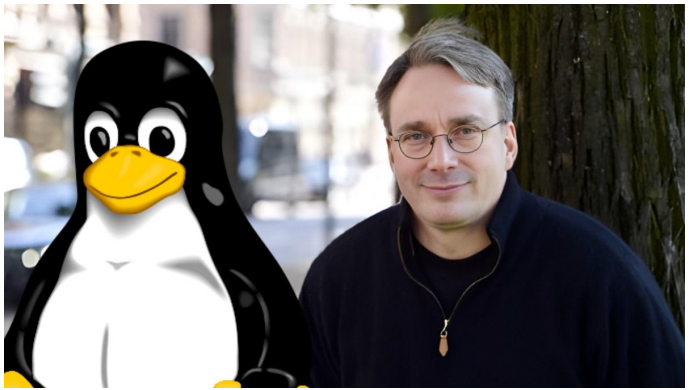

git
GitHub

Introdução ao Git e GitHub

- Controle de versão e colaboração em projetos de software
- Relevância de Git e GitHub no desenvolvimento de software moderno

O que é Git?

- Definição:
  - Git é um sistema de controle de versão distribuído criado por Linus Torvalds em 2005.
  - É amplamente utilizado para gerenciar o histórico de mudanças em projetos de software.



O que é Git?

- Principais Características:
  - **Distribuído**: Cada desenvolvedor possui uma cópia completa do repositório, incluindo todo o histórico de alterações.
  - **Eficiente**: Opera rapidamente mesmo em projetos grandes.
  - **Seguro**: Usa criptografia para assegurar a integridade do histórico de commits.
  - **Flexível**: Suporta múltiplas ramificações (branches) e fusões (merges).

Por que usar Git?

- Motivações:
  - **Controle de Alterações**: Rastreia mudanças ao longo do tempo, permitindo o retorno a versões anteriores.
  - **Colaboração**: Facilita o trabalho simultâneo de várias pessoas no mesmo projeto sem conflitos.

Por que usar Git?

- Motivações:
  - **Branching e Merging:** Permite experimentar novas funcionalidades e integrá-las ao projeto principal quando estiverem prontas.
  - **Backups e Histórico:** Garante que nenhuma alteração seja perdida, armazenando um histórico detalhado de todo o projeto.

Conceitos Básicos do Git

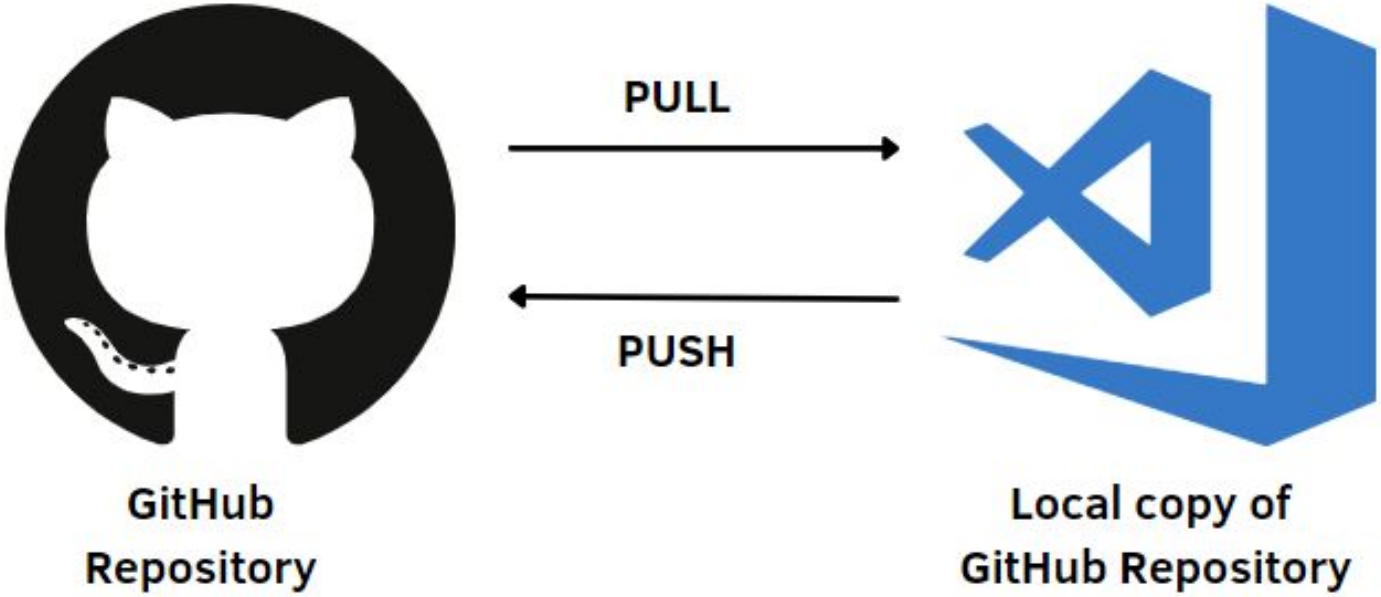
- **Repositório (Repository)**:
  - Um diretório que contém o código do projeto e o histórico de alterações. Pode ser local (no seu computador) ou remoto (em um servidor).
- **Commit**:
  - Um snapshot do seu projeto em um momento específico. Cada commit tem uma mensagem descritiva, um autor, e um identificador único (hash).
- **Branch**:
  - Uma ramificação do projeto onde você pode desenvolver funcionalidades separadamente do código principal. Exemplo: main, develop, feature-x.
- **Merge**:
  - O processo de integrar mudanças de uma branch em outra, combinando o trabalho feito separadamente.

Fluxo de Trabalho com Git

- **Clonar um repositório (git clone)**: Baixa uma cópia completa de um repositório remoto para a máquina local.
- **Criar uma nova branch (git branch e git checkout ou git switch)**: Cria e move-se para uma nova linha de desenvolvimento.
- **Fazer alterações no código**: Editar arquivos, adicionar novas funcionalidades, ou corrigir bugs.

Fluxo de Trabalho com Git

- **Comitar as alterações (git commit):** Salvar as alterações com uma mensagem descritiva.
- **Enviar as alterações para o repositório remoto (git push)**: Atualiza o repositório remoto com os commits feitos localmente.
- **Realizar um merge ou pull request**: Integra as mudanças da branch para a branch principal após revisão.


PULL
PUSH
GitHub
Repository
Local copy of
GitHub Repository

O que é GitHub?

- **Definição**:
  - GitHub é uma plataforma de hospedagem de código-fonte e controle de versão, que utiliza o Git para gerenciar repositórios. Também oferece ferramentas de colaboração e integração contínua.

O que é GitHub?

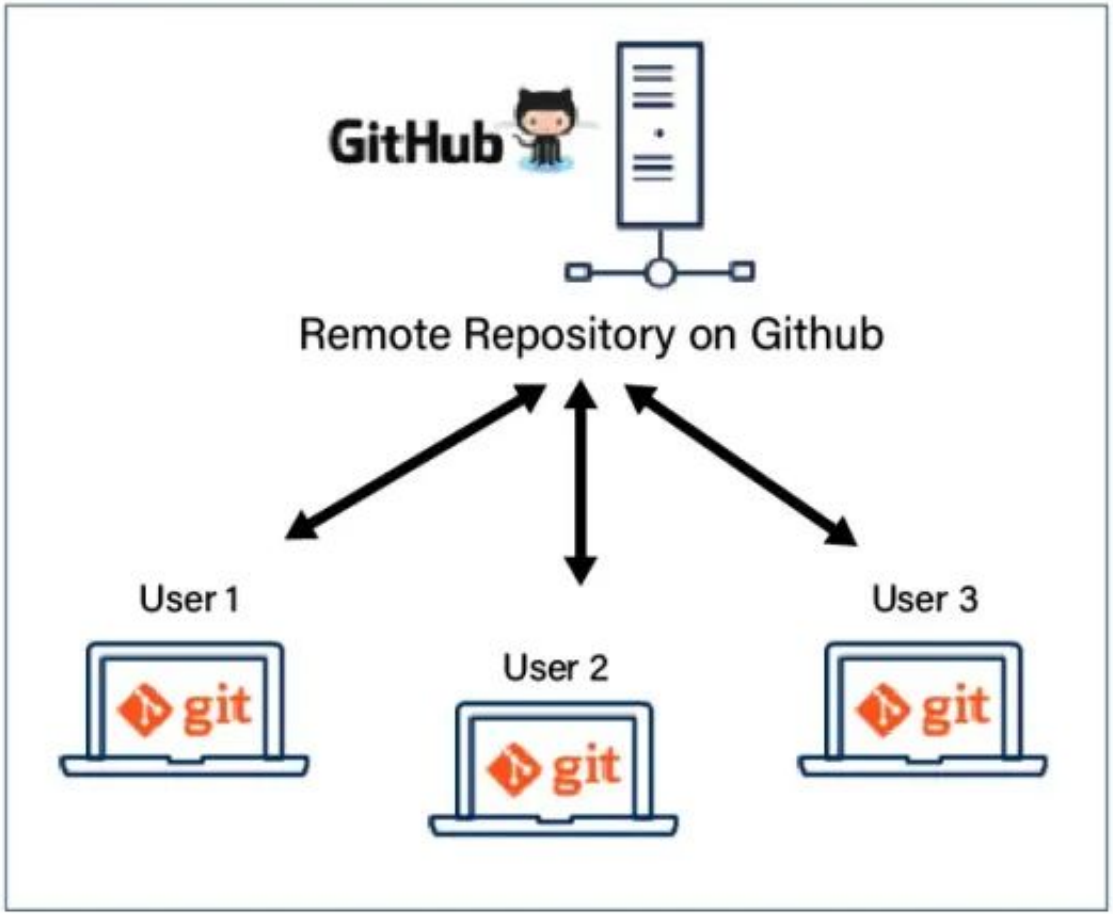
- Principais Características:
  - **Hospedagem de Repositórios**: Armazena repositórios Git na nuvem, acessíveis de qualquer lugar.
  - **Interface Web**: Facilita a navegação e o gerenciamento de repositórios através de uma interface gráfica.
  - **Colaboração**: Ferramentas como pull requests e issues facilitam a colaboração entre desenvolvedores.

GitHub vs. Git

- Diferença Principal:
  - **Git**: Um sistema de controle de versão distribuído utilizado localmente ou em servidores.
  - **GitHub**: Uma plataforma que hospeda repositórios Git e oferece ferramentas adicionais para colaboração, revisão de código, e automação de fluxos de trabalho.

GitHub vs. Git

- Comparação:
  - **Git**:
    - Ferramenta de linha de comando.
    - Usado para gerenciar e versionar código localmente ou em servidores.
  - **GitHub**:
    - Interface web com recursos como pull requests, issues, e GitHub Actions.
    - Integração com outras ferramentas e serviços, oferecendo um ambiente completo para o desenvolvimento de software.


GitHub
Remote Repository on Github
User 1
User 2
User 3
git
git
git

Criptografia de Chave Simétrica

Também chamada de criptografia de chave secreta ou única, utiliza uma mesma chave tanto para codificar como para decodificar informações, sendo usada principalmente para garantir a confidencialidade dos dados.

Criptografia de Chave Simétrica

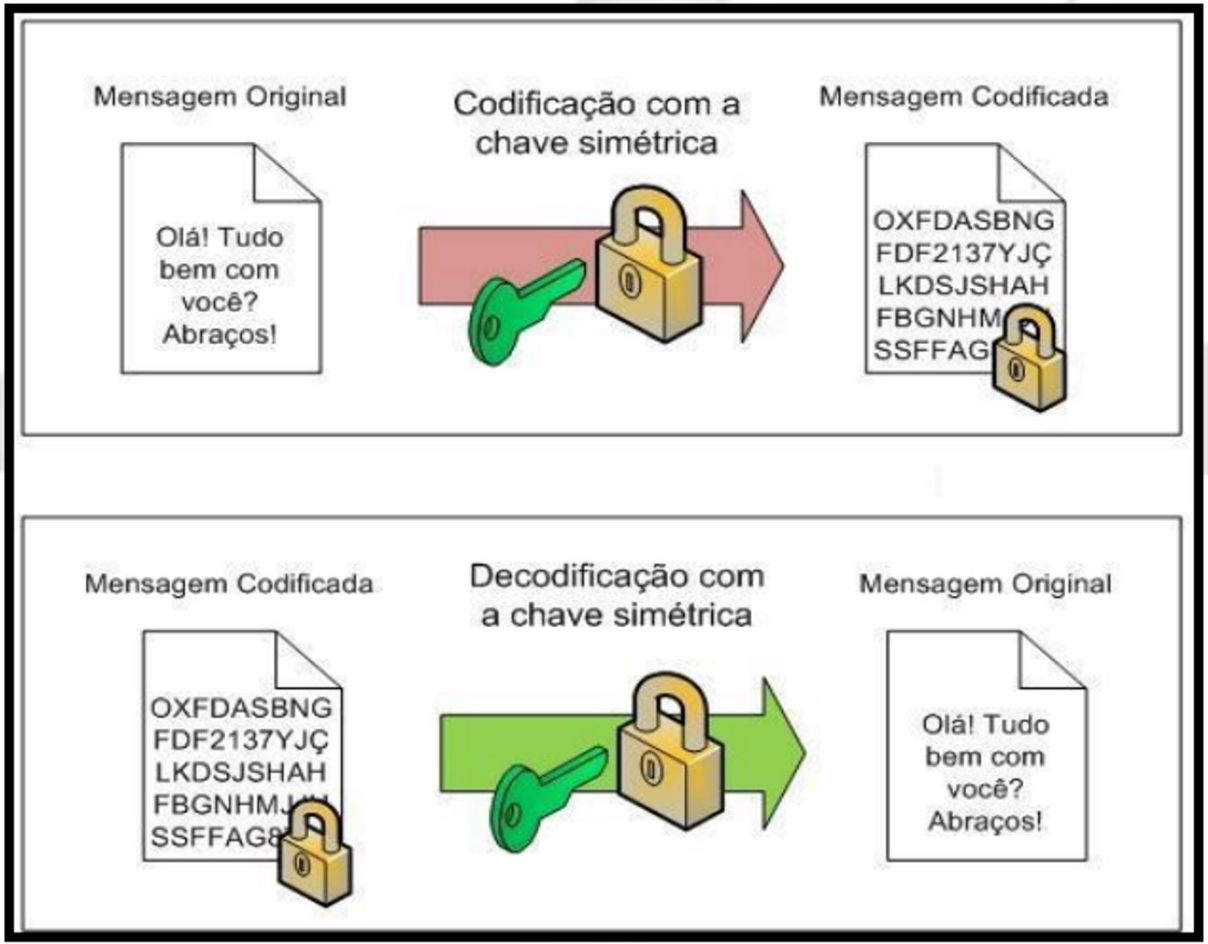
Como exemplo de seu funcionamento, se **Paula** quer enviar uma mensagem secreta para **Tatiana**, ela deve fazer isso:

Mensagem + ChaveSimétrica = MensagemCriptografada

Criptografia de Chave Simétrica

Então **MensagemCriptografada** é enviada para **Tatiana** por uma rede aberta, que para lê-la terá que fazer o seguinte:

MensagemCriptografada + ChaveSimétrica = Mensagem



Simétrica

Fonte: http://www.gta.ufrj.br

Criptografia de chaves Assimétricas

Também conhecida como criptografia de chave pública, utiliza duas chaves distintas: uma **pública**, que pode ser livremente divulgada, e uma **privada**, que deve ser mantida em segredo por seu dono.

Criptografia de chaves Assimétricas

- Quando uma informação é codificada com uma das chaves, somente a outra chave do par pode decodificá-la.
- A chave privada pode ser armazenada de diferentes maneiras, como um arquivo no computador, um smartcard ou um token.

Criptografia de chaves Assimétricas

Essas fórmulas têm a impressionante característica de o que for criptografado com uma chave, só pode ser descriptografado com seu par. Então, no nosso exemplo, **Bob** agora enviaria uma mensagem para **Alice** da seguinte maneira:

Mensagem + ChavePública(Alice) = MensagemCriptografada

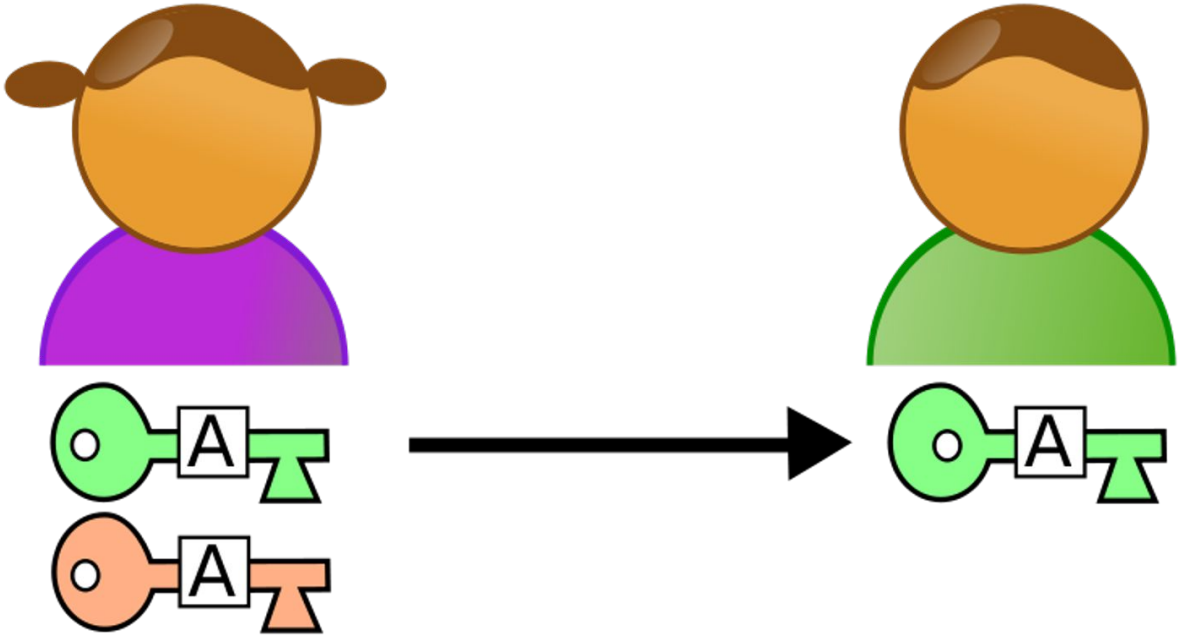
Criptografia de chaves Assimétricas

E **Alice** leria a mensagem assim:

MensagemCriptografada + ChavePrivada(Alice) = Mensagem

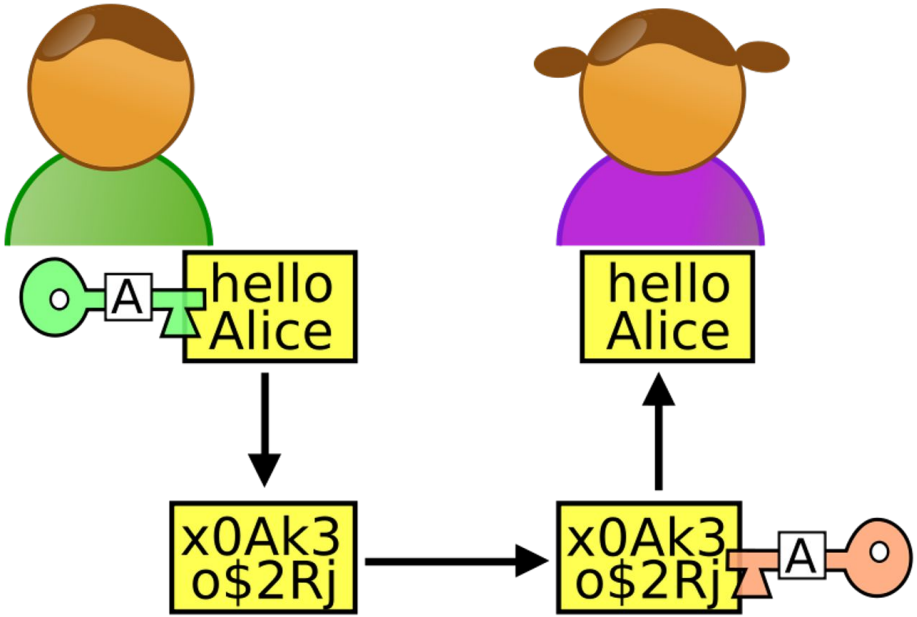
Em resumo:

Passo 1: Alice envia sua chave pública para Bob



Em resumo:

Passo 2: Bob cifra a mensagem com a chave pública de Alice e envia para Alice, que recebe e decifra o texto utilizando sua chave privada.



Em resumo:

Passo 3: E Alice responderia para Bob da mesma forma:

Resposta + ChavePública(Bob) = RespostaCriptografada