



**Certified Tech
Developer**
The Ultimate Degree

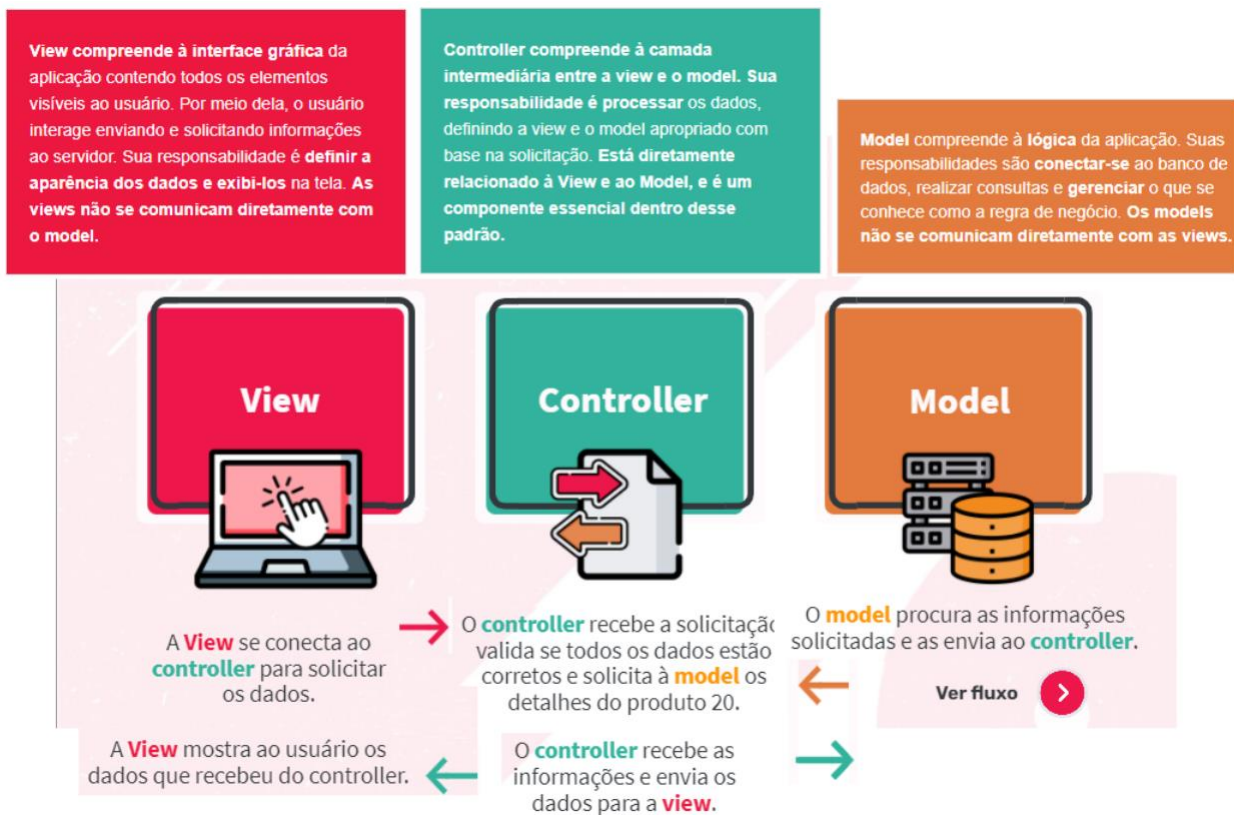
MVC

Vantagens MVC



Vamos ver um exemplo

Suponha que um usuário percorra uma lista de produtos através de uma interface gráfica e queira solicitar mais informações sobre o produto 20. Vamos ver o fluxo do modelo MVC.



Spring MVC e Spring Boot MVC

Na aula anterior, vimos a diferença entre Spring e Spring Boot. Por sua vez, mencionamos que o Spring Platform possui uma grande variedade de ferramentas para facilitar o trabalho do desenvolvimento. Um deles é o Spring Framework, que nos fornece um módulo que segue o padrão de design MVC e que se denomina de Spring MVC.

Como funciona o Spring MVC?

O Spring MVC tem um componente conhecido como **DispatcherServlet**, que é o responsável por delegar e coordenar as várias interfaces de controle para execução durante uma solicitação HTTP, como um controlador frontal (front controller). Isso significa que todas as solicitações provenientes de um recurso da aplicação serão tratadas por um único controlador e, em seguida, enviadas para o controlador apropriado para o tipo de solicitação. O controlador frontal (front controller) pode usar outros ajudantes para obter o mecanismo de encaminhamento.

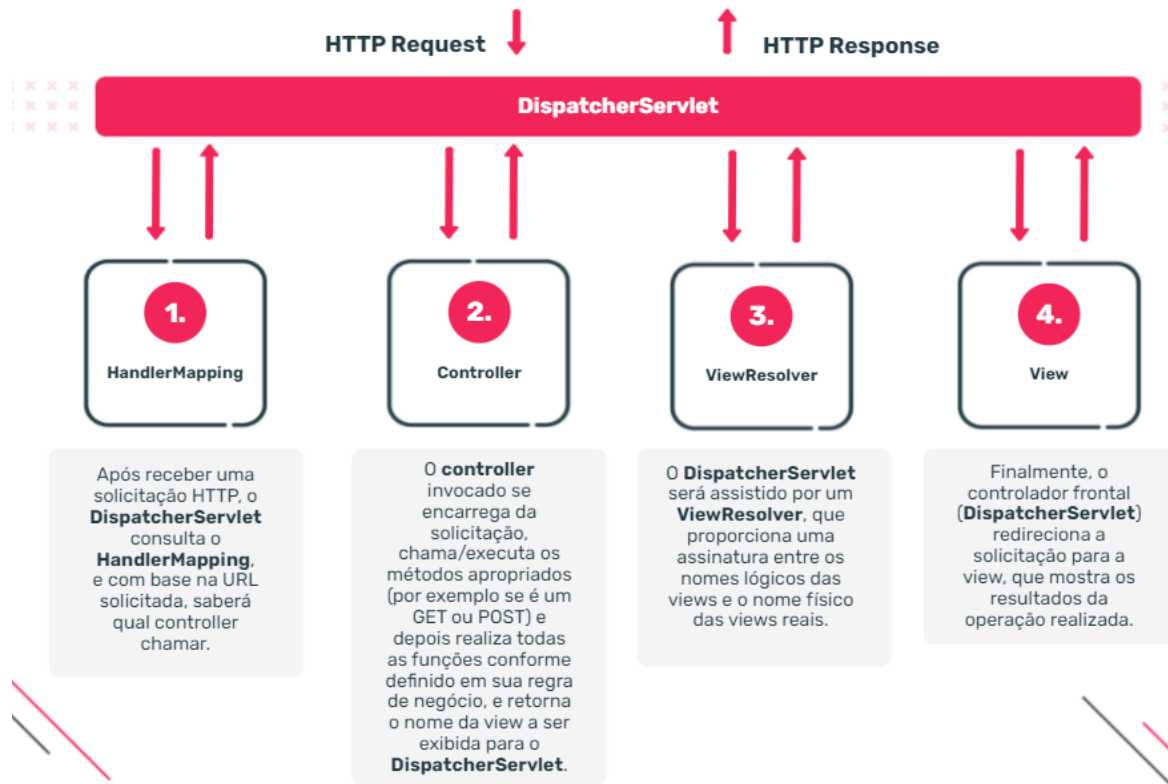
DispatcherServlet

DispatcherServlet



Vamos ver em detalhe como o DispatcherServlet funciona:

Descubra o que acontece em cada etapa clicando no número correspondente.



Spring Boot MVC

Vamos ver agora como o Spring Boot se integra ao Spring MVC para simplificar a integração entre o Model, a View e o Controller:

Tecnologias para a camada de apresentação

O Spring MVC oferece suporte a diversos tipos de views para diferentes tecnologias de apresentação: geralmente chamamos de view templates.

Essas tecnologias são descritas como templates, pois disponibilizam uma linguagem de marcação para poder expor os atributos da model, na view, durante sua renderização no servidor. Alguns desses templates: