

Revisão do trabalho 1

Blog

Cauê Guerra
Cecilia Fernandes

26 de março de 2009

1 Análise

Excelentes documentos.

Nota: 10

2 Interface com usuário

O layout ficou excelente, apenas sentimos falta de botões de voltar e cancelar ações.

Nota: 9.5

3 Arquitetura

FrontController: não entendemos a motivação do método `executaAcao` - a idéia de separar as ações da parte de visualização está bem de acordo com o MVC, mas se o nome da URL é o parâmetro passado para o request dispatcher, a dependência se mantém. A configuração de página de erro poderia ser feita diretamente no `web.xml` com a tag `<error-page>`.

`BaseAdminController`: essa classe deveria ser um filtro que rege apenas sobre os usuários que tentam acessar lógicas de administradores. Simplificaria a implementação sem adicionar mais uma camada de classes abstratas.

`FabricaDeRepositorios`: o nome "Fábrica" foi utilizado de forma errônea. Tanto o padrão de projeto Fábrica como citado no GoF e sua versão que utiliza interfaces tem como idéia fornecer uma classe genérica que devolve um objeto com características específicas, de acordo com uma configuração.

`BaseRepositorioJdbc`: não entendemos a necessidade das verificações de conexão nula nessa classe. Se de fato for a idéia, como parece, de sempre que terminar de usar o código fechar a conexão, por que não abrí-la e fechá-la sempre em cada método do repositório? A ausência dos "recebe null" e "se é nulo..." deixaria o código mais elegante.

Ranking: não há necessidade da separação do pacote Ranking do restante dos repositórios, já que ele também se parece muito com um repositório - até lança `RepositoryException`. Também vale lembrar que repetir um mesmo método em dois lugares é um má idéia, já que dificulta a manutenção posterior, se necessária (método "rankeia").

Actions em geral: as actions de uma aplicação fazem parte do modelo, e não do controlador. Chamá-las de "AppController" dá uma falsa impressão da parte do código em que estamos trabalhando.

Nota: 5.5

4 Legibilidade e boas práticas

Os nomes da maioria dos métodos e classes estão bastante significativos, eliminando a necessidade de comentários. Parabéns.

As 23 classes com sufixo "AppController" são poluição desnecessária para quem vê os nomes das classes. Seria bastante mais econômico e visível tirar esse sufixo.

Algumas variáveis foram chamadas "b", "rs" e afins. Com as ferramentas de assistência de código, escrever nomes significativos, ainda que longos, não é problema. Dessa forma, a legibilidade do código aumenta consideravelmente.

Sentimos um certo abuso da palavra `static` no código:

- `BaseRepositoryJdbc`: para comportarem-se como constantes, basta o uso da palavra-chave `final`, o `static` não é necessário
- `FabricaDeRepositorios`: os métodos poderiam, sem perda de legibilidade, não serem estáticos.

O problema de inserir muitos `static` no seu código é dar uma cara procedural para um código que deveria ser orientado a objetos. Tente evitar essa prática no futuro! =)

`BaseRepositoryJdbc`: usar sempre o `PreparedStatement` é uma boa idéia, já que ele é a classe que surgiu para substituir o `Statement` e prevenir por padrão a velha e conhecida `SQL Injection`.

Nos modelos, o uso da classe `Date` é não recomendável em Java, já que esta foi uma primeira implementação e vem sendo depreciada com o tempo. Prefira usar `Calendar` - que é internacionalizável e tem métodos que facilitam lidar com as datas.

Nota: 6.5

5 Corretude

Uma página de erro poderia ter sido configurada. Levamos uma `stacktrace` de erro 500 ao tentar acessar a parte de login da parte de administração porque no `Readme` faltava a parte `"/admin"`.

De resto, está tudo de acordo.

Nota: 9.5

6 Funcionalidade

As funcionalidades atendem e até excedem às expectativas. Excelente produto final entregue.

Nota: 10

Final

Nota final em média simples: 8.5

Para quaisquer esclarecimentos e dúvidas, estamos muito abertos à discussão. Sabemos que a análise foi um pouco além do que o explicado na matéria, mas acreditamos que a avaliação foi bastante justa. Caso discordem das notas, não hesitem em falar conosco.