

UNIVERSIDADE FEDERAL DE PERNAMBUCO

**Graduação em Ciência da Computação
Centro de Informática**

**UMA FERRAMENTA PARA SUPORTE À CRIAÇÃO
DE UM BENCHMARKING FRAMEWORK**

Aluno: Igor César Dourado

Orientador: Profº. Sérgio Castelo Branco Soares

Recife 2011

UNIVERSIDADE FEDERAL DE PERNAMBUCO

**UMA FERRAMENTA PARA SUPORTE À CRIAÇÃO
DE UM BENCHMARKING FRAMEWORK**

Apresentação de monografia à Universidade
Federal de Pernambuco como requisito para concluir
o curso de graduação em Ciência da Computação.

"Ninguém é mais escravo do que aquele que se julga livre sem o ser."
(Goethe)

ABSTRACT

In the activity of planning an application development, one of the required decisions is choosing technologies and techniques to be used. On the other hand, researchers are creating new technologies and techniques and it is necessary to evaluate them, providing data for decision makers. This work aims to support creating and using a set of specific benchmarking applications that can help researchers to perform evaluation studies. Benchmarking techniques for software projects have a high potential of improving evaluations. The concept of a benchmarking framework (a comparison environment) used to support evaluation studies can help technology transfer by making more evidences available. This work proposes a support tool for a general process of creating a benchmark framework. As a proof of concept, the implemented tool was used to create an Aspect-Oriented maintainability benchmark framework. The tool considers the need of characterize applications of a specific domain, to register such applications and to search for them while designing evaluation studies.

Key-words: benchmarking, benchmarking framework, aspect-oriented programming, maintainability, empirical studies.

RESUMO

Ao se planejar o desenvolvimento de uma aplicação, uma das tomadas de decisão necessárias corresponde às tecnologias e técnicas a serem utilizadas. Para tal, pesquisadores estão criando novas tecnologias e técnicas que necessitam ser avaliadas, a fim de que possam prover dados para tomadas de decisão.

Este trabalho visa oferecer um suporte à criação e ao uso de um conjunto de aplicações benchmarking que podem ajudar pesquisadores a realizar estudos de avaliação, já que técnicas de benchmarking para projetos de software tem um alto potencial de melhoramento dos mesmos. O conceito de benchmarking framework (ambiente de comparação) usado para suporte a estudos de avaliação podem estudar na transferência de tecnologia através da disponibilização de mais evidências.

Este trabalho propõe uma ferramenta de apoio para um processo genérico de criação de um benchmarking framework. Como prova do conceito, a ferramenta implementada foi usada na criação de um benchmark framework sobre manutenibilidade de aplicações OA (orientadas a aspectos). A ferramenta considera a necessidade de se caracterizar aplicações de um domínio específico, a fim de se registrar essas aplicações e de se realizar buscas sobre as mesmas durante a criação de estudos de avaliação.

Palavras-chave: *benchmarking, benchmarking framework*, orientação a aspectos, manutenibilidade, estudos empíricos.

SUMÁRIO

ABSTRACT	I
RESUMO	II
SUMÁRIO	III
ÍNDICE DE FIGURAS	IV
ÍNDICE DE TABELAS	Erro! Indicador não definido.
GLOSSÁRIO	VI
1. CAPÍTULO I - INTRODUÇÃO	1
1.1. Contexto do problema.....	1
1.2. Benchmarking Framework	1
1.3. Motivação da ferramenta	2
1.4. Organização do Trabalho	3
2. CAPÍTULO II – FUNDAMENTAÇÃO E OBJETIVOS	4
2.1. Aplicações de BF em engenharia de software.....	4
2.2. A instância de BF proposta.....	5
2.2.1. Estrutura do Processo	6
2.3. Suporte da ferramenta ao processo.....	8
3. CAPÍTULO III – CRIAÇÃO DA FERRAMENTA	10
3.1. Tecnologias utilizadas.....	10
3.1.1. Tecnologia para a geração de BFs	10
3.2. Definição das funcionalidades	11
3.2.1. Projetar Benchmarking Framework.....	11
3.2.1.1. Gerenciamento dos BFs	12
3.2.1.2. Definição de atributos básicos	12
3.2.1.3. Definição da estrutura do BF	13
3.2.2. Cadastrar Benchmarks	15
3.2.3. Pesquisar Itens	16
3.2.4. Permissões de acesso.....	16
3.3. Representação do BF (estrutura interna).....	17
4. CAPÍTULO IV – EXEMPLO DE USO DA FERRAMENTA	20
4.1. Criação de um BF	20
4.2. Cadastro de Benchmarks	27
5. CAPÍTULO V – CONCLUSÕES.....	32
5.1. Avaliação do uso da ferramenta	32
5.2. Trabalhos futuros	33
Bibliografia.....	34

ÍNDICE DE FIGURAS

Figura 1 - Estrutura do Processo de utilização de um BF Definido por Moura	6
Figura 2 - Gerenciamento dos BFs.....	12
Figura 3 - Edição dos atributos Nome e Objetivo	13
Figura 4 - Estrutura da DSL.....	14
Figura 5 - Tela de edição da estrutura do BF	15
Figura 6 - Tela de pesquisa de componentes.....	16
Figura 7 - Modelo Conceitual.....	18
Figura 8 - Modelo Lógico.....	19
Figura 9 - Tela de criação de um BF	20
Figura 10 - Tela de listagem dos BFs existentes.....	21
Figura 11 - Componentes Produto e Cenários de Manutenção adicionados	22
Figura 12 - Edição de componente.....	23
Figura 13 - Tela final da criação do BF Moura.....	26
Figura 14 - Tela Cadastrar Benchmarks.....	28

ÍNDICE DE TABELAS

Tabela 2.1 - Atributos Gerais do Produto (1 p. 38).....	7
Tabela 2.2 - Atributos Orientados a Aspecto do Produto (1 p. 45).....	7
Tabela 2.3 - Atributos do Cenário de Manutenção (1 p. 50)	8
Tabela 4.4 - Atributos Gerais do Produto com tipos e regras de quantidade....	24
Tabela 4.5 - Atributos Orientados a Aspecto do Produto com tipos e regras de quantidade	24
Tabela 4.6 - Atributos do Cenário de Manutenção com tipos e regras de quantidade	25
Tabela 4.7 - Presença dos Atributos OA na aplicação HealthWatcher	29
Tabela 4.8 - Atributos gerais do HealthWatcher preenchidos	29
Tabela 4.9 - Análise do Tipo de Manutenção para os Cenários do HW.....	30

GLOSSÁRIO

Sigla	Descrição
OA	Orientação a aspectos
BF	<i>Benchmarking Framework</i>
DSL	<i>Domain Specific Language</i>

1. CAPÍTULO I - INTRODUÇÃO

Neste capítulo são introduzidos o contexto do problema abordado, os conceitos fundamentais envolvidos e a motivação da criação da ferramenta.

1.1. Contexto do problema

O problema abordado por este trabalho tem um caráter genérico, porém é abordado um caso específico para estudo da solução proposta. O foco é um projeto de uma ferramenta que, por meio de uma DSL visual, ofereça suporte para processos de estudos empíricos sobre itens de alguma área específica. A facilitação de estudos sistemáticos, em especial experimentos, é necessária para a avaliação de atividades (produtos, itens) realizadas por pesquisadores, independentemente da área de estudo (1 p. 13).

A solução desenvolvida busca um meio de especificar as características das aplicações que serão utilizadas em estudos empíricos, as quais foram anteriormente definidas pelo trabalho de Moura (1). A solução deve permitir, além da especificação das características a serem informadas, o cadastro e a pesquisa destas aplicações cadastradas.

Dado seu caráter genérico, aplicável a diversos contextos, a ferramenta tem sua utilidade melhor compreendida através de sua aplicação prática. Com a definição do aplicativo e o exemplo de seu uso mostrado neste trabalho é possível entender não só o benefício de seu uso para o caso, mas também como seu caráter genérico não fica restrito ao exemplo estudado.

1.2. Benchmarking Framework

Um benchmark pode ser caracterizado como uma forma de encapsular informações e procedimentos (hipóteses, tratamentos, variáveis independentes, objeto de controle, objeto do estudo, resultados e demais componentes de um experimento) utilizados em estudos empíricos, a fim de se obter respostas relacionadas a algum aspecto comum a esses itens.

O conceito de Benchmarking Framework (BF) está ligado a um conjunto de regras e atributos particulares que devem caracterizar aplicações que serão disponibilizadas para avaliação de processos, técnicas e tecnologias em geral. Portanto, o BF ajuda a sistematizar estudos sobre os itens listados relacionados a uma determinada área, a fim de guiar pesquisadores em estudos empíricos. No caso deste trabalho, os itens a serem estudados serão aplicações de software com características de orientação a aspectos, a serem descritas e avaliadas em relação à manutenibilidade para esse tipo de aplicações. Também existem diversos outros casos bem sucedidos da aplicação desse conceito para estudos empíricos sobre software (2).

1.3. Motivação da ferramenta

Além da própria definição das características de um BF que apoiarão um estudo, existe a necessidade de um apoio ao processo. Afinal, este envolve diversos dados (de vários tipos e associados a hierarquias) que caracterizam os itens a serem estudados. Em um estudo empírico se faz necessária também a comparação de benchmarks para a avaliação de algum aspecto comum, como descrito na Seção 1.2.

A ferramenta proposta neste trabalho objetiva auxiliar a execução desses processos através da definição e atualização de BFs e do cadastro, atualização e pesquisa de aplicações no BF de forma estruturada e organizada.

Logo, uma vez definida a estrutura de um BF que fornecerá apoio a um estudo, um especialista poderá visualizar as aplicações cadastradas no mesmo a fim de avaliar potenciais itens representativos (benchmarks) e poderá cadastrá-los. Assim o BF estará associado a seus benchmarks, o que possibilita a pesquisa automática sobre esses dados, algo útil na execução de estudos empíricos.

1.4. Organização do Trabalho

O Capítulo 2 do trabalho descreve o contexto do uso de BFs em software em função de suas particularidades, especificando o conceito introduzido no Capítulo 1. Logo é mostrado o processo que serviu como base para o estudo da elaboração da ferramenta, com o detalhamento da etapa que a ferramenta proposta apóia, assim como dos dados de entrada e saída envolvidos.

No Capítulo 3 é detalhado o projeto da ferramenta em si: são explicados cada um dos principais casos de uso do projeto, que correspondem a suas principais funcionalidades. O fluxo dos mesmos já é apresentado em nível das telas do sistema. Logo, é explicado o projeto da DSL visual (sua estrutura e arquitetura), principal componente da ferramenta, responsável por permitir a criação dos BFs.

O Capítulo 4 demonstra um exemplo de uso da ferramenta: os dados do processo do BF descrito no capítulo 2 são utilizados a fim de se mostrar um exemplo de uso no apoio de um estudo.

O capítulo 5 conclui o trabalho através de uma avaliação da ferramenta criada, do seu apoio ao processo e apresenta sugestões de melhoria e expansão do projeto para trabalhos futuros.

2. CAPÍTULO II – FUNDAMENTAÇÃO E OBJETIVOS

Este capítulo explica a fundamentação da ferramenta, ou seja, detalha o problema que a mesma se propõe a solucionar. A partir disso, é justificado como seus objetivos são alcançados.

2.1. Aplicações de BF em engenharia de software

As aplicações de BF para a produção de software estão em uma fase inicial de seu estudo. Há diversos estudos de aplicações da técnica, porém são em sua maioria casos específicos de estudo em seu respectivo contexto e domínio.

Como citado no capítulo anterior, o suporte de BF para estudos empíricos é algo bastante genérico. Seu conceito tem potencial para ser aplicado em outras áreas que necessitem da sistematização e retroalimentação de processos. Afinal, a organização e atualização dessas informações podem ajudar no amadurecimento de tais processos através do acúmulo de conhecimento obtido empiricamente (1 p. 16).

Portanto, sua aplicação no contexto de software tem diversas particularidades. A abordagem dos estudos a serem realizados depende principalmente das tecnologias, do conhecimento especialista e do domínio das aplicações a serem estudadas.

Um exemplo claro da aplicabilidade desses estudos é o suporte a tomadas de decisões em organizações, quando surge a necessidade de se definir que caminhos tomar ao se planejar o desenvolvimento de um software. Obviamente, os gestores possuem o suporte de especialistas na empresa. Existem diversos modelos de gestão e de qualidade que definem um processo para esse estudo a serem executados com a ajuda de engenheiros de software, que devem possuir conhecimento e experiência do uso de diversos paradigmas, tecnologias, padrões de projeto etc. Todas essas características e técnicas de um projeto de software devem ser levadas em consideração para se tentar “prever” questões de complexidade e comportamento do sistema.

Porém, mesmo quando tomadas por profissionais capacitados, essas decisões podem ser otimizadas com o devido apoio ferramental. Pois por mais que o engenheiro de software seja experiente, o conhecimento acumulado aplicado à constante evolução de uma ferramenta que esteja direcionada para o estudo que esse profissional necessita realizar sempre poderá ajudá-lo.

O acúmulo do conhecimento resultante da experiência de vários profissionais deve ser realizado através de mecanismos que possibilitem tanto a sistematização como a retroalimentação dos processos a serem realizados por estudos empíricos.

Assim, em um caso hipotético onde um engenheiro necessite analisar o desempenho de aplicações orientadas a objetos com muitos casos de concorrência, o mesmo poderia comparar o seu projeto com outros semelhantes já realizados, e portanto conhecer melhor as características que influenciariam o desempenho do seu próprio projeto.

2.2. A instância de BF proposta

Para este trabalho de conclusão de curso foi escolhida uma instância específica do BF (1 p. 30). A mesma foi definida com os objetivos particulares de apoiar estudos sobre manutenibilidade de aplicações orientadas a aspectos (OA). Porém, na sua estrutura, o BF é genérico e pode ser utilizado para outros domínios, visando:

- Auxiliar a avaliação de técnicas de desenvolvimento de software;
- Apoiar a elaboração, replicação e avaliação de estudos empíricos;
- Facilitar a identificação de aplicações e cenários de mudança considerados como benchmarks;
- Estender dos critérios para atender a estudos de aplicações de diferentes contextos.

Em outras palavras, o BF é definido para dar suporte a estudos e a replicação de estudos sobre a manutenibilidade de softwares orientados a

aspectos. Logo, seu objetivo é definir características específicas das aplicações OA envolvidas nestes estudos para que as mesmas possam ser sistematicamente avaliadas e comparadas em relação ao contexto de manutenibilidade.

2.2.1. Estrutura do Processo

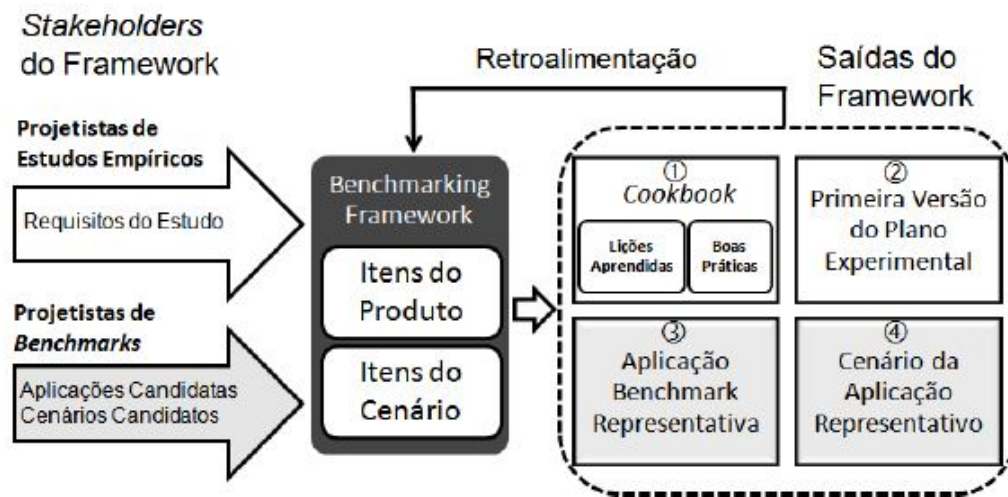


Figura 1 - Estrutura do Processo de utilização de um BF Definido por Moura (1 p. 34)

O processo define dois atores (*stakeholders*) que podem interagir com o BF, sendo eles:

- **Projetista de Estudos Empíricos:**
É responsável pela definição do estudo. Possui conhecimento suficiente sobre o domínio para definir a estrutura e as regras que irão caracterizar um estudo empírico sobre manutenibilidade de software com técnicas de OA. Este ator é ainda responsável pela especificação (definição) da instância do BF, ou seja, da lista de itens que deverão compor o BF. Tal lista define características que serão armazenadas sobre as aplicações e os cenários de evolução desta instância do BF.

- **Projetista de Benchmarks:**
É responsável pelo gerenciamento (cadastro e atualização) das aplicações benchmarks. Este usuário irá então selecionar aplicações e seus cenários para cadastrá-las.

O BF em si é o conjunto de Itens de Produto (vide Tabela 2.1 e Tabela 2.2) e Itens de Cenário (vide Tabela 2.3).

Tabela 2.1 - Atributos Gerais do Produto (1 p. 38)

Atributos Gerais	
Identificação do Sistema	Nome
	Domínio
Packaging	Detalhes da Versão
	Disponibilidade de Código Fonte
	Linguagens de Programação
	Sistema Operacional e Requisitos Mínimos
Documentação do Ciclo de Vida	
Técnicas de Desenvolvimento	

Tabela 2.2 - Atributos Orientados a Aspecto do Produto (1 p. 45)

Atributos Orientados a Aspectos	
Classificação de Interesses Transversais	Funcional e Não Funcional
	Homogêneo e Heterogêneo
	Intra-Componente e Inter-Componente
Interação e Composição de Interesses Transversais	Baseado em Invocação
	Entrelaçamento no Nível de Componentes
	Entrelaçamento no Nível de Operações
	Sobreposição
Escopo de Interesses Transversais	
Construções de Linguagens AO	

Tabela 2.3 - Atributos do Cenário de Manutenção (1 p. 50)

Atributos do Cenário de Manutenção		
Descrição do Cenário		
Tipo de Mudança	Objetivo da Mudança	Manutenção Corretiva
		Manutenção Adaptativa
		Manutenção de Aperfeiçoamento
	Natureza da Mudança	Estrutural
		Comportamental
Nível das Mudanças	Requisitos	
	Análise e Projeto	
	Implementação	

Os itens “Aplicação Benchmark Representativa” e “Cenário da Aplicação Representativo” são avaliados pelo Projetista de Benchmarks como representativos para os itens do produto (aplicações OA) e os itens do cenário (atividades de manutenção sobre uma aplicação) definidos no BF, de acordo com o explicado para o usuário Projetista de Benchmarks. Na prática, os itens do produto são aplicações AO que serão avaliadas, e os cenários são atividades de manutenção sobre essas aplicações.

Os itens “Cookbook” e “Primeira Versão do Plano Experimental” não entram no escopo definido para a ferramenta proposta e não serão detalhados neste trabalho. Porém todos os itens da Figura 1 devem ser levados em consideração para a devida retroalimentação do BF. Na prática, a retroalimentação significa a melhoria do processo de apoio a um estudo com determinados objetivos.

2.3. Suporte da ferramenta ao processo

A ferramenta propõe um apoio à definição dos itens do processo de Moura (1) através da criação e manipulação dos dados correspondentes às Tabelas 2.1, 2.2 e 2.3 (atividade relacionada ao Projetista de Estudos Empíricos). Logo, através do uso de uma estrutura criada a partir dos itens citados, o Projetista de Benchmarks pode cadastrar Produtos e Cenários resultantes avaliados pelo mesmo como benchmarks.

Também foi levada em consideração a necessidade de pesquisas sobre os benchmarks cadastrados, funcionalidade esta além do escopo do processo de Moura.

3. CAPÍTULO III – CRIAÇÃO DA FERRAMENTA

Neste capítulo são justificadas as tecnologias, seguidas dos detalhes de projeto da criação da ferramenta através da explicação de suas funcionalidades.

3.1. Tecnologias utilizadas

Para o desenvolvimento da ferramenta foram utilizadas tecnologias web gratuitas, assim como um servidor de banco de dados para a persistência de dados.

A opção por uma tecnologia com suporte web foi escolhida devido à possibilidade de acesso facilitado a partir de qualquer ponto, já que a intenção é que a aplicação possa ser acessada por diversos usuários não necessariamente em um mesmo local. Outro ponto fundamental é que fosse possível a edição de uma interface gráfica rica e dinâmica, de modo a representar a criação de BFs de forma clara e intuitiva.

Foi escolhida a plataforma ASP.NET, que possibilita a criação de páginas ricas mesmo utilizando-se uma versão gratuita do Visual Web Developer 2010 (Visual Web Developer 2010 Express). O banco de dados utilizado foi o SQL Server 2008, também em sua versão gratuita.

3.1.1. Tecnologia para a geração de BFs

Para possibilitar a criação de BFs foi cogitado o uso de tecnologias específicas para a criação de DSLs (*Domain Specific Languages*), já que temos um caso de propósito específico para a solução de um problema que se propõe a ser reutilizado diversas vezes (3). Porém, devido ao caráter genérico das possibilidades de construção da estrutura, foi suficiente a representação dos componentes da linguagem através de objetos (em C#). Na Seção 3.2.1.3 é detalhada essa estrutura, que basicamente permite a criação de componentes

que podem possuir outros componentes e uma série de atributos que podem estar associados a estes.

3.2. Definição das funcionalidades

A ferramenta possui quatro funcionalidades principais que correspondem aos principais casos de uso identificados. As duas primeiras funcionalidades foram derivadas dos dois papéis definidos do BF proposto por Moura (1): o Projetista de Estudos Empíricos e o Projetista de Benchmarks; a terceira corresponde ao usuário que deseja realizar pesquisas por componentes; a quarta e última diz respeito às permissões dos usuários segundo seu uso da ferramenta.

O objetivo deste capítulo é explorar cada funcionalidade, de forma a apresentar de forma clara sua construção e utilidade.

3.2.1. Projetar Benchmarking Framework

Esta funcionalidade representa o caso de uso relativo ao Projetista de Estudos Empíricos. Trata-se do processo de criação do BF, incluindo a definição da estrutura, das regras de restrição e dos tipos de dados associados aos componentes.

O processo de criação foi definido de tal forma que seja possível representar todos os itens da etapa de Benchmarking Framework proposto, a fim de oferecer um suporte à execução de um estudo empírico. Porém, apesar de tal processo servir como referência, o processo de criação definido possui características suficientemente genéricas para dar suporte a estudos de contextos totalmente diferentes, não se prendendo ao domínio de manutenibilidade de softwares OA.

3.2.1.1. Gerenciamento dos BFs

A primeira tela (Figura 2) da opção “Projetar BFs” corresponde ao gerenciamento geral dos BFs. É exibida uma tabela com os BFs previamente cadastrados, incluindo sua breve descrição (Nome e Objetivo). O usuário poderá visualizar detalhes de BFs existentes, deletá-los, editá-los, ou criar um novo BF.

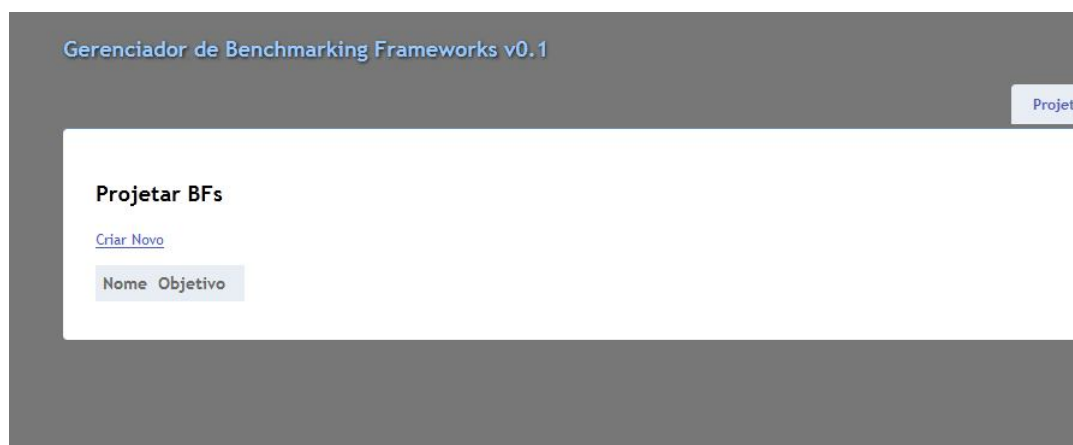


Figura 2 - Gerenciamento dos BFs.

3.2.1.2. Definição de atributos básicos

Na primeira tela de edição ou criação de BFs (Figura 3), o usuário pode definir os atributos de Nome e Objetivo. São apenas critérios básicos de descrição de um BF. Logo, o usuário pode passar à próxima etapa, correspondente à edição da estrutura do BF.

Gerenciador de Benchmarking Frameworks v0.1

Projeta

Criar

BF

Nome
Meu BF

Objetivo
Exemplo

Criar

[Voltar](#)

Figura 3 - Edição dos atributos Nome e Objetivo

3.2.1.3. Definição da estrutura do BF

Esta etapa corresponde à criação/edição do BF em si. Na parte esquerda da tela, é exibida a estrutura de forma a se visualizar claramente a hierarquia dos componentes. O usuário pode então tanto criar como editar componentes já existentes.

Essa criação, como citado no início do tópico 3.2.1, depende da definição da própria estrutura, regras de restrição e tipos de dados associados aos componentes. A estrutura por sua vez é uma hierarquia de componentes, que poderá estar associadas a estas regras de restrições. Cada componente, além de poder “possuir” outros componentes, também pode possuir tipos de dados determinados pelo usuário.

Essas possibilidades de definição para o BF constituem as regras de uma DSL externa, modelada especialmente para esse caso. Apesar de o modelo de criação para BFs proposto objetivar o suporte a estudos empíricos em geral, pode-se considerar como domínio dessa linguagem a própria criação de BFs. Portanto, não se trata da aplicação do caso geral de uma DSL, que envolve domínio específico com diversas particularidades.

A linguagem definida tem caráter declarativo, e suas possibilidades de uso estão disponíveis em uma interface gráfica para o usuário. Abaixo é mostrada uma figura com sua estrutura e regras, seguida de uma explicação mais detalhada dessas características.

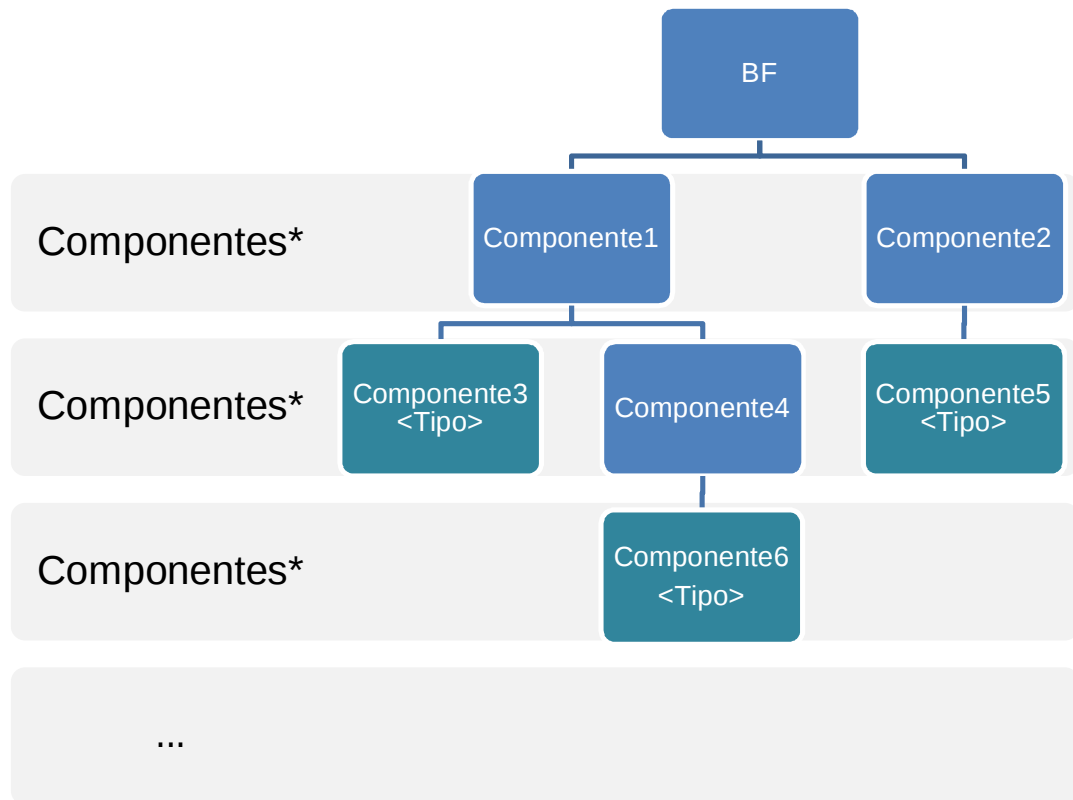


Figura 4 - Estrutura da DSL

- A estrutura pode possuir uma altura n qualquer;
- Os componentes finais podem estar em alturas arbitrárias;
- O usuário define quais são os filhos de todos os componentes, assim como seus tipos que podem ser:
 - o Texto
 - o Opções de Texto
 - o Numérico
 - o Arquivos
 - o Booleano

- Cada componente final pode conter n elementos do tipo definido, com n a ser definido pelo usuário;

A tela de definição da estrutura disponibiliza graficamente a estrutura descrita (Figura 5), de forma que o usuário possa adicionar, remover e editar os componentes citados, assim como definir sua hierarquia.

Gerenciador de Benchmarking Frameworks v0.1

Atributos Gerais

[Criar Novo](#)

Componente	Texto	Opções	Numérico	Arquivos	Booleano	
Atributos Gerais						Editar Detalhes Deletar

Salvar

Figura 5 - Tela de edição da estrutura do BF

3.2.2. Cadastrar Benchmarks

Esta funcionalidade é define o caso de uso relativo ao Projetista Benchmarks. Aqui o usuário irá escolher um BF previamente projetado para adicionar itens representativos aos componentes. A finalidade é oferecer suporte ao projetista que estuda componentes (como aplicações ou cenários) que sirvam de benchmarks (casos representativos) para estudos empíricos sobre o BF escolhido.

É disponibilizada ao usuário uma visualização semelhante à interface gráfica da funcionalidade Projetar Benchmarking Framework, porém aqui não é

possível editar a estrutura ou componentes. Neste caso o usuário pode adicionar dados/arquivos de acordo com tipos definidos na estrutura, podendo ser textos, valores numéricos/booleanos ou arquivos. A limitação da quantidade desses dados a serem associados a um componente é definida na fase de projeto do BF.

3.2.3. Pesquisar Itens

A pesquisa de componentes (Figura 6) se refere a um usuário não explicitado no processo definido estudado. Trata-se do caso de uso onde o usuário pode procurar por itens cadastrados associados a componentes de BFs. Nas opções de busca é possível buscar atributos pelo nome e seu tipo.



Gerenciador de Benchmarking Frameworks v0.1

Projetar I

Pesquisar Componentes

Nome do componente:

qualquer tipo

BF Componente Texto Opções Numérico Arquivos Booleano

Figura 6 - Tela de pesquisa de componentes

3.2.4. Permissões de acesso.

Há um gerenciamento de permissões de acesso às funcionalidades descritas com o objetivo separar as visões do sistema de acordo com cada interesse. Logo, são três tipos de usuários definidos:

- Projetista de Estudos Empíricos: possui acesso à funcionalidade Projetar Benchmarking Framework;

- Projetista de Benchmarks: possui acesso à funcionalidade Cadastrar Benchmarks;
- Administrador: possui acesso a todas as funcionalidades, assim como ao gerenciamento de usuários;

A terceira funcionalidade (Pesquisar Itens) não possui controle de acesso e pode ser utilizada sem necessidade de login.

3.3. Representação do BF (estrutura interna)

Para representar a estrutura do BF descrita no tópico anterior foi definida uma arquitetura de dados de caráter genérico. Com a mesma, é possível representar diversos BFs com estruturas livres, de acordo com as limitações descritas na Seção 3.2.1.3.

A Figura 7 mostra a arquitetura que descreve essa estrutura, através do modelo de dados conceitual e lógico utilizados na implementação.

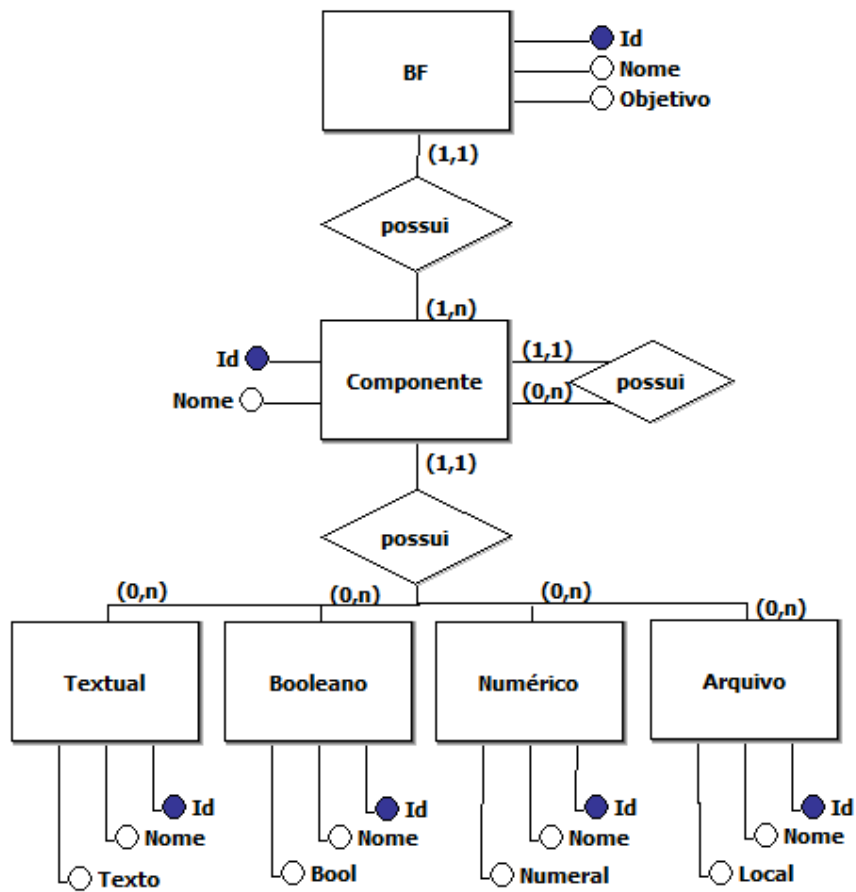


Figura 7 - Modelo Conceitual

A estrutura define um BF com nome e objetivo, que por sua vez possui um ou vários componentes. Como um componente pode possuir um número arbitrário de outros componentes, foi atribuída ao mesmo uma auto-relação. Por fim, como cada componente pode possuir atributos textuais, booleanos, numéricos ou de arquivos, cada atributo foi representado em entidades separadas (uma para cada tipo).

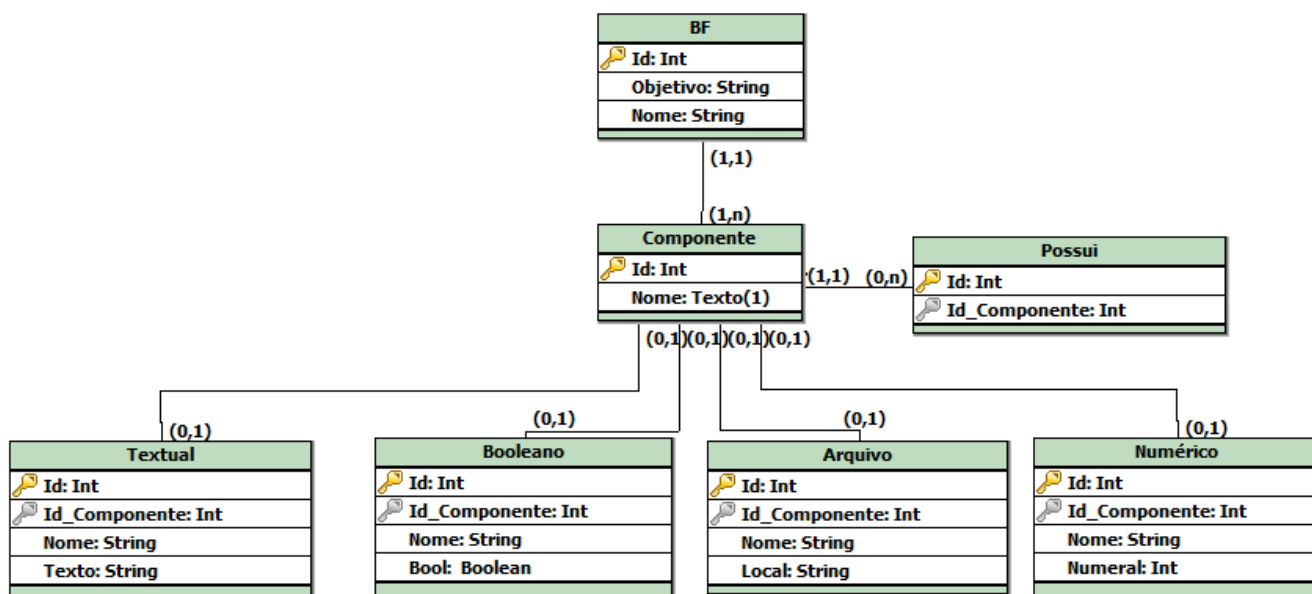


Figura 8 - Modelo Lógico

Na Figura 8 há a arquitetura representada pelo modelo lógico derivada do modelo conceitual da Figura 7. Foram criadas as devidas chaves estrangeiras para as relações assim como uma tabela que representa a auto-relação dos componentes, possibilitando a associação de um componente aos seus n filhos.

4. CAPÍTULO IV – EXEMPLO DE USO DA FERRAMENTA

Este capítulo é voltado para demonstração de uso da ferramenta, mostrando na prática como é realizado seu apoio ao processo proposto.

4.1. Criação de um BF

O primeiro passo da demonstração é a criação do BF, de acordo com os atributos e regras específicas do BF estudado.

Através de um usuário autenticado com permissão de “Projetista de Estudos Empíricos”, acessamos “Projetar BFs”. Na nova tela (Figura 9), clicamos na opção “Criar Novo”. Aqui inserimos os primeiros dados de descrevem o BF: nome (definido arbitrariamente como “BF Moura”) e seu objetivo, que deve se referir ao objeto principal do estudo empírico (no caso, “Manutenabilidade OA”). Clicamos em “Criar”.

A imagem mostra a interface de usuário do "Gerenciador de Benchmarking Frameworks v0.1". No topo, há uma barra de título com o nome do sistema e um botão "Projetar BFs" no canto superior direito. O conteúdo principal é uma seção intitulada "Criar" que contém um formulário para criar um novo Benchmarking Framework (BF). O formulário possui dois campos de texto: "Nome" com o valor "BF Moura" e "Objetivo" com o valor "Manutenabilidade OA". Abaixo dos campos, há um botão "Criar". No canto inferior esquerdo da seção, há um link "Voltar".

Figura 9 - Tela de criação de um BF

Após a criação de um novo BF, voltamos para a tela que lista os BF existentes (Figura 10). Podemos observar o BF Moura recém-criado na lista, que apenas possui os atributos básicos que acabamos de definir. Clicamos então em “Editar”.



Figura 10 - Tela de listagem dos BFs existentes

Nesta etapa, é utilizada a DSL visual para a criação da estrutura do BF (Figura 11). Como descrito no Capítulo 2, o BF a ser representado possui dois componentes: Produtos e Cenários. Como cada Produto está associado a seus Cenários, é conveniente a criação do componente principal Produto possuindo um sub-componente chamado Cenário.

Para isso, clicamos em “Adicionar Componente” definindo seu nome como “Produto” e repetimos o processo como nome “Cenário”, dessa vez definindo o componente Produto como pai.

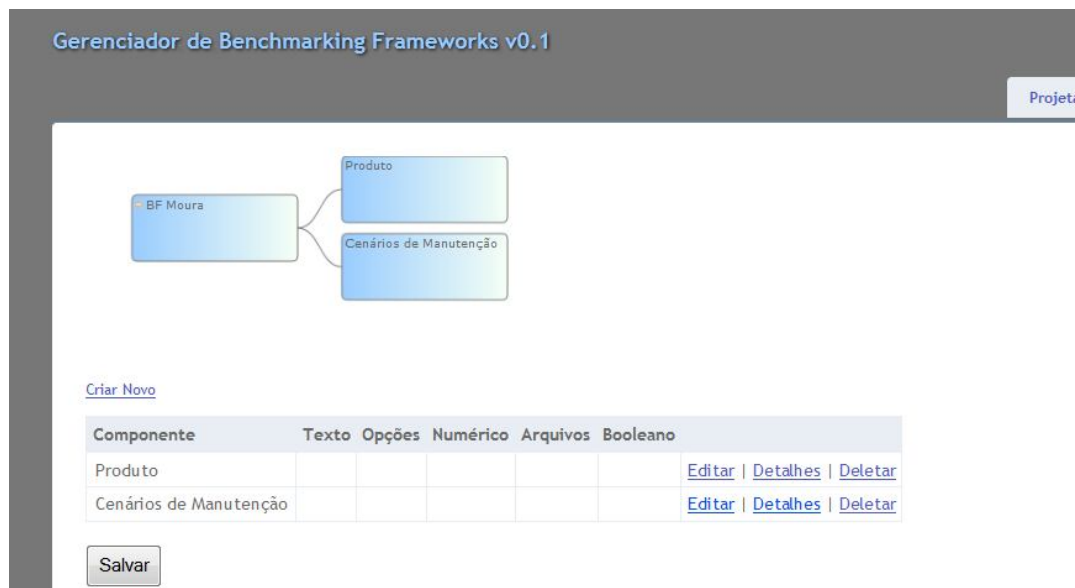


Figura 11 - Componentes Produto e Cenários de Manutenção adicionados

Agora devemos representar os atributos dos componentes criados, correspondentes às tabelas 2.1 e 2.2 para Produto e à tabela 2.3 para Cenário. Para alguns atributos, observa-se que os mesmos se subdividem em outros. Novamente podemos adicionar componentes que representem esses atributos compostos, já que a DSL é genérica o suficiente para não diferenciar reais “componentes” do processo de atributos compostos. Criamos então os novos componentes “Atributos Gerais” e “Atributos Orientados a Aspecto”, como filhos de Produto.

Mais uma vez, criamos componentes para os casos dos atributos compostos “Identificação do Sistema”, “Packaging” (para o caso de Atributos Gerais), “Classificação de Interesses Transversais” e “Interação e Composição de Interesses Transversais” (para o caso de Atributos Orientados a Aspecto). No caso do cenário, criamos os sub-componentes “Tipo de Mudança” e “Nível de Mudança”.

Analogamente, para outros atributos que contêm atributos, utilizamos a criação de sub-componentes até chegarmos aos atributos finais. Estes, representaremos como atributos propriamente, sempre fazendo a devida associação a seu componente-pai.

A diferença na criação dos atributos finais é que devemos definir seu tipo (Texto, Opções de Texto, Booleano, Arquivo ou Numérico), assim como regras de quantidade (Figura 12). Como exemplo, no caso do atributo “Nome”, definimos seu tipo como “Textual” e sua quantidade em um, e ainda sendo obrigatório (a Identificação do Sistema deve necessariamente possuir um nome).

Gerenciador de Benchmarking Frameworks v0.1

Projetar

Editar Componente

Atributos do Componente

Nome

Tipo
escolha um tipo ▼

Quantidade

☐ obrigatório
☒ facultativo

Save

[Voltar](#)

Figura 12 - Edição de componente

Todos os outros componentes e atributos são criados de forma análoga, de acordo com a hierarquia, os tipos e as regras de quantidade descritas nas tabelas 4.4, 4.5 e 4.6.

Para os itens do cenário será levada em consideração apenas sua descrição, já que na avaliação de benchmarks são criadas tabelas adicionais que de apoio, não previstas pelo Projetista de Estudos Empíricos.

Tabela 4.4 - Atributos Gerais do Produto com tipos e regras de quantidade

Atributos Gerais	
Identificação do Sistema	Nome <Textual> <Único><Obrigatório>
	Domínio <Textual> <Único><Obrigatório>
Packaging	Detalhes da Versão<Textual><Único><Facultativo>
	Disponibilidade de Código Fonte <Booleano><Único><Obrigatório>
	Linguagens de Programação <Textual><Multivalorado><Obrigatório>
	Sistema Operacional e Requisitos Mínimos <Textual> <Único><Obrigatório>
Documentação do Ciclo de Vida <Textual><Único><Facultativo>	
Técnicas de Desenvolvimento <Textual><Único><Facultativo>	

Tabela 4.5 - Atributos Orientados a Aspecto do Produto com tipos e regras de quantidade

Atributos Orientados a Aspectos	
Classificação de Interesses Transversais	Funcional e Não Funcional <Booleano><Único><Obrigatório>
	Homogêneo e Heterogêneo <Booleano><Único><Obrigatório>
	Intra-Componente e Inter-Componente <Booleano><Único><Obrigatório>
Interação e Composição de Interesses Transversais	Baseado em Invocação <Booleano><Único><Obrigatório>
	Entrelaçamento no Nível de Componentes <Booleano><Único><Obrigatório>
	Entrelaçamento no Nível de Operações <Booleano><Único><Obrigatório>
	Sobreposição <Booleano><Único><Obrigatório>
Escopo de Interesses Transversais <Textual> <Único><Obrigatório>	
Construções de Linguagens AO <Textual> <Único><Obrigatório>	

Tabela 4.6 - Atributos do Cenário de Manutenção com tipos e regras de quantidade

Atributos do Cenário de Manutenção		
Descrição do Cenário <Textual><Único><Facultativo>		
Tipo de Mudança	Objetivo da Mudança	Manutenção Corretiva
		Manutenção Adaptativa
		Manutenção de Aperfeiçoamento
	Natureza da Mudança	Estrutural
		Comportamental
Nível das Mudanças	Requisitos	
	Análise e Projeto	
	Implementação	

A estrutura final do BF deve ficar representada de acordo com a Figura 13.

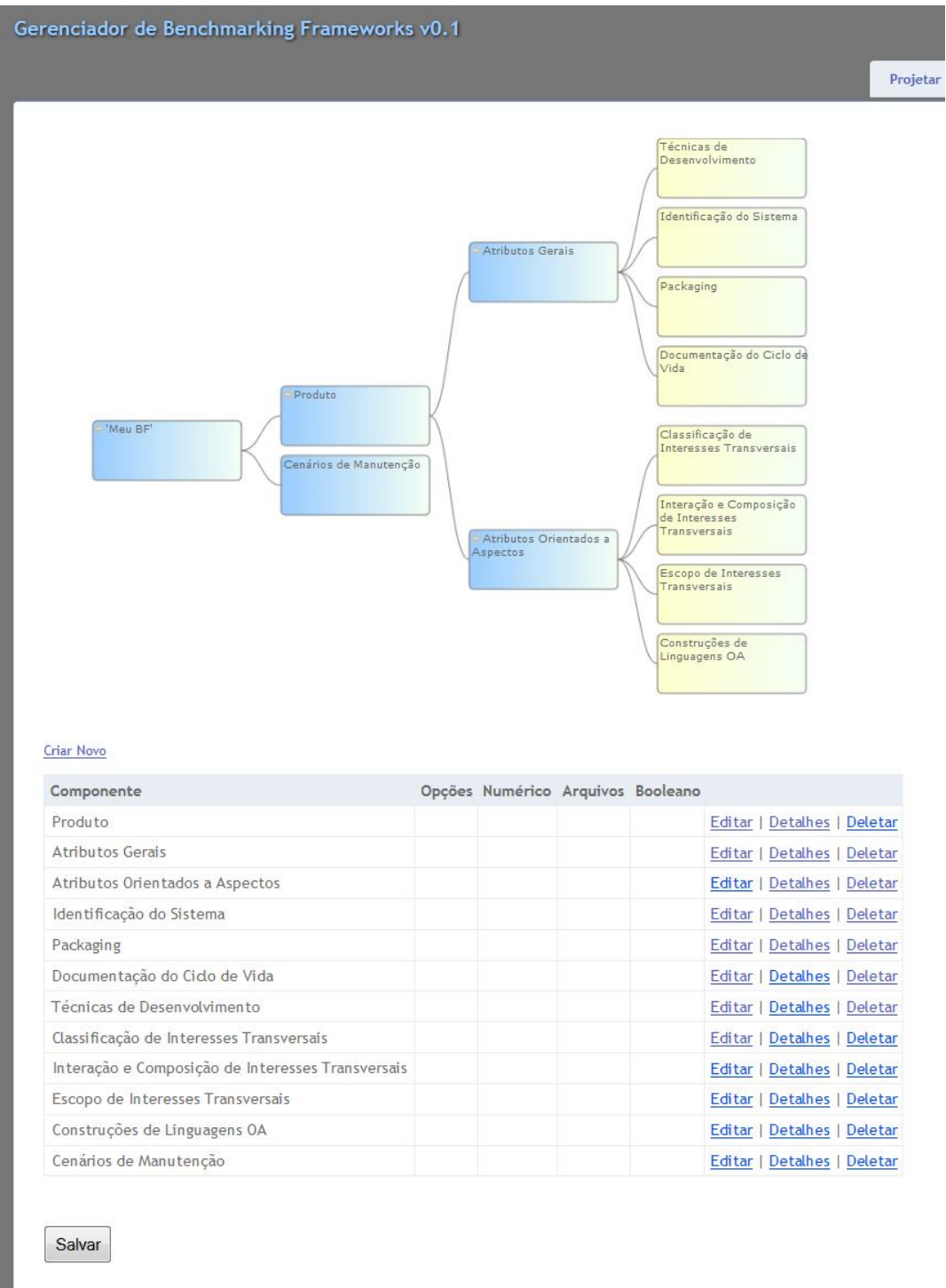


Figura 13 - Tela final da criação do BF Moura

Para concluir a criação, clicamos em “Salvar” para salvar a estrutura criada e retornar à tela principal.

4.2. Cadastro de Benchmarks

Uma vez criada a estrutura do BF, é possível associar valores a seus atributos respeitando as restrições relacionadas ao mesmo. Como temos na estrutura de “BF Moura” todos os atributos de Produto e Cenário devidamente representados, agora o usuário com permissões de “Projetista de Benchmarks” pode dar seqüência ao processo através da inserção de dados para representar os benchmarks.

A avaliação de benchmarks é descrita na tese de Moura (1). São tomadas duas aplicações com características OA, cuja representatividade deve ser avaliada pelo projetista. Também é realizada uma avaliação da utilidade do próprio BF no planejamento de estudos empíricos.

Porém, como o foco do presente trabalho é propor uma ferramenta de suporte à execução do processo, serão consideradas as atividades que dizem respeito à avaliação de benchmarks, mostrando assim como a ferramenta auxilia na execução das mesmas.

Com o login de Projetista de Benchmarks, acessamos o menu “Cadastrar Benchmarks”. Na tela seguinte (Figura 14) escolhemos o recém-criado “BF Moura” clicando na opção “Cadastrar Benchmark” ao lado do mesmo.



Figura 14 - Tela Cadastrar Benchmarks

Vamos considerar a aplicação HealthWatcher [7] para exemplificar o cadastro de um benchmark.

Nesta etapa o Projetista de Benchmark ainda irá avaliar a representatividade do programa. O benefício que o usuário terá com a ferramenta é a apresentação da estrutura do BF com suas regras: o mesmo poderá comparar as características do HealthWatcher derivadas de seu estudo com a estrutura proposta, e assim avaliar a representatividade do aplicativo. À medida que o estudo é realizado, o usuário poderá preencher os valores dos atributos de acordo com o que permite o BF Moura. Logo, prosseguirá avaliando se o HealthWatcher realmente pode ser caracterizado como benchmark. Se assim o julgar, poderá cadastrá-lo, deixando-o associado ao BF Moura.

Durante seu estudo, o usuário terá a visualização completa da estrutura, suas regras e valores. Se o projetista desejar manter o foco no estudo de um determinado componente, o mesmo poderá expandir e contrair o diagrama nos pontos que achar conveniente (ocultando informações sobre componentes não estudados no momento).

Na tabela 4.7 temos os atributos orientados a aspectos preenchidos, onde a coluna HW representa a aplicação HealthWatcher. Introduzimos então os valores booleanos correspondentes a cada um desses atributos. Para o caso dos atributos gerais (tabela 4.8), introduzimos os valores de acordo com a tabela, que possui as informações extraídas da descrição do HealthWatcher.

Para o caso dos atributos Documentação do Ciclo de Vida, fazemos o upload correspondente à documentação do projeto. Como o Sistema roda sobre a plataforma Java, deixamos o atributo Sistema Operacional e Requisitos Mínimos em branco (como permite o BF Moura criado). A tela de atribuição de dados aos componentes é realizada de forma análoga à edição do componente, porém com a possibilidade de preencher dados ao invés de definir os atributos e regras.

Tabela 4.7 - Presença dos Atributos OA na aplicação HealthWatcher

Atributos Orientados a Aspectos		HW
Classificação de Interesses Transversais	Funcional	-
	Não-funcional	X
	Homogêneo	X
	Heterogêneo	X
	Intra-Componente	X
	Inter-Componente	X
Interação e Composição de Interesses Transversais	Baseado em Invocação	X
	Entrelaçamento no Nível de Componentes	X
	Entrelaçamento no Nível de Operações	X
	Sobreposição	X

Tabela 4.8 - Atributos gerais do HealthWatcher preenchidos

Atributos Gerais	
Identificação do Sistema	Nome "HealhWatcher"
	Domínio "Medicina"
Packaging	Detalhes da Versão "Experimental"
	Disponibilidade de Código Fonte "Sim"
	Linguagens de Programação "Java"
	Sistema Operacional e Requisitos Mínimos
Documentação do Ciclo de Vida	
Técnicas de Desenvolvimento: "Orientação a Aspectos" "Orientação a Objetos"	

Para o caso do preenchimento dos dados relativos ao estudo dos cenários, temos dois casos de manutenção para cada cenário descrito (tabela 4.9). O caso 1 é relativo ao estudo feito pelo estudante e o caso 2 pelo especialista mais experiente, onde cada um procurou fazer sua classificação dos tipos de manutenção. Adicionaremos então os nove cenários de acordo com o código e descrição da tabela. Adicionaremos também dois valores para os casos 1 e 2 (correspondente ao atributo classificação do componente cenário).

Tabela 4.9 - Análise do Tipo de Manutenção para os Cenários do HW

Cenários	Descrição dos Cenários do HealthWatcher	Caso 1	Caso 2
R1	Decompor vários Servlets para melhorar a extensibilidade	Aperfeiçoamento	Refactoring
R2	Garantir que o estado do objeto Complaint não será atualizado depois de ser fechado, evitando múltiplas atualizações	Aperfeiçoamento	Corretiva
R3	Encapsular as operações de atualização para melhorar a manutenibilidade	Aperfeiçoamento	Refactoring
R4	Melhorar o encapsulamento do interesse de distribuição para permitir reuso e customização.	Aperfeiçoamento	Refactoring
R5	Generalizar o mecanismo de persistência para permitir reuso e extensão	Aperfeiçoamento	Refactoring
R6	Remover dependências nos objetos de Servlet Response e Request, para facilitar o processo de adição de novas GUIs.	Aperfeiçoamento	Refactoring
R7	Generalizar o mecanismo de distribuição para permitir reuso e extensão	Aperfeiçoamento	Refactoring
R8	Nova funcionalidade adicionada para permitir	Aperfeiçoamento	Evolucionária

	consulta de mais tipos de dados.		
R9	Modularizar o interesse de tratamento de exceção e incluir um comportamento de recuperação de erros mais eficaz nos controladores.	Aperfeiçoamento	Aperfeiçoamento

Assim temos representados os itens de produto e de cenário devidamente associados ao BF Moura, de modo a dar suporte aos estudos do especialista de avaliação sobre os dados (e assim decidir se deseja considerar ou não o HealthWatcher como benchmark e salva-lo). Através deste processo, o projetista tem o benefício de apoiar seu estudo de benchmarking sobre uma estrutura pré-moldada e reutilizável, o que significa que para futuros estudos será possível também o acesso aos dados de avaliações já salvas (permitindo o acúmulo de conhecimentos sobre estudos com um mesmo objetivo).

5. CAPÍTULO V – CONCLUSÕES

Este capítulo faz a análise da ferramenta e dos benefícios da mesma sobre a execução do processo da Seção 2.2. Logo, são discutidas propostas de melhoria e novas funcionalidades para trabalhos futuros.

5.1. Avaliação do uso da ferramenta

Como descrito da Seção 2.3, a ferramenta oferece um meio de definir os dados de instâncias do BF, bem como cadastrar aplicações e cenários de mudanças associados a cada instância. Através das possibilidades de construção da estrutura de BF foi possível definir as hierarquias de componentes do processo de Moura, assim como associar aos mesmos todos os tipos de atributos e regras necessárias para estes (Seção 4.1).

O Projetista de Estudos Empíricos pode montar e editar a estrutura sempre visualizando a hierarquia montada, os tipos e regras associadas de forma intuitiva em apenas uma tela, o que unifica de forma clara diversas informações que, sem a ferramenta, estariam espalhadas nas tabelas de Atributos Gerais do Produto, Atributos Orientados a Aspecto do Produto e Atributos do Cenário de Manutenção.

A representação hierárquica também é utilizada através de uma interface similar pelo Projetista de Benchmarks, que também pode associar os dados relativos aos benchmarks a partir de uma única tela. Sem a ferramenta, seriam necessárias ainda mais tabelas para cada caso de benchmark, que não estariam ligados ao seu BF nem teriam um meio de serem guardados de forma associada à estrutura.

Além de tais benefícios diretos, espera-se que o uso da ferramenta também contribua para a sistematização e organização da execução do processo: uma vez que o BF pode ficar salvo no sistema com seus benchmarks associados, projetistas de benchmarks podem sempre visualizar essas

informações de forma a facilitar a avaliação e o cadastro de novos benchmarks de produtos e cenários.

5.2. Trabalhos futuros

A ferramenta foi pensada com funcionalidades diretas, visando uma maneira prática de apoiar a manipulação de dados associados ao processo de Moura. Porém, assim como foi introduzida a funcionalidade de busca por componentes (algo não descrito no processo original), facilidades e melhoramentos podem ser adicionadas. Isso pode ser realizado principalmente através da continuação da validação a aplicação em outros casos de estudos empíricos.

A interface gráfica, que tem papel fundamental na ferramenta, pode ser melhorada. Apesar de já existir uma representação adequada para as hierarquias de componentes, a mesma poderia ser mais rica, contendo visualização dos tipos, regras e conteúdos associados a essa hierarquia de forma mais dinâmica. Assim como a interface, a própria DSL poderia evoluir a medida que fossem averiguadas novas necessidades, de modo que fosse feita uma especificação formal (BNF) e fossem utilizadas tecnologias específicas para a definição dessa linguagem.

No caso da descrição dos cenários, que foi simplificado em relação a sua definição original (Tabela 4.6), a ferramenta poderia oferecer meios de facilitar a avaliação de benchmarks sem alterar a estrutura original do BF.

Outro exemplo de melhoramento identificado é a falta de versatilidade para a edição do BF após sua criação: uma vez que o BF é definido e são associados benchmarks ao mesmo, sua modificação não é recomendada, já que a versão modificada do BF pode passar a não ser adequada para os benchmarks. Uma versão futura da ferramenta poderia dar suporte a essas modificações, gerando avisos e recomendações aos projetistas acerca dessas modificações.

Bibliografia

1. **Moura, M.** Um Benchmarking Framework para Avaliação da Manutenibilidade de Software Orientado a Aspectos. *Dissertação de Mestrado em Engenharia da Computação, Universidade de Pernambuco*. 2008.
2. *On Using a Benchmark to Evaluate C++ Extractors. In Proceedings of the 10th international Workshop on Program Comprehension.* **Sim, S.E., Holt, R. C., Easterbrook, S.** 2002. Washington, DC : IEE Computer Society, June 27 - 29, 2002, Vol. 114.
3. **Fowler, Martin.** *Domain-Specific Languages*. s.l. : Addison-Wesley, 2010.