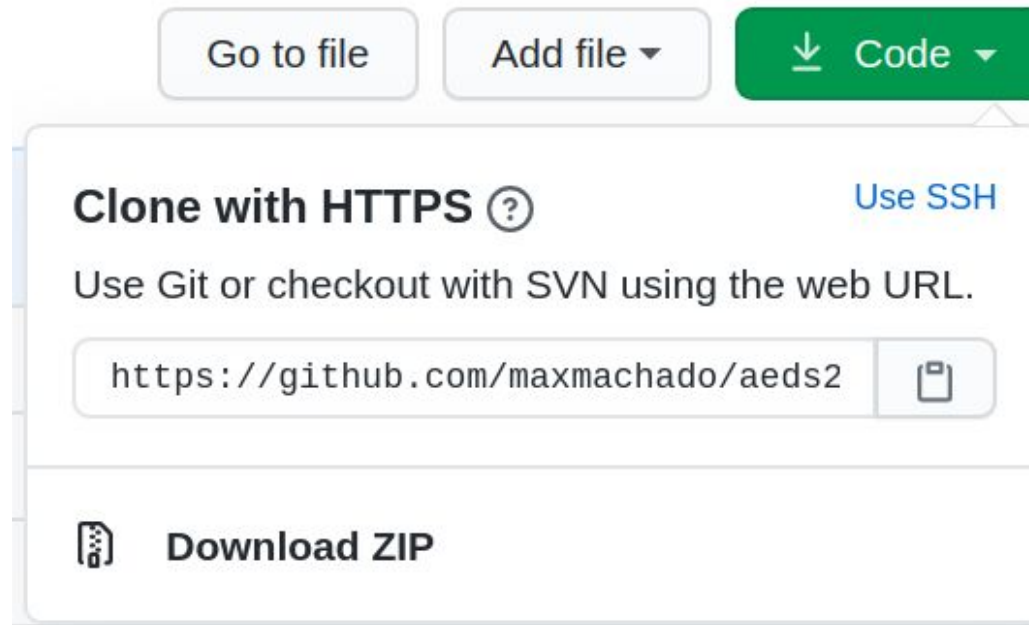


# Segundo Passo: Criar Repositório

- Crie um repositório no GitHub



- Copie o endereço do seu repositório



## Terceiro Passo: No Terminal do Linux

- Entre na pasta onde será seu repositório local
- Execute o comando **git clone *link-do-seu-repositório***
- Copie seus códigos para a pasta local do repositório
- Execute o comando **git add .**
- Execute o comando **git commit -m "importação inicial",**
- Execute o comando **git push**
- No site do GitHub, seus códigos estão atualizados



# Trabalhando em Equipe

- Depois que atualizamos nossos códigos, executamos um **pull request** para efetuar um merge dos códigos através de um *branch*
- **pull request** é um pedido de incorporação do seu código à árvore principal
- Podemos ter vários ramos (*branches*) sendo que o principal é o master
- Não é recomendado fazer *commits* e *pushes* diretamente no branch master

# Trabalhando em Equipe

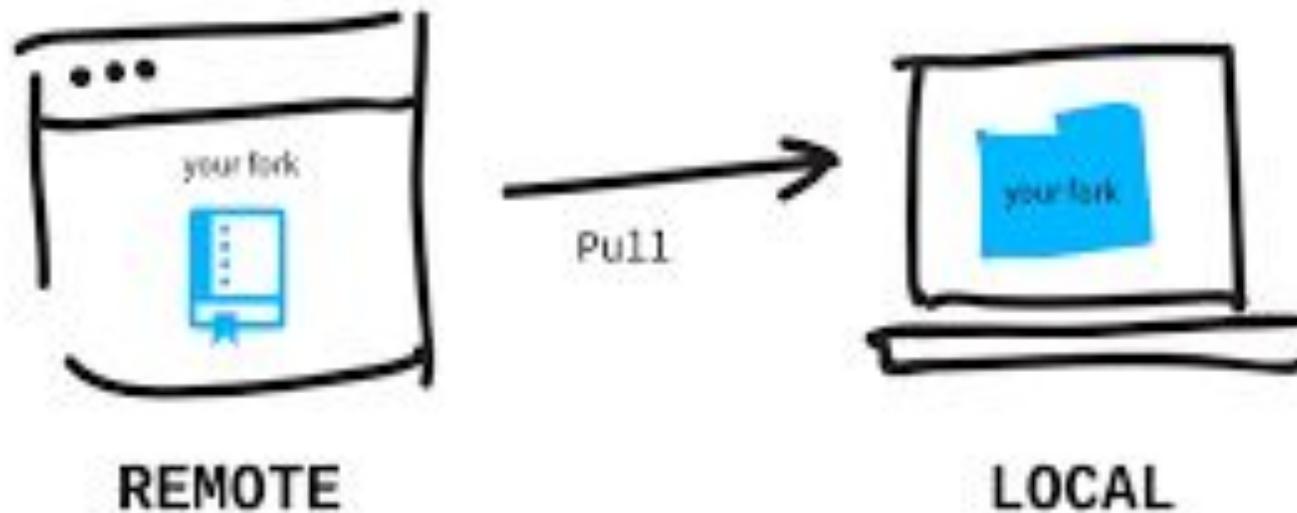
- Após um *pull request*, temos uma pessoa da equipe responsável por aceitar o merge dos códigos
- O responsável efetua o *code review* e faz comentários caso o código demande alguma melhoria. Depois, de corrigido e ajustado, o responsável incorpora o código ao branch master

# Alterando o Código

- Como nosso projeto está na branch master, executamos o comando **git checkout -b nome-do-branch** para criar uma nova branch
- Efetuamos as mudanças desejadas no código
- Executamos os comandos abaixo e, em seguida, aguardamos que o responsável pela equipe faça a code review:
  - **git add .**
  - **git commit -m “descrição-da-alteração”**
  - **git push**
- **Observe atentamente as mensagens retornadas por cada comando!!!**

# Receber Atualizações do Código

- Execute o comando **git pull**



# Gerenciamento de Conflitos

- Antes de modificar um código, baixamos a versão mais atual do mesmo
- Quando modificamos uma versão diferente da mais atual e executamos um git push, o GIT barra tal modificação e informa os conflitos para que sejam verificados pelo programador
- Após a correção dos conflitos, o programador executa novamente os comandos de add, commit (com mensagem adequada) e push