

Mar 07

Divisão de responsabilidades

Postado em : Sábado, 7 de Março de 2009 | Autor: Paulo Teixeira

Aproveitando o assunto que o nosso amigo Tofinha abordou " **Por que utilizar ColdFusion Components – CFCs?** " e dentro deste assunto o mesmo abordou sobre OOP, um modelo de programação que eu simplesmente adoro. Uso em todo trabalho possível, seja em ColdFusion (como disse o Tofinha, sim é possível) como em outras tecnologias que trabalho.

Gostaria de falar sobre um fato que rolou no dia-a-dia do meu trabalho, onde fui "comentado", por um projeto onde eu fiz a arquitetura e eu desenvolvi sozinho. Esse projeto integrava-se com outras tecnologias como VB e C++, sendo que essa integração não foi bem aceita. Mas não vem ao caso agora.

O que aconteceu de interessante que me inspirou esse post, foi um "comentário" que fizeram a respeito do meu trabalho, mais ou menos assim:

"O Paulo fez uns componentes, onde nesses componentes tinham métodos que carregavam outros métodos e não se entendia quem era quem!!!, por que não fez tudo no mesmo método."

Bom eu achei interessante o comentário gente, sinceramente. Mas isso me fez vim aqui postar sobre isso. Na verdade sobre o que é esse negócio complicado de um método chamar outro método e etc...

Eu acredito que a principal responsabilidade da Orientação a Objetos, é tentar criar modelos que trabalhem de uma forma mais parecida com o mundo real, por isso que em todo livro, apostila e todo tipo de material sobre OOP, temos aquele velho exemplo do objeto cachorro, carro e pessoa. Tenho um amigo que diz que só entendeu OOP quando entendeu que tudo na vida real é um Objeto. Inclusive ele! Pois é, ele está com grande razão nisso, podemos pensar assim.

E é pensando assim que temos como uma grande função da Orientação a Objetos a Divisão de Responsabilidades ou Separação de Responsabilidades como achar melhor dizer.

Pensando como meu amigo, que somos um objeto, o Objeto Paulo, tem métodos "Pegar, Pisar, Falar".... e etc... é possível o método "Falar", pegar alguma coisa?

Creio que não né? Então para o braço "Pegar" alguma coisa ele não pode usar o método "Falar" certo??? Mesmo o método "Falar" pertencendo a mesma classe que o método "Pegar", não é por que estão no mesmo lugar que devem ser um só.

Então a divisão de responsabilidade, é mais ou menos isso, digamos que eu quero explicar para outra pessoa como se pega num "mouse". Eu vou usar o método "Falar" para explicar e chamarei no método "Falar" o meu método "Pegar" para fazer a demonstração de como pegar no "mouse" do computador.

Na programação, tem situações que são bem mais fechada, existem métodos que só existem para determinado método chamá-lo, para isso temos uma outra situação que temos que definir na hora de criá-los. Que são seu tipo de acesso. Para um método que só será usado por outro método daquela classe, o correto seria defini-lo como tipo de acesso "Private" e o mesmo não poderá ser chamado diretamente pela aplicação, só através do método que o terá a responsabilidade de usá-lo.

Bom illustrei o que aconteceu, tentei explicar o que é e para que serve a "Divisão de Responsabilidades" na Programação Orientada a Objetos.

São conceitos que valem a pena ser analisados e entendidos... mesmo que te chamem de maluco, o interessante é usá-lo, mas de forma correta, nada de excessos.