



**CURSO DE ANÁLISE E DESENVOLVIMENTO DE SISTEMAS**  
**ARTIGO SOBRE PADRONIZAÇÃO DE PROJETOS**

Danillo de Souza Koch

**Padrões de Projetos**  
**Model-View-Controller (MVC)**

**JOINVILLE**  
**2025**

**Nome: Danillo de Souza Koch**

Padrões de projetos (Design Patterns) são soluções reutilizáveis aplicadas para resolver problemas recorrentes no desenvolvimento de software. Eles ajudam a manter o código mais claro, padronizado e fácil de dar manutenção, além de melhorar a comunicação entre os desenvolvedores. Entre os padrões mais utilizados estão o Singleton, o Factory Method e o MVC.

O padrão **Singleton** é empregado quando se deseja garantir que uma classe possua apenas uma única instância disponível para toda a aplicação. Ele controla a criação do objeto e assegura que exista apenas um ponto de acesso centralizado.

Já o **Factory Method** é um padrão voltado para sistemas que precisam de flexibilidade na criação de objetos. Ele permite que uma superclasse defina a estrutura básica, enquanto delega às subclasses a responsabilidade de determinar qual tipo de objeto será criado. Isso facilita o escalonamento e a expansão do sistema.

O padrão **MVC (Model-View-Controller)** divide o software em três camadas principais para melhorar a organização e a manutenção do código. A camada Controller recebe os dados do usuário e os envia ao Model; este processa a lógica e retorna os resultados ao Controller, que repassa as informações para a camada View, responsável por exibir tudo ao usuário. Essa arquitetura é muito utilizada em aplicações web pela sua clareza e eficiência.

Além disso, os **pacotes em PHP** são conjuntos de classes ou bibliotecas reutilizáveis criadas para resolver problemas específicos e que podem ser adicionados facilmente ao projeto. Geralmente gerenciados pelo Composer, os pacotes tornam o desenvolvimento mais ágil ao evitar a reescrita de funcionalidades comuns, além de melhorar a modularização do sistema. A nomenclatura segue o padrão “vendor/package”, onde o primeiro termo identifica o autor ou organização do pacote, e o segundo corresponde ao nome da biblioteca, ambos escritos em letras minúsculas e separados por barra.

## Referência

ALURA. *PHP na Web: conhecendo o padrão MVC*. Disponível em:

<https://www.alura.com.br/conteudo/php-web-conhecendo-padrao-mvc>

Acesso em: 10 dez. 2025.

MOZILLA FOUNDATION. *Server-side website programming – Model View Controller (MVC)*.

Disponível em:

[https://developer.mozilla.org/en-US/docs/Learn/Server-side/Express\\_Nodejs/mvc](https://developer.mozilla.org/en-US/docs/Learn/Server-side/Express_Nodejs/mvc)

Acesso em: 10 dez. 2025.