



Java Orientado a Objetos

by Antonio Rodrigues Carvalho Neto



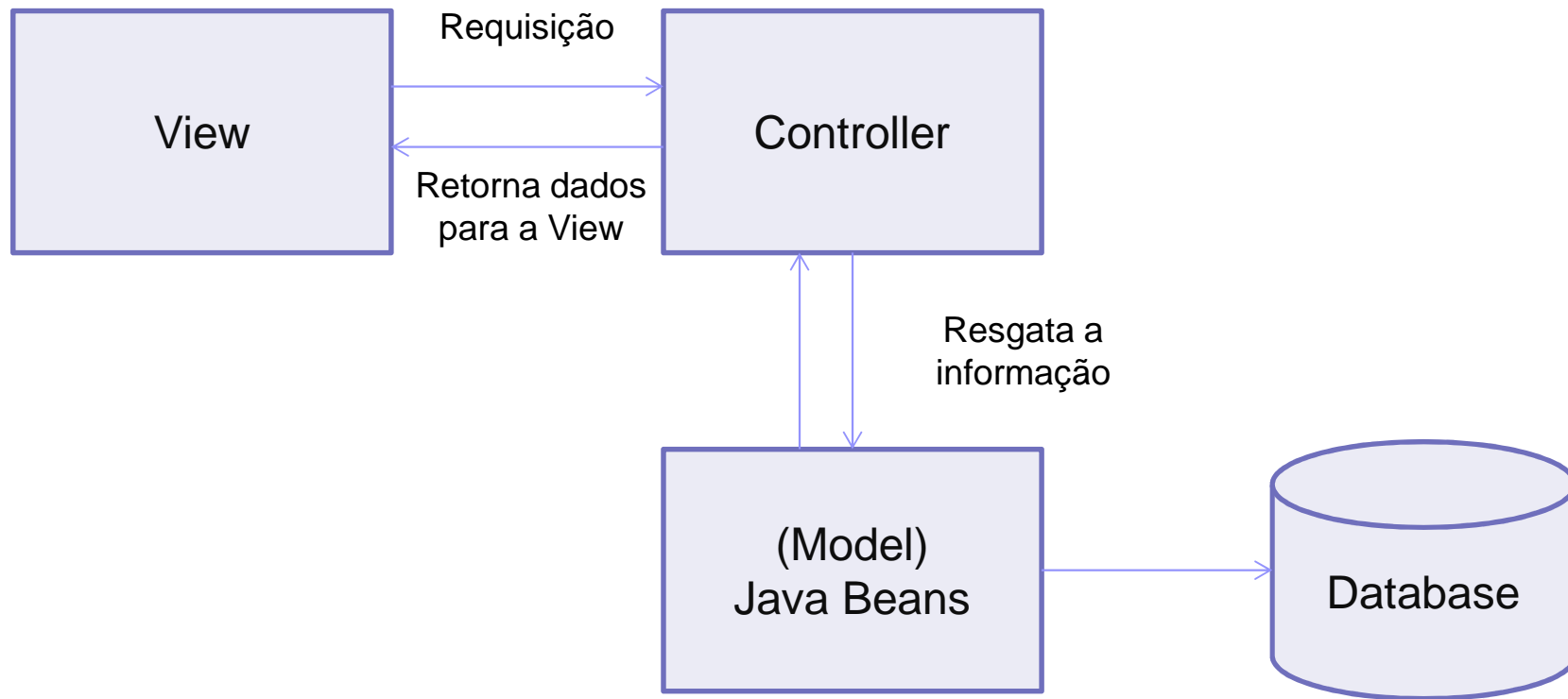
(MVC) Model View Controller



Padrão de Projeto MVC

- Conforme a quantidade de requisitos atendidos cresce, a aplicação se torna maior e mais complexa dificultando as manutenções e modificações.
- Para facilitar o desenvolvimento e a manutenção das aplicações, surgiu um padrão chamado MVC (Model View Controller) que divide a aplicação em segmentos de forma que cada um deles represente uma perspectiva do software.
- Em outras palavras quando usamos o MVC nossa aplicação será dividida da seguinte forma:
 - **Model**, serão classes responsáveis para lidar com a manipulação dos dados, regras de negócios e atualização do banco de dados.
 - **View**, são classes que voltadas a mostrar os dados para o usuário final, de forma a não se preocuparem com o regras de negócio, nem como os dados serão extraídos ou inseridos no banco de dados.
 - **Controller**, este último interage com o usuário, acionando a View solicitada e o Modelo (Objeto) que será mostrado nesta view. As interações dos usuários com seu sistema são direcionados para os controllers.

Arquitetura MVC (aplicação desktop)





View

- As classes de visualização dos dados são feitas normalmente em uma linguagem própria que possua comandos fáceis para mostrar os dados. Na plataforma Standard Edition (aplicações em desktop) as views serão classes responsáveis por montar a interface com o usuário, através do uso de uma biblioteca gráfica como Swing ou SWT.



Model

- As classes de modelos normalmente são Java Beans que possuem métodos para gravar, ler suas propriedades. Existem classes adicionais que podem ser criadas para ler e gravar os java beans nos repositórios de dados.
- A classe de modelo contém atributos de uma determinada entidade.
- Estes atributos são acessados apenas por meio de Gets and Sets.



Controler

- A classe de controle interage com o usuário recebendo a sua requisição. A partir deste recebimento ela direciona qual modelo deverá ser criado ou resgatado do repositório de dados e qual view será disponibilizada para o usuário, para mostrar este modelo.