Introdução ao padrão de projeto MVC – Model-View-Controller

Miguel Jonathan – IM/DCC e NCE – UFRJ (junho de 2011)

Nesse texto introduzimos o padrão MVC, com alguns exemplos simples:

A ideia é dividir uma aplicação em três pacotes de classes:

a) model (modelo): contém as classes que descrevem os objetos da aplicação, por exemplo, alunos, disciplinas, turmas, professores, etc.

b) controller (controlador): cada objeto controlador é uma espécie de "gerente" da aplicação, encarregado de intermediar as solicitações dos usuários feitas através da interface com o usuário (vista). O controlador tem acesso aos objetos do modelo, às estruturas de dados, e embute nos seus métodos as "regras de negócio" da aplicação. As regras de negócio contêm a "lógica da aplicação": filtros para controlar a criação de objetos e o acesso aos dados, valores máximos e mínimos aceitáveis, etc.

   Uma aplicação simples pode ter um só objeto controlador, mas se for um sistema mais complexo pode preferir utilizar um controlador para cada subsistema. O pacote deve conter as classes controladoras, e também as classes que representam as exceções que o controlador pode lançar, sempre que uma solicitação violar alguma das regras. Em geral se utiliza o padrão de projeto "Singleton" para garantir que sempre haverá no máximo uma instância de cada controlador. Isso é conseguido fazendo o construtor do controlador ser privado.

   Os métodos do controlador não devem ser escritos com uma interface específica com o usuário em mente. Seus métodos devem receber argumentos e retornar valores ou lançar exceções. Dessa forma, a aplicação fica independente da interface usada para acessá-lo.

c) view (vista, a interface com o usuário): este pacote contém as classes que fazem a interação com o usuário. Uma aplicação pode utilizar mais de uma interface diferente, e pode trocar de interface, sem que seja necessário interferir com o restante da aplicação. Para conseguir isso, a interface captura as solicitações do usuário e as converte em mensagens para o controlador. As exceções lançadas pelo controlador são capturadas pelos métodos da vista, que geram as ações correspondentes para repassá-las ao usuário. Para poder fazer isso, cada vista deve ter acesso a objetos controladores.

Em resumo, o uso do padrão MVC permite desacoplar a interface da aplicação em si. A interface não conhece a estrutura da aplicação, e a aplicação não depende de nenhuma interface em particular. Um sistema bem desenhado com MVC permite trocar de interface mais facilmente, assim como permite mudar a estrutura interna dos dados sem alterar a interface com o usuário.

Os exemplos abaixo ajudam a entender o mecanismo usado.