

Arquitetura Model-View-Control clássica (toda no servidor)

A chamada arquitetura MVC pode ser implementada de várias maneiras. Para tornar a apresentação mais concreta fizemos algumas escolhas ligadas a uma implementação particular.

1. Model:

A parte do modelo é responsável pelo cálculo das funcionalidades últimas da aplicação.

Um objeto da parte do modelo deve receber os dados de uma forma padronizada (por exemplo, através um objeto DTO passado como argumento de um método) e devolver os resultados também de forma padronizada (por exemplo, através um objeto DTO pelo valor de retorno do método).

Os objetos DTO de entrada de dados e de saída de dados são geralmente de classes diferentes. Os valores passados por estes DTOs devem ser próprios da funcionalidade do modelo e NÃO devem possuir nenhuma característica ligada aos aspectos WEB desta aplicação.

A classe do modelo deve poder ser reutilizada em qualquer outro tipo de aplicação, seja web, desktop, nativa móvel, embarcada ou outra.

2. View:

A parte da visão é geralmente composta por uma página JSP. A página JSP, geralmente, recebe os dados dinâmicos através atributos passados por expressões da linguagem EL.

3. Control:

A parte de controle tem como objetivo receber os objetos `HttpRequest` e `HttpResponse`, extrair do primeiro o nome da classe que deve tratar o pedido (handler ou tratador) colocar os dados no DTO apropriado e fazer o pedido ao objeto do modelo.

Para organizar melhor a parte de controle, usamos um único servlet para receber todos os pedidos. O método do servlet que recebe os pedidos HTTP extrai destes o nome do tratador apropriado.

Temos então um tratador para cada página. Página e respectivo tratador são definidos sempre aos pares. Cada página possui um elemento input do tipo hidden com o nome da classe do tratador correspondente. Todo tratador implementa a interface `IFTratador` e implementa o método `processar`. Este método `processar` é o método chamado pelo servlet e passa os objetos `HttpRequest` e `HttpResponse`.

O tratador é então capaz de tratar todos os pedidos feitos a partir da sua página, repassar o pedido para o objeto do modelo, receber sua resposta e encaminhar o fluxo de controle para a JSP de resposta.

Na página seguinte apresentamos uma figura que representa a arquitetura.

SERVIDOR JAVA WEB (TOMCAT) E APLICAÇÃO COM ARQUITETURA MVC

