

# MVC e MTV; Django

O Django não é identificado como um MVC diretamente, mas sim como uma variação, um framework MTV. Na interpretação, define-se "View" como a informação que será apresentada para o usuário. Então, "View" é como a informação será apresentada para o usuário. Não é necessariamente o "como", mas o "qual". "View" descreve qual informação é vista, não como ela é vista. Nesse caso, a view é a função callback de Python para uma URL específica. Porque essa função callback descreve qual informação será apresentada.

No Django, uma "view" descreve quais dados são apresentados, mas normalmente uma view delega a apresentação para um template, que descreve como os dados serão apresentados.

Quanto ao controller, no caso do Django, ele seria o próprio framework, que funciona como a "máquina" que envia a requisição para uma view apropriada.

Embora o Django use o termo MTV, a arquitetura é semelhante ao modelo MVC. No MVC, o controller é responsável por receber a requisição e organizar a lógica e a resposta. No Django, essa responsabilidade é principalmente executada pela View, enquanto a camada de Template assume uma função mais próxima da "View" do MVC, lidando com a apresentação visual.

O Django é um framework web de alto nível que facilita o desenvolvimento rápido de aplicações web seguras, escaláveis e com código limpo, cuidando de grande parte da complexidade do desenvolvimento e permitindo que o desenvolvedor foque nas regras de negócio.

## MTV e MTV

### M em MTV e MVC:

Ambos os padrões compartilham um M que representa o **Modelo**, responsável por gerenciar os dados e a lógica de negócios. Neste componente estão as interações com o banco de dados, o armazenamento de dados e as regras de negócios.

### V em MTV e MVC:

No MVC, o **View** é a interface do usuário, a representação visual dos dados, o que o usuário vê. O **View** obtém os dados do **Modelo** e os apresenta de alguma forma.

No MTV, o **View** no Django não é a interface do usuário, mas sim a função que recebe as requisições HTTP, processa os dados e decide qual informação devolver. Portanto, no MTV, o **View** é o controlador das requisições, mas não está diretamente relacionado à apresentação dos dados.

### T e C em MTV e MVC:

No MVC, o **Controller** gerencia a lógica da aplicação, recebe as solicitações dos usuários, interage com o **Modelo** e então devolve a resposta para o **View**. Pode-se dizer que é o "intermediário" entre o **Modelo** e o **View**.

Enquanto no MTV, o **Template** é responsável pela apresentação. Enquanto no MVC o **Controller** decide quais dados são mostrados, no MTV, o **View** faz isso, mas a apresentação visual desses dados é gerida pelo **Template**.