

Projeto e Implementação de uma Ferramenta Gráfica para UID

Guilherme Schoepping

Bacharelado em Ciências da Computação, 2007

Departamento de Informática e Estatística

Universidade Federal de Santa Catarina (UFSC), Brasil, 88040-900

schoepping@gmail.com

Resumo

Este trabalho tem como propósitos o projeto e a implementação de uma ferramenta gráfica para a edição de diagramas de interação do usuário. Para tanto, foi realizado um estudo da notação destes diagramas, com o fim de familiarização com o domínio do projeto. Para auxiliar nos desafios técnicos se utilizou o Graphical Modeling Framework, uma plataforma direcionada para o desenvolvimento de editores gráficos. Com o suporte de um framework que compara os diversos métodos ágeis, se definiu um processo de desenvolvimento simples e rápido. O desenvolvimento, propriamente, foi feito de forma iterativa, com reuniões no final de cada iteração para a validação do que foi entregue.

Abstract

The purposes of this project are the design and the implementation of a graphical tool for user interaction diagram editing. For this, a study about these diagrams notation was done, seeking a better knowledge about the domain project. The Graphical Modeling Framework, a platform for graphical editors development, was used to help with the technical challenges. A framework that compares several agile methods was used to define a simple and fast development process. The development, properly, occurred in a iterative process, with meetings in the end of each iteration for system validation.

1. Introdução

Interação é uma denominação para a comunicação que ocorre entre um usuário e um sistema de computação (VILAIN, 2000). Através da modelagem da interação, as informações manipuladas pelo sistema, assim como as funcionalidades que este deve oferecer, podem ser descobertas. Estas informações e funcionalidades são de suma importância para o desenvolvimento do sistema, principalmente para o levantamento dos requisitos.

O Diagrama de Interação do Usuário, ou UID (User Interaction Diagram), é uma notação gráfica para representar a interação entre um usuário e um sistema. O UID surgiu da inexistência de uma técnica diagramática com alto nível de abstração voltado especificamente para a análise e modelagem de interação (VILAIN, 2002).

Este trabalho apresenta o projeto e a implementação de um software para edição de UIDs; esta edição deve ser realizada em um ambiente que proporcione facilidades para a sua sintaxe. O levantamento dos requisitos, assim como a análise do software, não foi necessário, pois se fez uso de um trabalho realizado anteriormente que tinha o mesmo escopo (SCHULTER, SPAGNOL, 2004). Foi necessária apenas uma revisão deste documento para se certificar da viabilidade do seu reaproveitamento considerando as tecnologias adotadas no projeto atual.

O projeto e a implementação foram realizados levando em consideração o uso de um framework que traz uma série de benefícios para o desenvolvimento de aplicações deste segmento: GMF (Graphical Modeling Framework). A linguagem de programação adotada foi Java, a mesma adotada no desenvolvimento do GMF; e o ambiente de desenvolvimento foi o Eclipse, o mesmo onde a aplicação desenvolvida funcionará como plug-in.

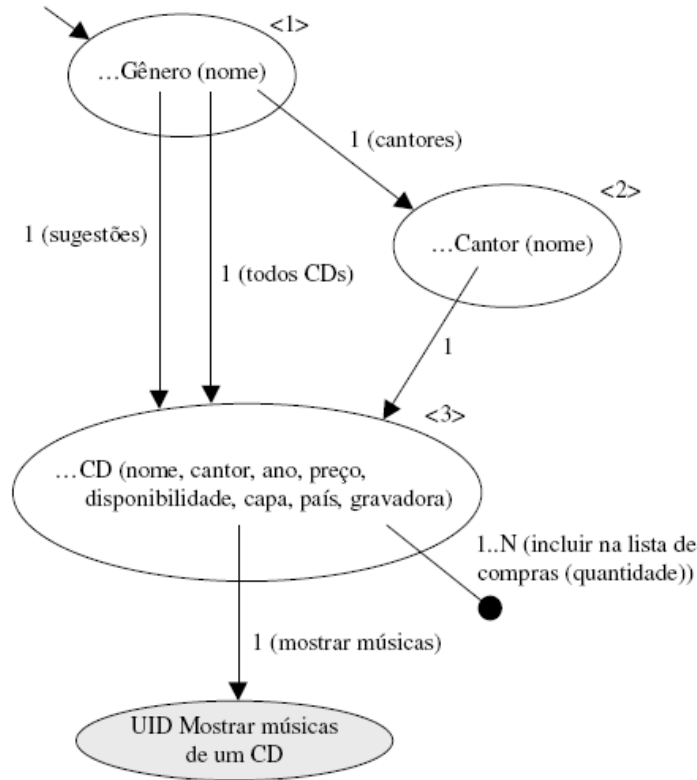
A motivação do trabalho surgiu da ausência de uma ferramenta específica para a manipulação de UIDs. Hoje, utilizar UIDs em projetos implica em recorrer a ferramentas gráficas genéricas, desperdiçando tempo com detalhes pertinentes à sintaxe em si e não ao seu conteúdo.

2. Diagramas de Interação do Usuário

O diagrama de interação do usuário (UID) é uma notação diagramática para representar a interação entre o usuário e o sistema. Este diagrama descreve apenas a troca de informação entre o sistema e o usuário, não considerando a interface do usuário e navegação.

É importante não esquecer que interações com vários usuários não são representadas, ou seja, apenas interações do sistema com um usuário.

A figura abaixo mostra um UID que representa a interação em uma tarefa de selecionar um CD baseado no tipo de música. No primeiro estado de interação, o sistema mostra uma série de tipos de música. A partir desse estado o usuário deve escolher apenas 1 tipo de música, denotado pelo 1 nas setas, seguido pela opção de artistas, todos os CDs e sugestões. Se o usuário selecionar a opção artistas, o diagrama segue para um estado de interação onde o sistema mostra uma variedade de artistas, dos quais deve ser selecionado apenas 1 para prosseguir. Ao selecionar o artista o sistema mostra os CDs do respectivo artista. Mas se o usuário selecionar uma das duas outras opções, todos os CDs e sugestões, o sistema mostrará uma lista com todos os CDs e os CDs recomendados pela loja, respectivamente. Para cada CD, o sistema mostra: título, nome do artista, ano, preço, disponibilidade, capa, país e gravadora. Se o usuário desejar, ele pode escolher um CD e visualizar suas músicas.



3. Graphical Modeling Framework

Antes do surgimento do Graphical Modeling Framework (GMF), desenvolver um editor gráfico costumava ser bastante complicado e exigir muito esforço. Mesmo com a existência de alguns frameworks que auxiliam o desenvolvimento de editores gráficos em Java. Entre eles, estão frameworks open source bastante interessantes como o JHotDraw e o Graphical Editing Framework (GEF). Mas enquanto estes fornecem sofisticadas ferramentas para o desenvolvimento gráfico, a parte que exige maior trabalho e cuidado, que é a modelagem do domínio e o mapeamento dos elementos gráficos, é deixada sob responsabilidade do desenvolvedor.

O framework utilizado, portanto, foi o GMF. Este é um projeto do Eclipse que provê uma plataforma de alto nível que faz uma conexão entre outros dois de seus projetos, o GEF e o EMF. Esta plataforma foi projetada para eliminar a necessidade de o desenvolvedor gerenciar a persistência do modelo e deixá-lo focar na lógica de negócio do editor, eliminando os problemas encontrados ao usar o GEF.

4. Processo de Desenvolvimento

Antes de começar o projeto da ferramenta, foi definido um processo para o seu desenvolvimento, garantindo, assim, produtividade e qualidade durante o projeto.

Pelo fato de trabalhar com tecnologias recentes, como o GMF, e que não se conhece totalmente o domínio destas tecnologias, e também com objetivo de mostrar as evoluções do desenvolvimento de tempos em tempos para evitar discrepâncias no final, buscou-se um processo iterativo. Um processo iterativo consiste em desenvolver o software em partes, o que ajuda a identificar mais cedo os problemas que possam surgir. Assim, a fim de construir um processo simples, rápido e sem burocracia optou-se por utilizar um processo com características dos métodos ágeis.

Nos últimos anos os métodos ágeis têm despertado bastante interesse na comunidade de engenharia de software, o que resultou no surgimento de vários métodos. A existência de vários métodos ágeis dificulta um estudo individual de cada um a fim de escolher um processo de desenvolvimento. Para facilitar a escolha de um processo ágil a ser utilizado neste trabalho, buscou-se auxílio no framework de comparação e análise de diversos métodos ágeis (FAGUNDES, 2005), que será utilizado durante o processo de desenvolvimento do software. Este framework realiza um estudo dos métodos ágeis Extreme Programming (XP), Scrum, Feature Driven Development (FDD), Adaptive Software Development (ASD), Dynamic System Development Method (DSDM), Crystal Clear, Lean Software Development (LSD) e Agile Modeling (AM), apresentando as características de cada método, seguido de uma análise das atividades propostas. Como resultado do estudo é proposto um framework para métodos ágeis que reúne atividades dos métodos estudados, visando facilitar a adoção de um processo ágil de desenvolvimento. A grande vantagem da utilização deste framework é a possibilidade de compor um processo com atividades de diferentes métodos e que se adapte às necessidades do projeto.

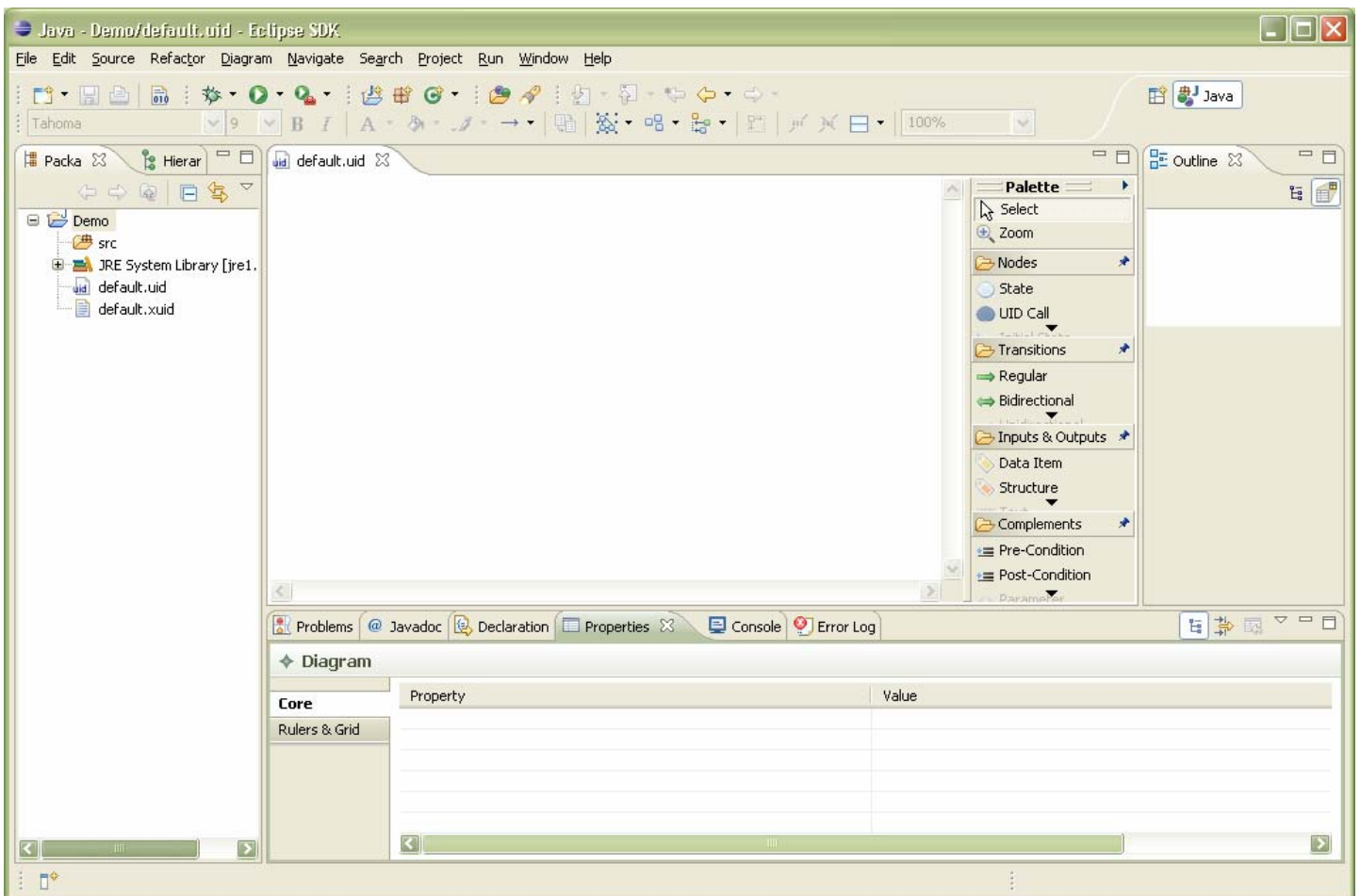
O framework de comparação e análise dos métodos ágeis faz um agrupamento das diversas atividades, dos diferentes métodos, a fim de evitar redundâncias. Estas atividades são agrupadas de acordo com a estrutura sugerida pelo desenvolvimento incremental. É importante ressaltar que nem todas as atividades apresentadas pelo framework fazem parte do processo definido para o desenvolvimento desta ferramenta. As atividades que fazem parte do presente processo são:

- Atividades de Definição dos Requisitos;
- Atividades de Atribuição dos Requisitos às Iterações;
- Atividades de Projeto da Arquitetura do Sistema;
- Atividades de Desenvolvimento do Incremento do Sistema;
- Atividades de Validação do Sistema;
- Atividades de Entrega Final.

No desenvolvimento desta ferramenta gráfica foi definido um total de três iterações com duração de aproximadamente vinte dias cada.

5. Ferramenta

A ferramenta gráfica desenvolvida deverá ser utilizada como plugin do ambiente de desenvolvimento Eclipse, desta forma tira-se proveito de toda a infra-estrutura do ambiente de desenvolvimento para suportar os novos editores feitos neste trabalho. A figura abaixo ilustra um dos editores e também todos os outros componentes que são aproveitados para compor a interface gráfica. Esta forma de distribuição oferece grande facilidade para a divulgação e uso da ferramenta, visto que este ambiente de desenvolvimento é amplamente utilizado.



6. Considerações Finais

A ferramenta se encontra em funcionamento, logo, usuários poderão usá-la para a manipulação de UIDs. E receberão suporte para todos os elementos da notação UID, assim como diversas vantagens que uma ferramenta específica de edição proporciona. Esta ferramenta, assim como seu código fonte e sua documentação, está disponível em: <http://code.google.com/p/uieditor/>

Sabia-se que o desenvolvimento de uma ferramenta gráfica era concreto, ou seja, não era nada inovador e não tinha um domínio completamente desconhecido. O grande problema, entretanto, era a complexidade envolvida para a concretização de uma ferramenta destas. Em um primeiro momento foi sugerido o uso de um framework bastante sólido e conhecido: GEF (Graphical Editing Framework). Buscou-se saber mais sobre este framework, suas vantagens e desvantagens, com pessoas que já o utilizaram para desenvolvimento e se concluiu que seu uso apresentava certa complexidade. A busca por outras tecnologias não tinha sido concluída quando se chegou ao framework adotado neste trabalho: GMF (Graphical Modeling Framework).

O GMF, assim como o GEF, é um projeto do grupo Eclipse e ainda é bastante recente. A versão utilizada neste trabalho foi lançada em junho deste ano (2007). Sua curta história implica, conseqüentemente, em uma comunidade pequena de desenvolvedores e uma limitada biblioteca de exemplos que acabam dificultando os passos dos desenvolvedores que desejam adotá-lo. Sua pouca maturidade, porém, é compensada com as diversas e poderosas facilidades que este oferece. A mais expressiva das vantagens é a possibilidade de criar uma ferramenta gráfica com quase nenhuma linha de código.

O verdadeiro sucesso deste trabalho se concretizará quando a ferramenta for utilizada, trazendo benefícios e atendendo as necessidades dos desenvolvedores que vierem a utilizar UIDs em seu processo de desenvolvimento.

Referências Bibliográficas

ECLIPSE CON, 2006. **Introduction to the Eclipse Graphical Modeling Framework**. Disponível em: <<http://eclipsezilla.eclipsecon.org/php-/attachment.php?bugid=1>>. Acesso em: 4 jun. 2007.

FAGUNDES, Priscila B. **Framework para comparação e análise de métodos ágeis**. Tese de Doutorado, Universidade Federal de Santa Catarina, Florianópolis, 2005.

SITE OFICIAL GRAPHICAL MODELING FRAMEWORK. Disponível em: <www.eclipse.org/gmf/> Acesso em: 21 mai. 2007.

VILAIN, Patrícia. **Modelagem da interação com o usuário em aplicações hipermídia**. Tese de Doutorado, PUC-Rio, Rio de Janeiro, 2002.

VILAIN, Patrícia; SCHWABE, Daniel; SOUZA, Clarisse Sieckenius de. **A Diagrammatic Tool for Representing User Interaction in UML**. Proceedings of the UML 2000 Conference. Lecture Notes in Computer Science, Vol. 1779, Springer-Verlag, Berlin, 2000, 133-147.