João Vítor

Manoel Batista

Desenvolvimento de Sistemas

Igor Lucas

GIT E GITHUB: CONTROLE DE VERSÃO E COLABORAÇÃO EM DESENVOLVIMENTO DE SOFTWARE

Pará de Minas

2025

João Vítor

GIT E GITHUB: CONTROLE DE VERSÃO E COLABORAÇÃO EM DESENVOLVIMENTO DE SOFTWARE

Trabalho apresentado à disciplina de Desenvolvimento Back-End, do curso de Desenvolvimento de Sistemas, da instituição Manoel Batista, como parte dos requisitos para obtenção de nota parcial.

Orientador: Igor Lucas

Pará de Minas

2025

# SUMÁRIO

1. Introdução

2. Metodologia

3. Git: Conceito e Funcionamento

3.1 Histórico do Git

3.2 Principais Comandos e Estrutura Interna

3.3 Workflow de Trabalho com Git

4. GitHub: Plataforma de Hospedagem

4.1 Diferenças entre Git e GitHub

4.2 Recursos do GitHub

4.3 GitHub e a Comunidade de Desenvolvimento

5. Casos de Uso: Aplicações Reais de Git e GitHub

6. Vantagens e Desvantagens

7. Considerações Finais

8. Referências

# 1. INTRODUÇÃO

O controle de versão é uma prática essencial no desenvolvimento de software moderno. Entre as ferramentas mais utilizadas nesse contexto estão o Git, um sistema de controle de versões distribuído, e o GitHub, uma plataforma online baseada em Git que permite hospedagem de repositórios e colaboração entre desenvolvedores. Este trabalho visa apresentar uma análise detalhada sobre o funcionamento, os recursos e a importância dessas ferramentas para o desenvolvimento colaborativo de software.

# 2. METODOLOGIA

A pesquisa foi conduzida por meio de revisão bibliográfica, análise de documentação oficial, tutoriais técnicos e estudos de caso de projetos reais que utilizam Git e GitHub. Também foram considerados artigos acadêmicos e manuais técnicos.

# 3. GIT: CONCEITO E FUNCIONAMENTO

## 3.1 Histórico do Git

O Git foi criado em 2005 por Linus Torvalds, o mesmo criador do kernel do Linux. A ferramenta surgiu a partir da necessidade de um sistema de versionamento descentralizado, eficiente e seguro, após problemas com a ferramenta proprietária BitKeeper.

## 3.2 Principais Comandos e Estrutura Interna

Git trabalha com instantâneos (snapshots) de arquivos ao invés de diferenciais entre versões. Os principais comandos incluem: git init, git add, git commit, git status, git log, git branch, git merge e git checkout.

## 3.3 Workflow de Trabalho com Git

Os fluxos de trabalho mais comuns com Git incluem: Feature Branch Workflow, Git Flow e Forking Workflow. Cada um possui estratégias distintas para ramificação e integração de código.

# 4. GITHUB: PLATAFORMA DE HOSPEDAGEM

## 4.1 Diferenças entre Git e GitHub

Git é o sistema de versionamento; GitHub é uma plataforma online que hospeda repositórios Git e facilita a colaboração entre desenvolvedores por meio de funcionalidades adicionais.

## 4.2 Recursos do GitHub

Incluem repositórios remotos, issues, pull requests, GitHub Actions e GitHub Pages.

## 4.3 GitHub e a Comunidade de Desenvolvimento

GitHub se tornou um pilar do desenvolvimento open-source, abrigando milhões de projetos e facilitando o contato entre desenvolvedores. A plataforma também oferece ferramentas educacionais e suporte à colaboração empresarial.

# 5. CASOS DE USO: APLICAÇÕES REAIS DE GIT E GITHUB

Empresas como Microsoft, Google e startups utilizam Git/GitHub em seus fluxos de desenvolvimento. Projetos como TensorFlow, Linux e React são mantidos por comunidades ativas no GitHub.

# 6. VANTAGENS E DESVANTAGENS

Vantagens:

- Controle de versões eficiente

- Facilidade de colaboração

- Suporte a múltiplos fluxos de trabalho

- Plataforma com ampla integração (GitHub)

Desvantagens:

- Curva de aprendizado inicial

- Conflitos de merge complexos em projetos grandes

- Dependência de linha de comando para uso avançado

# 7. CONSIDERAÇÕES FINAIS

O domínio de ferramentas como Git e GitHub é indispensável para profissionais da área de tecnologia. Além de melhorar a organização dos projetos, essas ferramentas incentivam a colaboração e a transparência no desenvolvimento de software.

# 8. REFERÊNCIAS

# TORVALDS, Linus. Git Documentation. Disponível em: <https://git-scm.com/doc> Acesso em: 05/04/2025. GITHUB Docs. Disponível em: <https://docs.github.com> Acesso em: 05/04/2025.