



FIAP



# Domain Driven Design using Java





# AGENDA



1

API RESTful



2

Exercícios



# Introdução

- Padrões abordados:

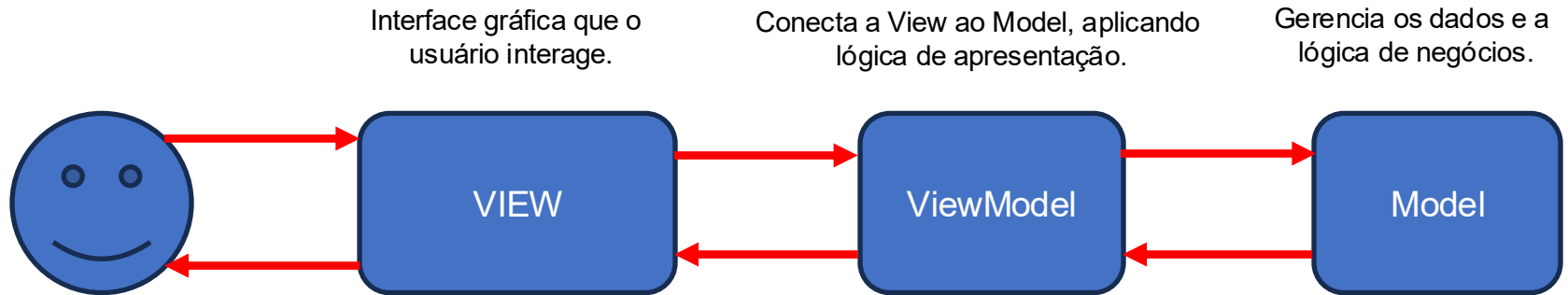
- Arquitetura em Camadas (*Layered Architecture*) 
- Arquitetura MVC 
- Arquitetura MVVW (Model-View-ViewModel)
- Arquitetura RESTful
- Arquitetura de Microsserviços
- Arquitetura Baseada em Eventos
- Padrão de Camadas Hexagonais (Arquitetura Hexagonal ou Ports and Adapters)
- Arquitetura CQRS (Command Query Responsibility Segregation)
- Padrão Serverless
- Padrão de Arquitetura Orientada a Serviços (SOA)
- Arquitetura Monolítica

# Arquitetura MVVW (Model-View-ViewModel)

## Arquitetura MVVM (Model-View-ViewModel)

- O que é?
  - MVVM é um padrão arquitetural que separa a interface do usuário (UI) da lógica de negócios e dados. Ele é amplamente utilizado em aplicações modernas, como aplicativos desktop e mobile, para melhorar modularidade e facilitar testes.

## Estrutura da Arquitetura MVVM



**Usuário.**

## Detalhamento dos componentes

- **Model:**
  - Representa os dados e as regras de negócios.
  - Interage com bancos de dados, APIs e fontes externas.
- **View:**
  - Exibe os dados para o usuário.
  - Captura interações e eventos do usuário..
- **ViewModel:**
  - Intermedia entre a View e o Model..
  - Atualiza a interface com base nas mudanças nos dados do Model.
  - Implementa lógica de apresentação (ex.: formatação).



# Arquitetura RESTful

# Introdução à Arquitetura RESTful

- O que é?

- REST (**Representational State Transfer**) é um estilo arquitetural para construção de APIs Web.
- Uma **API RESTful** é uma interface que adota os princípios REST para comunicação entre sistemas.

- Por que usar RESTful?

- Simplicidade e padronização na troca de informações.
- Separação entre cliente e servidor.
- Alta escalabilidade e independência tecnológica.

## Características de uma API RESTful

- Stateless (Sem Estado)
  - Cada requisição é independente e não depende de requisições anteriores.
- Baseada em Recursos
  - Endpoints representam **objetos** do sistema, como */produtos*, */usuarios*.
- Uso de Métodos HTTP
  - Define operações claras sobre os recursos (CRUD).
- Formato Padrão para Respostas
  - JSON ou XML são os formatos mais comuns para troca de dados.

# Métodos HTTP e Seus Propósitos

## Métodos HTTP na API RESTful

<b>Método</b>	<b>Descrição</b>	<b>Exemplo de Endpoint</b>
<b>GET</b>	Recupera um recurso.	GET /produtos
<b>POST</b>	Cria um novo recurso.	POST /produtos
<b>PUT</b>	Atualiza um recurso existente.	PUT /produtos/{id}
<b>PATCH</b>	Atualiza parcialmente um recurso.	PATCH /produtos/{id}
<b>DELETE</b>	Remove um recurso.	DELETE /produtos/{id}

## Código HTTP de Resposta

<b>Código</b>	<b>Significado</b>
<b>200 OK</b>	Requisição bem-sucedida.
<b>201 Created</b>	Recurso criado com sucesso.
<b>204 No Content</b>	Recurso atualizado ou removido, sem resposta.
<b>400 Bad Request</b>	Erro na requisição do cliente.
<b>401 Unauthorized</b>	Autenticação necessária.
<b>403 Forbidden</b>	Acesso negado ao recurso.
<b>404 Not Found</b>	Recurso não encontrado.
<b>500 Internal Server Error</b>	Erro inesperado no servidor.

# Boas Práticas para Desenvolver APIs RESTful

# Estruturando Endpoints de Forma Clara

- URLs descritivas e sem verbos

- /clientes (Lista de clientes)



- /clientes/{id} (Obter um cliente específico)



- URLs no formato errado

- /getClient?id=10

- /listarTodosOsClientes



## Utilizando Status Codes Apropriadamente

- Ao criar um novo registro: Retornar 201 Created.
- Ao consultar um recurso existente: Retornar 200 OK.
- Se um recurso não for encontrado: Retornar 404 Not Found.

# Segurança em APIs RESTful

- Autenticação e Autorização
  - Utilize **JWT** (JSON Web Token) ou **OAuth 2.0**.
- Rate Limiting
  - Limite requisições para evitar sobrecarga no servidor.
- Uso obrigatório de HTTPS
  - Protege contra ataques de interceptação.

# Exemplos Práticos de Implementação

## Ferramenta de teste - Postman

- <https://www.postman.com/downloads/>

## Projeto Abertura de Mercado

- <http://documenter.getpostman.com/view/30322966/2sAXqqchhJ>



FIAP

