# **PROJETO DE SOFTWARE ORIENTADO A OBJETOS**

Nesta parte do projeto serão apresentados o Padrão de Desenvolvimento MVC e os Princípios SOLID. Ambos conteúdos foram trabalhados em sala de aula na matéria de Projeto de Software Orientado a objetos e estão sendo anexados aqui a pedido da professora para que possa ser constituída a nota do Trabalho de Conclusão de Curso.

## **Arquitetura de Aplicação Padrão MVC**

As arquiteturas de aplicação são modelos que definem a estrutura na qual determinadas aplicações devem ser desenvolvidas englobando componentes de software e suas propriedades, o relacionamento entre esses componentes e os elementos que compõe a estrutura e como se comunicam. Neste tópico será explicado como funciona a Arquitetura de Aplicação MVC, que está sendo utilizada para desenvolver o sistema que será apresentado no Trabalho de Conclusão de Curso.

Desenvolvido em 1979 por Trygve Reenskaug que inicialmente foi criado para desenvolver aplicações desktop, mas se popularizou em sistemas web. Esse é um padrão que ainda é muito utilizado até hoje.

O padrão MVC consiste em dividir a aplicação em três camadas que se comunicam entre si, sendo elas:

* Model: Essa camada serve como um modelo, ela contém a estrutura dos dados utilizados na aplicação, ela lê, manipula e valida os dados e trata regras de negócio.
* View: Apresenta uma interface que exibe uma interpretação dos dados processados pela aplicação, e é a única que realiza contato direto com o usuário final.
* Control: A camada controle é responsável por gerenciar todos os processos realizados pela aplicação, ela interpreta as requisições realizadas pelo usuário e processa se essas informações devem ser enviadas para a camada View ou para a camada Model.

Esse padrão facilita a progamação orientada a objetos ao realizar essa divisão em três camadas pois possibilita que seja realizada uma mesma lógica em diversas interfaces, alterações feitas em uma camada automaticamente atualizam as outras sem que seja necessário realizar alterações em outras partes da aplicação, vale ressaltar que podem ocorrer exceções. Outro fator importante é a segurança e o encapsulamento do código, que são bem maiores ao se utilizar esse padrão pois cada camada atua com seu próprio código e somente transmite dados e comandos através da camada Control que administra todas as outras, A camada Model serve somente como modelo nela não é executado nenhum processo que seja resultado de uma interação com o usuário, ela serve apenas como um modelo para realizar as operações necessárias, a View serve de intermédio com o usuário, apresentando uma interface, e todas as operações que necessitam processamento de informações e funções do sistema são enviados para a camada Control que é a responsável pelo gerenciamento e execução dos principais processos de uma aplicação, a View só acessa a View por intermédio da camada Control.

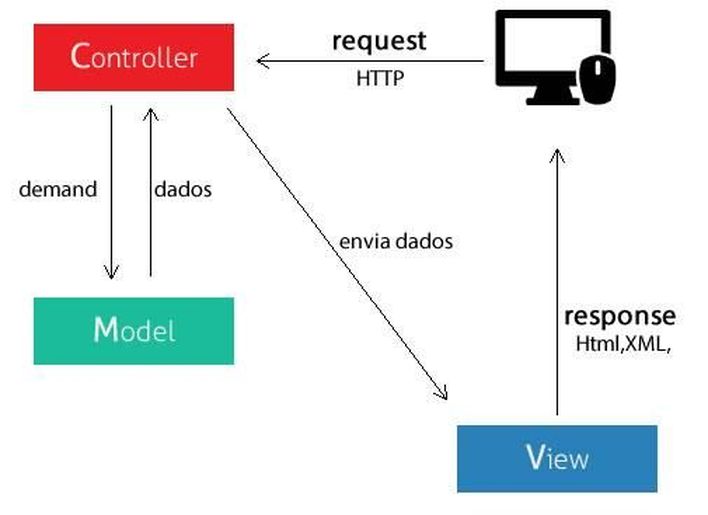


Figura 1 - Exemplo funcionamento MVC

## **Princípios SOLID**