

MIGDeveloper (Desenvolvedor de Módulos de Interface Gráfica) XII INIC / VIII EPG - UNIVAP 2008

Fabiano Freitas Silva¹ e Roosevelt Alves da Silva²

¹⁻²

Universidade do Vale do Paraíba - Faculdade de Ciência da Computação
Av. Shishima Hifumi, 2911 CEP 12244-000 São José dos Campos - SP – Brasil

¹ff.javax@gmail.com ²roossilva@univap.br

Resumo - O desenvolvimento de softwares para controle de dados tem se tornado cada vez mais comum e também a sua freqüente necessidade de melhorias e mudanças, devido a constante evolução nas tecnologias que são utilizadas. Assim sendo, este artigo visa falar sobre o desenvolvimento de um programa para facilitar a criação e manutenção de tela de armazenamento de dados (arquivo de texto, banco de dados), utilizando uma interface amigável para criar os controles do sistema de forma declarativa (se preocupando somente com o que deve ser gerado e não como deve ser gerado), melhorando assim a velocidade no desenvolvimento, a escalabilidade e a confiabilidade do sistema. Para realizar este projeto foi utilizada a ferramenta de desenvolvimento NetBeans 6.0 e a linguagem de programação Java.

Palavras-chave: Extensibilidade, Escalabilidade.

Área do Conhecimento: Ciência da Computação

Introdução

Atualmente, é comum a necessidade de se criar programas para armazenamento de dados, desde arquivos de texto usados para análises científicas até grandes sistemas de gestão de pessoas e processos.

Junto com isso, alterações e melhorias nesses programas se tornam corriqueiras. Porém, se o programa não for bem estruturado e organizado, isso pode ser muito complicado.

Para amenizar este problema, o uso de frameworks, como o Hibernate^[1], se torna uma solução eficiente e eficaz, por criar um nível de abstração maior do programador com o sistema, fazendo com ele escreva menos código e conseqüentemente cometa menos erros.

Seguindo esta linha de pensamento, surgiu a idéia de desenvolver uma ferramenta que criasse programas de forma declarativa (sem precisar digitar nenhuma linha de código), onde você poderia criar diversas telas, informando seus campos, tipo de dados que cada uma teria e como eles seriam exibidos, além de definir que regras de validação eles deveriam ter. O projeto também deveria ser multiplataforma e que fosse fácil a sua extensibilidade, ou seja, a criação de novos objetos, controles, etc.

Metodologia

Para realizar este projeto foi utilizada a IDE NetBeans 6.0, para facilitar o desenvolvimento das telas do sistema na linguagem de programação JAVA. Todas as informações cadastradas pela IDE são salvas em arquivos XML, devido a sua

portabilidade e facilidade para o acesso dos dados.

A implementação do projeto se divide em duas partes:

- O ambiente de desenvolvimento, onde são cadastradas as telas do sistema;
- O interpretador do sistema gerado.

Na primeira, o usuário (administrador do software) define qual o tipo de projeto será gerado (MenuProject, onde as telas do sistema são organizadas por menu ou SimpleForm, sendo gerada apenas uma tela).

Após isso, deverão ser criadas as telas do sistema, onde você pode definir para cada campo a sua descrição, o tipo de objeto para exibição (uma caixa de texto, um caixa de combinações), o tipo de informação que irá conter (números, texto, data) junto com seu formato (01/01/08, 1/1/2008).

Além disso, é possível definir regras para controlar o uso dos campos, como a regra RequiredField, onde o campo associado a esta regra tem o seu preenchimento obrigatório, não permitindo que sejam gravadas as informações da tela enquanto o usuário não informar algum valor nele. Também é possível criar tipos de dados de usuário para serem utilizados em diferentes campos do sistema como, por exemplo, uma caixa de combo do tipo número com itens no intervalo de 0 a 50.

Terminando o cadastramento das telas, o usuário solicita a geração do projeto.

Após isso, começa a segunda parte do sistema que consiste em ler as informações do sistema gerado e exibir as suas telas. Esta é parte crítica do sistema, pois ela que irá controlar de fato o acesso do usuário final ao sistema, tendo que ter

bom desempenho e realizar corretamente as configurações definidas.

Resultados

Para testar a capacidade do sistema desenvolvido, foram selecionados alguns projetos existentes sendo eles o MIG-SSM[2] e uma interface para geração de arquivos usados por um projeto de simulação óssea, denominado Bones. Também foi criado um sistema de cadastro de vendas.

O primeiro possui várias telas onde são informados alguns dados, e cada uma delas gera um arquivo de texto. Depois, esses arquivos são concatenados, criando um arquivo que é utilizado por um executável chamado pela interface.

A criação das telas na ferramenta MIGDeveloper foi realizada facilmente, por serem muito parecidas e relativamente simples. Neste projeto foi possível a criação de tipo de dados de usuários, aplicação de regras de controle de campo e a utilização de um tipo de objeto chamado Procedure, onde foram cadastradas algumas expressões matemáticas semelhantes ao do projeto modelo.

O produto final acabou ficando melhor que o programa original, pois corrigiu alguns erros que ele possuía.

O segundo projeto gera um arquivo de texto contendo o endereço de n arquivos informados pelo usuário. Utilizando a ferramenta, foi criada uma tela contendo um objeto do tipo Grid, onde é adicionado cada um dos arquivos escolhidos para serem gravados.

O último consiste em construir um sistema para controle de vendas de uma loja. Primeiro, foi escolhido o tipo de banco de dados que seria usado, neste caso foi utilizado o SQL Server. Pela própria IDE foram criadas as tabelas de Clientes, Produtos, Vendas e ItensdaVenda, com seus atributos e relacionamentos de integridade.

Depois de criadas, foram selecionados alguns estilos de telas pré-definidas (Manutenção e Prompt de Consulta), e realizados pequenos ajustes (exclusão de alguns campos).

O resultado final foi à criação de uma tela para cada tabela de manutenção, onde você consegue incluir, alterar e excluir dados, utilizar botões de navegação (primeiro registro, próximo registro, etc) e um botão para chamar uma tela de lista, onde são consultados os dados de uma tabela e que retorna os dados de um registro selecionado.

Alguns recursos interessantes que foram criados, foi a junção das tabelas de Vendas e ItensdaVenda numa mesma tela de manutenção(tendo o cabeçalho montado pela tabela de Vendas e uma Grid com os dados da tabela ItensdaVenda), a inclusão da chamada de prompts para o campos que eram chave

estrangeira(tabelas de Cliente e Produto) e a criação de dois campos do tipo Fórmula, um para calcular o subtotal de cada produto(Quantidade X Preço) e o total da venda(somatório dos campos SubTotais) que eram executados sempre que a quantidade de produtos ou preço era alterado.

Discussão

Apesar da perda de desempenho e no uso de memória desta solução, a facilidade na criação e manutenção dos sistemas gerados, juntamente com sua confiabilidade, a torna uma alternativa interessante em diversas áreas.

Conclusão

As definições iniciais do projeto foram superadas, pois originalmente o projeto iria somente gravar informações em arquivos de texto, mostrando que a extensibilidade dos recursos desta ferramenta foram bem implementados.

Além disso, a parte do projeto para desenvolvimento das aplicações tornou-se eficiente e amigável, devido à criação de vários recursos para facilitar a criação de cada tipo de objeto e também pelo desempenho satisfatório obtido na interpretação das configurações do sistema.

Assim sendo, esta ferramenta pode ser facilmente integrada a outros sistemas, permitindo que não se perca muito tempo com a parte de criação de telas para controle de dados, permitindo que haja maior empenho nas partes inovadoras do sistema.

Pretensões Futuras

Como o requisito de tornar este projeto genérico foi realizado com sucesso, a continuação dele seria a criação de novos componentes (TabPage, CheckBox, PictureBox) e tipos de projetos (como um sistema para gerenciar processamento distribuídos e aplicações para web), mantendo a mesma facilidade de uso e performance na aplicação final.

Referências

- [1] www.hibernate.org. Acesso em 10 jun. 2008
- [2] MIG-SSM: Módulo de Interface Gráfica de um Sistema de Simulação Molecular – Aplicações com o Programa LMPROT. X Encontro Latino Americano de Iniciação Científica