

## **CONTROLE DE VERSÃO**

Nome: Romão Batista de Aquino

RA: 919120664

Professor: Hebert Bratefixe Alquimim

## Sumário

Introdução	3
Controle de Versão	4
Ferramentas	5
Git	6
GitHub	7
Conclusão	8
Bibliografia	9

# Introdução

Quando se trata de desenvolvedores, é essencial métodos que auxiliem esse trabalho, uma prática que por si só é algo árduo e complexo, um método que é indispensável para todos os desenvolvedores, é o controle de versão, e existem ferramentas que fornecem esse método, como a aplicação Git e o serviço GitHub.

## **Controle de Versão**

O controle de versão é uma das ferramentas mais indispensáveis do desenvolvimento de software, ele serve para manter variações do projeto, controlar múltiplas edições do mesmo projeto em caso de trabalho em equipe e manter o histórico do projeto. Esse processo ajuda a mostrar quem mudou, quando mudou e o que mudou no projeto em questão, auxiliando em futuras atualizações necessárias.

Tendo esse controle do projeto ao todo, a segurança é maior, podendo evitar casos de erros simplesmente voltando a versão anterior do projeto. Possibilitar que várias pessoas editem o mesmo projeto sem que haja sobreposição de arquivos.

## **Ferramentas**

Uma ferramenta de versionamento é extremamente necessária para auxiliar no trabalho, gerenciando todo o projeto e guardando todas as suas versões. Toda ferramenta de versionamento tem como base dois elementos, o repositório e o diretório de trabalho. O repositório é onde armazena o histórico do projeto e o diretório de trabalho é um diretório monitorado pela versão que contém os arquivos do projeto.

As ferramentas de versionamento mais populares atualmente são, Subversion, Git, Mercurial, CVS e TFS possuindo como recursos:

- Gerenciamento do acesso que permite o trabalho em equipe;
- Histórico de alterações desde a primeira versão;
- Possibilita o retorno a uma versão anterior;
- Possui uma ramificação de versões, tendo várias janelas de edição.

# Git

Trata-se de um sistema de controle de versão, onde projetos podem ser editados simultaneamente por várias pessoas sem sobrescrever uma informação na outra. Tendo sua linguagem própria, é necessário saber alguns comandos para usá-lo

Caso você esteja em um projeto com outras pessoas e queira fazer alterações privadas para si mesmo, sem que interfira no projeto original, basta criar uma cópia do projeto, a função chamada “Branch”, aplicando as mudanças nesse branch, não interfere no projeto original, ainda podendo incrementar seu branch no projeto original.

Com o Git, você pega o projeto do GitHub e altera ele no seu próprio monitor, podendo modificá-lo a vontade, ele só fará parte do projeto original, caso você envie esse seu projeto editado, de volta ao GitHub.

# GitHub

O GitHub é como se fosse uma extensão do Git, contendo outras funcionalidades ao Git. Sendo gratuito, você pode guardar seus projetos privados assim como ver outros projetos que pessoas de todo o mundo compartilham, além de acompanhar, contribuir, informando e corrigindo esses projetos.

Como é um serviço, o GitHub não tem uma instalação, você cria uma conta com seu e-mail e então já pode usá-lo, criando repositórios, que são os seus projetos, podendo deixá-los privado ou aberto para todos.

Pode ser comparado a uma “nuvem”, como o Google Drive, entre outros, pois ele armazena o projeto e todos podem pegar para editar caso for público, porém a metodologia é mais organizada e direta.

## **Conclusão**

O controle de versão é uma ferramenta natural para quem quer seguir carreira no meio do TI, tão natural quanto respirar, essa ferramenta é indispensável na vida do programador, dando a chance de consertar erros, testar seus códigos antes de publicar, ou criar um projeto compartilhado e ainda sim organizado, conseguindo acompanhar cada parte do desenvolvimento sem perder nenhum detalhe.



## **Bibliografia**

Tableless:

<https://tableless.com.br/tudo-que-voce-queria-saber-sobre-git-e-github-mas-tinha-vergonha-de-perguntar/>

Gaea:

<https://gaea.com.br/sabe-fazer-versionamento-de-software/>

<https://gaea.com.br/entenda-por-que-versionamento-de-software-e-tao-importante/>