

**Black Friday:** Apenas **R\$ 1,99** no 1º mês e R\$ 49,90 nos demais!**ASSINE AGORA**

Artigo

Invista em você! Saiba como a DevMedia pode ajudar sua carreira.

## Sistemas de Controle de Versão

Veja nesse artigo uma introdução aos Sistemas de Controle de Versão, indicado para desenvolvedores que estão começando agora ou que ainda não tiveram contato com um projeto maior que seja necessário o desenvolvimento em paralelo.



Anotar



Marcar como concluído

Artigos



Canal Mais



Sistemas de Controle de Versão

Utilizamos cookies para fornecer uma melhor experiência para nossos usuários. Para saber mais sobre o uso de cookies, consulte nossa política de privacidade. Ao continuar navegando em nosso site, você concorda com a nossa política.

Aceitar

**Black Friday:** Apenas **R\$ 1,99** no 1º mês e R\$ 49,90 nos demais!

**ASSINE AGORA**

sem mexer na versão principal.

Um Iniciante da área de desenvolvimento de software ao se deparar com seu primeiro projeto de maior porte com uma equipe de desenvolvimento, geralmente, não conhece as ferramentas usadas para trabalhar paralelamente de forma eficiente e acaba sempre com aquela terrível dúvida:

Como iremos desenvolver o projeto em paralelo? Existe a possibilidade de fazer isso sem que haja sobreposição das minhas alterações ou alguém esteja utilizando a versão errada? Como iremos resolver esse problema sem apelar para a nossa linda, muitas vezes útil, gambiarra?

É aí que surge a solução: usar um sistema de controle de versão! Ótimo... Mas o que é um sistema de controle de versão? Como ele funciona?

## Como funciona um Controle de Versão?

Basicamente, os arquivos do projeto ficam armazenados em um repositório (um servidor em outras palavras) e o histórico de suas versões é salvo nele. Os desenvolvedores podem acessar e resgatar a última versão disponível e fazer uma cópia local, na qual poderão trabalhar em cima dela e continuar o processo de desenvolvimento. A cada alteração feita, é possível enviar novamente ao servidor.

Utilizamos cookies para fornecer uma melhor experiência para nossos usuários. Para saber mais sobre o uso de cookies, consulte nossa política de privacidade. Ao continuar navegando em nosso site, você concorda com a nossa política.

Aceitar

**Black Friday:** Apenas **R\$ 1,99** no 1º mês e R\$ 49,90 nos demais!

**ASSINE AGORA**



Estação de Trabalho



Estação de Trabalho

**Figura 1:** Esquema de funcionamento dos controles de versão

E se por acaso os desenvolvedores estiverem editando o mesmo arquivo? O que irá acontecer se enviarem ao mesmo tempo para o servidor?

Para evitar problemas como esse, o Sistema de Controle de Versão oferece ferramentas uteis para mesclar o código e evitar conflitos.

---

Relacionado: [Guia Completo de Git e Github](#)

---

Por exemplo: Você atualizou seu projeto (usando uma função chamada de check-out ou update) e começou a fazer suas alterações. Ao mesmo tempo, outro desenvolvedor fez alterações e atualizou a versão no servidor. Quando for enviar sua versão (usando uma função chamada de check-in ou commit) o Sistema de

Utilizamos cookies para fornecer uma melhor experiência para nossos usuários. Para saber mais sobre o uso de cookies, consulte nossa política de privacidade. Ao continuar navegando em nosso site, você concorda com a nossa política.

Aceitar

**Black Friday:** Apenas **R\$ 1,99** no 1º mês e R\$ 49,90 nos demais!

**ASSINE AGORA**

O **centralizado** trabalha apenas com um servidor central e diversas áreas de trabalho, baseados na arquitetura cliente-servidor. Por ser centralizado, as áreas de trabalho precisam primeiro passar pelo servidor para poderem comunicar-se. Essa versão atende muito bem a maioria das equipes de desenvolvimento que não sejam enormes e trabalhem em uma rede local, além de não ter problemas de velocidade para enviar e receber os dados e ter um bom tempo de resposta do servidor. Um dos principais sistemas com o tipo de controle de versão centralizado é o Subversion.

O **distribuído** vai mais além. Ele é recomendado para equipes com muitos desenvolvedores e que se encontram em diferentes filiais. Esta versão funciona da seguinte maneira: cada área de trabalho tem seu próprio “servidor”, ou seja, as operações de check-in e check-out são feitas na própria máquina. Porém diferentemente do centralizado, as áreas de trabalho podem comunicar-se entre si, recomenda-se usar um servidor como centro do envio dos arquivos para centralizar o fluxo e evitar ramificações do projeto e a perda do controle sobre o mesmo, geralmente o sistema te dá essa opção, oferecendo um servidor remoto para hospedar o projeto. A comunicação entre o servidor principal e as áreas de trabalho funciona com outras duas operações, para atualizar e mesclar o projeto, chamadas de `pull` e `push` (puxar e empurrar).

Utilizamos cookies para fornecer uma melhor experiência para nossos usuários. Para saber mais sobre o uso de cookies, consulte nossa política de privacidade. Ao continuar navegando em nosso site, você concorda com a nossa política.

Aceitar

**Black Friday:** Apenas **R\$ 1,99** no 1º mês e R\$ 49,90 nos demais!

**ASSINE AGORA**

situações não permite a comparação entre atualizações concorrentes. O sistema centralizado trabalha com arquivos de texto, que permite a comparação em atualizações concorrentes e da opção ao desenvolvedor para escolher a melhor solução.

Portanto, por esse tratamento de mesclagem ser diferente, podem ocorrer situações onde o trabalho de alguém possa ser sobreposto e gerando tormento para os desenvolvedores. Para isso existe uma função chamada lock, que bloqueia o arquivo para que não seja modificado por outros enquanto estiver com você. Os sistemas distribuídos mais conhecidos são o Git e o Mercurial.

A maioria das aplicações para desenvolvimento de software contam com plug-ins para o uso dos , como o NetBeans, Eclipse, Delphi XE2 e o Visual Studio. O Sistema de Controle de Versão é uma ótima ferramenta para o desenvolvimento do seu projeto e com certeza, deve ser usada e bem aproveitada para o bem do mesmo.

---

## Saiu da DevMedia!

### ■ Controle de acesso no Laravel com ACL:

Neste curso vamos aprender como implementar um sistema de autorização utilizando o Laravel, framework PHP mais utilizado no mundo. Para isso, partiremos de uma aplicação que possui toda a parte de registro e autenticação

Utilizamos cookies para fornecer uma melhor experiência para nossos usuários. Para saber mais sobre o uso de cookies, consulte nossa política de privacidade. Ao continuar navegando em nosso site, você concorda com a nossa política.

Aceitar

**Black Friday:** Apenas **R\$ 1,99** no 1º mês e R\$ 49,90 nos demais!

**ASSINE AGORA**

Neste curso você aprenderá o que é o Git, ferramenta para controle de versão mais utilizada por desenvolvedores de software. Você aprenderá sobre a importância de utilizar um sistema de controle de versão, como o Git funciona, assim como os principais comandos para criar um repositório, avaliar o status do diretório do projeto, criar versões do código, entre outras tarefas que simplificam o nosso dia a dia.

#### ■ O que é GitHub?:

Neste curso você aprenderá o que é o GitHub, um serviço web de hospedagem de código construído sobre o Git que possibilita o controle de versão de projetos de software. Para que você possa conhecê-lo, vamos abordar alguns motivos que demonstram o quão positivo pode ser a adoção dessa tecnologia, assim como analisaremos, na teoria e na prática, as configurações para uso e o fluxo de trabalho com o GitHub.

#### ■ Curso de Controle de Versão com Git:

Este curso aborda o tema controle de versões distribuído com Git. Durante este treinamento consideraremos tópicos conceituais e práticos sobre o controle de versões em empresas de software.

#### ■ Controle de versões distribuído com Git:

Nesta palestra faremos uma introdução ao sistema de controle de versão distribuído conhecido como Git.

#### ■ Gerência de mudanças utilizando Git:

Esse artigo apresenta como implantar um sistema controle de versão para pequenos projetos utilizando o serviço de hospedagem Bitbucket e o sistema de controle de versões Git.



Anotar



Marcar como concluído

# Black Friday DevMedia

Utilizamos cookies para fornecer uma melhor experiência para nossos usuários. Para saber mais sobre o uso de cookies, consulte nossa política de privacidade. Ao continuar navegando em nosso site, você concorda com a nossa política.

Aceitar

**Black Friday:** Apenas **R\$ 1,99** no 1º mês e R\$ 49,90 nos demais!

**ASSINE AGORA**

Ainda está em dúvida? Experimente a plataforma durante 3 dias sem cartão. **Faça um teste grátis**

## BENEFÍCIOS

Suporte em tempo real

Certificado de autoridade

Exercícios para praticar

Estudo gamificado

Planos de estudo para  
cada carreira de  
programador

**Saiba mais**



Por **Eduardo**

Em 2012

Utilizamos cookies para fornecer uma melhor experiência para nossos usuários. Para saber mais sobre o uso de cookies, consulte nossa política de privacidade. Ao continuar navegando em nosso site, você concorda com a nossa política.

Aceitar

DEV MEDIA

**Black Friday:** Apenas **R\$ 1,99** no 1º mês e R\$ 49,90 nos demais!**ASSINE AGORA**

## Suporte ao aluno - Tire a sua dúvida.

[Tecnologias](#)[Exercícios](#)[Cursos](#)[Artigos](#)[Revistas](#)[Fale conosco](#)[Plano para Instituição de ensino](#)[Assinatura para empresas](#)[Assine agora](#)

Utilizamos cookies para fornecer uma melhor experiência para nossos usuários. Para saber mais sobre o uso de cookies, consulte nossa política de privacidade. Ao continuar navegando em nosso site, você concorda com a nossa política.

[Aceitar](#)