

**UNIVERSIDADE REGIONAL DE BLUMENAU**  
**CENTRO DE CIÊNCIAS EXATAS E NATURAIS**  
**CURSO DE CIÊNCIAS DA COMPUTAÇÃO – BACHARELADO**

**FERRAMENTA DE APOIO A REALIZAÇÃO DE**  
**EXPERIMENTOS EM ENGENHARIA DE SOFTWARE**

**JEISON DANDOLINI**

**BLUMENAU**  
**2006**

**2006/2-17**

**JEISON DANDOLINI**

**FERRAMENTA DE APOIO A REALIZAÇÃO DE  
EXPERIMENTOS EM ENGENHARIA DE SOFTWARE**

Trabalho de Conclusão de Curso submetido à  
Universidade Regional de Blumenau para a  
obtenção dos créditos na disciplina Trabalho  
de Conclusão de Curso II do curso de Ciências  
da Computação — Bacharelado.

Prof. Everaldo Artur Grahl - Orientador

**BLUMENAU  
2006**

**2006/2-17**

# **FERRAMENTA DE APOIO A REALIZAÇÃO DE EXPERIMENTOS EM ENGENHARIA DE SOFTWARE**

Por

**JEISON DANDOLINI**

Trabalho aprovado para obtenção dos créditos na disciplina de Trabalho de Conclusão de Curso II, pela banca examinadora formada por:

Presidente: \_\_\_\_\_  
Prof. Everaldo Artur Grahl, M Engenharia – Orientador, FURB

Membro: \_\_\_\_\_  
Prof. Ricardo de Alencar Azambuja, M. Administração – FURB

Membro: \_\_\_\_\_  
Prof. Alexander Roberto Valdameri, M. Engenharia – FURB

Blumenau, 12 de dezembro de 2006

Dedico este trabalho a meu pai, minha mãe e minha irmã e a todos os amigos, especialmente aqueles que me ajudaram diretamente na realização deste.

## **AGRADECIMENTOS**

À minha mãe Helena, ao meu pai Valério e a minha irmã Jane que me deram muito apoio, principalmente nos momentos em que precisei.

Aos meus amigos, que mesmo com a minha ocupação, sempre estavam ao meu lado.

Ao meu orientador, Everaldo Artur Grahl, por ter acreditado e contribuído bastante para a conclusão deste trabalho.

A Deus que me deu saúde para poder vencer mais esta fase da minha vida.

Se você quiser fazer alguma coisa deliberadamente, faça um tanto a mais do que apenas o seu dever.

Bruce Lee.

## **RESUMO**

Este trabalho tem como objetivo apresentar a especificação e implementação de uma ferramenta de apoio a experimentos em Engenharia de Software, que permite a definição dos objetivos, a definição das questões e métricas e a verificação das hipóteses. A ferramenta é direcionada para experimentos relacionados à melhoria de processo.

Palavras-chave: Engenharia de software experimental. Melhoria de processo.

## **ABSTRACT**

This work has a objective to present the specification and implementation of a support tool the experiments in Software Engineering, that allows the definition of objectives, the definition of the metric and questions and the verification of the hypotheses. The tool is directed to experiments related to the process improvement.

Key-words: Empirical software engineering. Improvement of process.

## LISTA DE ILUSTRAÇÕES

Figura 1 - Os conceitos de um experimento .....	20
Figura 2 - Etapas e fases do processo GQM.....	23
Figura 3 - Tela do sistema <i>MedPlan</i> .....	29
Figura 4 - Tela do sistema Metrics .....	30
Figura 5 - Diagrama de casos de uso .....	33
Figura 6 - Diagrama de atividades.....	34
Figura 7 - Diagrama de classes.....	36
Figura 8 - Diagrama entidade relacionamento conceitual .....	48
Figura 9 - Diagrama entidade relacionamento físico.....	49
Figura 10 - Tela principal do sistema .....	54
Figura 11 - Tela de login .....	55
Figura 12 - Tela de criação de um novo experimento .....	56
Figura 13 - Tela principal do experimento .....	56
Figura 14 - Tela de definição dos objetivos .....	57
Figura 15 - Tela de definição de questões e métricas.....	58
Figura 16 - Tela de descrição da instrumentação .....	59
Figura 17 - Tela de seleção do contexto .....	59
Figura 18 - Tela de cadastro de grupo de projetos .....	60
Figura 19 - Tela de cadastro de projetos .....	60
Figura 20 - Tela de cadastro de participantes do experimento .....	61
Figura 21 - Gráfico do perfil dos participantes.....	61
Figura 22 - Tela de preparação e recursos utilizados .....	62
Figura 23 - Tela do questionário.....	62
Figura 24 - Impressão do questionário .....	63
Figura 25 - Tela da estatística descritiva .....	64
Figura 26 - Tela do teste de hipótese .....	64
Figura 27 - Gráfico das variáveis .....	65
Figura 28 - Gráfico das variáveis .....	65
Figura 29 - Tela da análise quantitativa.....	66
Figura 30 - Tela da análise qualitativa.....	66
Figura 31 - Tela da descrição das validades .....	67

Figura 32 - Relatório dos participantes.....	85
Figura 33 - Dados gerais do estudo .....	86
Figura 34 - Dados específicos do estudo .....	86
Figura 35 - Gráfico da análise quantitativa .....	87
Figura 36 - Gráfico da estatística descritiva .....	87

## **LISTA DE TABELAS**

Tabela 1 - Tabela do teste de hipótese.....	28
---	----

## **LISTA DE SIGLAS**

ESE – Engenharia de Software Experimental

OO – Orientação a objetos

UML – *Unified modeling language*

GQM – *Goal question metric*

## **LISTA DE SÍMBOLOS**

B – beta

$\alpha$  - Alfa

# SUMÁRIO

<b>1 INTRODUÇÃO.....</b>	<b>15</b>
1.1 OBJETIVOS DO TRABALHO .....	15
1.2 ESTRUTURA DO TRABALHO .....	16
<b>2 FUNDAMENTAÇÃO TEÓRICA .....</b>	<b>17</b>
2.1 ENGENHARIA DE SOFTWARE .....	17
2.2 EXPERIMENTOS EM ENGENHARIA DE SOFTWARE.....	18
2.2.1 Vocabulário da experimentação .....	19
2.2.2 Organização do experimento.....	21
2.2.3 A medição do experimento .....	21
2.2.4 GOAL QUESTION METRIC (GQM) .....	22
2.2.5 Validades dos dados do experimento.....	24
2.2.6 Fases do processo de experimentação.....	25
2.3 ESTATÍSTICA DIRECIONADA A EXPERIMENTOS.....	26
2.3.1 Medidas de tendência central .....	26
2.3.2 Medidas de variação.....	27
2.3.3 Estatística descritiva.....	27
2.3.4 Teste de hipótese .....	27
2.4 TRABALHOS CORRELATOS .....	28
<b>3 DESENVOLVIMENTO DO TRABALHO .....</b>	<b>31</b>
3.1 REQUISITOS PRINCIPAIS DO PROBLEMA A SER TRABALHADO.....	31
3.2 ESPECIFICAÇÃO .....	33
3.2.1 Diagramas de caso de uso .....	33
3.2.2 Diagrama de atividades .....	34
3.2.3 Diagrama de classes .....	35
3.2.3.1 Descrição da classe “TRegrasFatorVariacao” .....	37
3.2.3.2 Descrição da classe “TRegrasVariavel” .....	37
3.2.3.3 Descrição da classe “TRegrasValoresPossiveis” .....	38
3.2.3.4 Descrição da classe “TRegrasVerificações” .....	38
3.2.3.5 Descrição da classe “TRegrasRepostasParticipante” .....	39
3.2.3.6 Descrição da classe “TRegrasHipoteses” .....	39
3.2.3.7 Descrição da classe “TRegrasProjetos” .....	40

3.2.3.8	Descrição da classe “TRegrasGruposProjetos” .....	40
3.2.3.9	Descrição da classe “TRegrasExperimento” .....	40
3.2.3.10	Descrição da classe “TRegrasContexto” .....	41
3.2.3.11	Descrição da classe “TRegrasDadosAnaliseQualitativa” .....	41
3.2.3.12	Descrição da classe “TRegrasDescricaovalidades” .....	42
3.2.3.13	Descrição da classe “TRegrasTesteHipotese” .....	42
3.2.3.14	Descrição da classe “TRegrasParticipante” .....	43
3.2.3.15	Descrição da classe “TRegrasEstatistica” .....	43
3.2.3.16	Descrição da classe “TRegrasDescricaoAnalises” .....	44
3.2.3.17	Descrição da classe “TRegrasCalculos” .....	44
3.2.3.18	Descrição da classe “TRegrasDadosAnaliseQuantitativa” .....	45
3.2.3.19	Descrição da classe “TAggrupadordados” .....	45
3.2.3.20	Descrição da classe “TCalculosEstatisticos” .....	46
3.2.4	Diagrama de entidade e relacionamento conceitual .....	48
3.2.5	Diagrama de entidade e relacionamento físico .....	49
3.2.6	Dicionário de dados .....	49
3.3	IMPLEMENTAÇÃO .....	50
3.3.1	Técnicas e ferramentas utilizadas .....	50
3.3.2	Trecho de código do cálculo da moda da <i>unit</i> uregrasestatistica.pas .....	51
3.3.3	Trecho de código do cálculo da mediana da <i>unit</i> uregrasestatistica.pas .....	52
3.3.4	Trecho de código do cálculo da variância da <i>unit</i> uregrasestatistica.pas .....	53
3.3.5	Operacionalidade da implementação .....	54
3.3.5.1	Tela principal do sistema .....	54
3.3.5.2	Tela de <i>Login</i> .....	55
3.3.5.3	Tela de criação de um novo experimento .....	55
3.3.5.4	Tela principal do experimento .....	56
3.3.5.5	Tela de definição dos objetivos .....	56
3.3.5.6	Tela de definição das questões e métricas .....	57
3.3.5.7	Tela de descrição da instrumentação .....	58
3.3.5.8	Tela de seleção do contexto .....	59
3.3.5.9	Tela de cadastro de grupo de projetos .....	59
3.3.5.10	Tela de cadastro de projetos .....	60
3.3.5.11	Tela de cadastro de participantes do experimento .....	60
3.3.5.12	Tela de preparação e recursos utilizados .....	61

3.3.5.13 Tela do questionário .....	62
3.3.5.14 Tela da estatística descritiva .....	63
3.3.5.15 Tela do teste de hipótese.....	64
3.3.5.16 Tela da análise quantitativa e qualitativa.....	65
3.3.5.17 Tela da descrição das validades.....	67
3.3.5.18 Apresentação dos dados.....	68
3.4 RESULTADOS E DISCUSSÃO .....	68
<b>4 CONCLUSÕES.....</b>	<b>70</b>
4.1 EXTENSÕES .....	70
<b>REFERÊNCIAS BIBLIOGRÁFICAS .....</b>	<b>71</b>
<b>APÊNDICE A – Descrição dos casos de uso .....</b>	<b>72</b>
<b>APÊNDICE B – Relatórios e gráficos do sistema.....</b>	<b>85</b>
<b>APÊNDICE C – Quadro do dicionário de dados .....</b>	<b>88</b>

# 1 INTRODUÇÃO

A Engenharia de Software Experimental é uma das áreas de pesquisa da Engenharia de Software, cujo objetivo é o aprimoramento das técnicas de Engenharia de Software a partir de um método específico (experimentação) na elaboração de novos métodos para apoio ao desenvolvimento de software. Seu surgimento aconteceu com a necessidade da verificação de teorias, no que se diz respeito a melhoria de processos de desenvolvimento de software. Segundo Travassos (2002, p. 3), “ Somente experimentos verificam teorias, somente experimentos podem explorar os fatores críticos e dar luz ao fenômeno novo para que as teorias possam ser formuladas e corrigidas.”

Para se verificar a importância de experimentos em Engenharia de Software, pode-se citar um exemplo onde determinada empresa de software decide incorporar uma nova ferramenta de apoio no desenvolvimento de seus softwares. A inclusão desta ferramenta tem o objetivo de melhorar o processo de desenvolvimento. Para que o presidente da empresa possa confirmar a sua teoria de que a ferramenta irá realmente contribuir para o melhoramento do processo, a equipe de desenvolvimento deverá passar por um processo de experimentação, comparando os resultados obtidos com a utilização da ferramenta com os resultados obtidos sem a sua utilização. No final do processo de experimentação, os dados serão analisados e a decisão entre utilizar ou não a nova ferramenta será tomada.

Pretendeu-se com este trabalho estudar e relatar de forma mais aprofundada a ESE e avaliar o potencial de sua aplicação em projetos de software, usando técnicas como o *Goal Question Metric* (GQM) que será usado na definição de medidas para os experimentos. A ferramenta desenvolvida para este trabalho possui o nome de EX – Experimentos em Engenharia de Software.

## 1.1 OBJETIVOS DO TRABALHO

O objetivo deste trabalho é desenvolver uma ferramenta para suportar a realização de experimentos em Engenharia de Software.

Os objetivos específicos do trabalho são:

- a) direcionar os experimentos para questões relativas a melhoria do processo de software;
- b) utilizar a técnica GQM para a definição das medidas nos experimentos;
- c) adotar técnicas de estatística na análise de resultados dos experimentos.

## 1.2 ESTRUTURA DO TRABALHO

Este trabalho possui quatro capítulos: Introdução, fundamentação teórica, desenvolvimento do trabalho e conclusões. Neste capítulo são apresentados o contexto do trabalho, objetivos e estrutura.

No capítulo 2 são abordados temas como Engenharia de Software, Experimentos em Engenharia de Software, *GOAL QUESTION METRIC* (GQM) e cálculos estatísticos aplicados aos dados do experimento.

No capítulo 3 são apresentados a definição dos requisitos da ferramenta, a especificação, a implementação, trechos de código, telas do sistema e por fim os resultados e discussões do trabalho.

O capítulo 4 contém a conclusão e extensões do trabalho.

## 2 FUNDAMENTAÇÃO TEÓRICA

Neste capítulo serão abordados os temas principais relacionados ao trabalho como: Engenharia de Software, Experimentos em Engenharia de Software e estatística direcionada a experimentos.

### 2.1 ENGENHARIA DE SOFTWARE

A Engenharia de Software serve para, além de auxiliar o desenvolvimento de software, integrar os clientes com os desenvolvedores, possibilitando uma maior eficiência do software e satisfação do cliente em questão. Segundo Pfleeger (2004, p. 11), “Um componente-chave do desenvolvimento de software é a comunicação entre os clientes e os desenvolvedores; caso ela falhe, o sistema também falhará.”.

Uma melhor definição sobre Engenharia de Software pode ser relacionada a afirmação de Sommerville (2003, p. 5): “A engenharia de software é uma disciplina da engenharia que se ocupa de todos os aspectos da produção de software, desde os estágios iniciais de especificação do sistema até a manutenção desse sistema, depois que ele entrou em operação.” Em outras palavras, pode-se relacionar a Engenharia de Software aos métodos a serem seguidos para um bom desenvolvimento e funcionamento do software, onde a importância se dá ao fato de se ter os requisitos do software devidamente documentados, o desenvolvimento feito com objetivos bem definidos e os testes realizados com precisão, para que finalmente o produto possa ser entregue ao usuário final. Esta característica de desenvolvimento está relacionada à melhoria de processo de software, cujo objetivo é melhorar o desempenho na atividade que está sendo executada.

Pode-se dizer que a melhoria do processo afeta diretamente a qualidade do produto, pois faz parte do ciclo de vida do produto até a sua conclusão. Segundo Sommerville (2003, p. 477): “A melhoria do processo significa compreender os processos existentes e modificá-los, a fim de melhorar a qualidade do produto e/ou reduzir os custos e o tempo de desenvolvimento.” Algumas características de processo podem ser visualizadas no quadro 1.

Característica do processo	Descrição
Facilidade de compreensão	Até que ponto o processo está explicitamente definido e com que facilidade se pode compreender a definição do processo?
Visibilidade	As atividades de processo culminam em resultados nítidos, de modo que o progresso do processo seja exatamente visível?
Facilidade de suporte	Até que ponto as atividades do processo podem ser apoiadas por ferramentas case?
Aceitabilidade	O processo definido é aceitável e utilizável pelos engenheiros responsáveis pela produção do produto de software?
Confiabilidade	O processo está projetado de tal maneira que seus erros sejam evitados ou identificados antes que resultem em erros do produto?
Robustez	O processo pode continuar, mesmo que surjam problemas inesperados?
Facilidade de manutenção	O processo pode evoluir para refletir os requisitos mutáveis da organização ou melhorias de processo identificadas?
Rapidez	Com que rapidez pode ser concluído o processo de entrega de um sistema, a partir de uma determinada especificação?

Fonte: Sommerville (2003, p. 477).

Quadro 1 - Características do processo

O quadro 1 representa as questões que podem ser feitas cujo foco é a melhoria do processo de software, indicando algumas características que o processo deve ter. As questões voltadas para a melhoria do processo são de extrema importância para se obter métricas que quantificam os dados do problema a ser tratado. Segundo Sommerville (2003, p. 481): “Os engenheiros que trabalham em um projeto são questionados sobre o que realmente está em andamento. As respostas a um questionário formal são aprimoradas durante entrevistas pessoais com os envolvidos no processo.”

## 2.2 EXPERIMENTOS EM ENGENHARIA DE SOFTWARE

Como foi abordado anteriormente, a Engenharia de Software é bastante usada para o desenvolvimento de softwares em projetos de pequenas, médias ou grandes empresas. Ela indica a possibilidade de uso de determinadas ferramentas para modelar um software, ou determinadas técnicas para o construir de maneira mais padrão e documentada possível.

A Engenharia de Software voltada a experimentos pode ser relacionada a fatores como melhoria de processo, como visto na seção anterior, porém obtendo dados de avaliação de forma mais rápida e precisa. Segundo Pfleeger (2004, p. 417), “Em um experimento formal, diversos métodos são utilizados para reduzir tendências e eliminar fatores que se confundem, de modo que a causa e o efeito possam ser avaliados com alguma confiança”.

Para se entender a diferença entre Engenharia de Software e Experimentos em Engenharia de Software, deve-se ter em mente que a primeira disponibiliza as ferramentas necessárias para desenvolver o software, e a segunda obtém métricas para posteriormente trabalhar com os dados coletados e tomar decisões.

Segundo Travassos (2002, p. 2), “Experimentação é o centro do processo científico. Somente experimentos verificam teorias. Somente experimentos podem explorar fatores críticos e dar luz ao fenômeno novo para que as teorias possam ser formuladas e corrigidas.” Para se entender o verdadeiro conceito de experimento, deve-se aplicá-lo ao desenvolvimento de software, onde existe a necessidade de melhoria constante do processo de desenvolvimento, cujas regras só poderão ser melhoradas com o uso de técnicas de experimentação que mostrem as partes deficientes do processo.

### 2.2.1 Vocabulário da experimentação

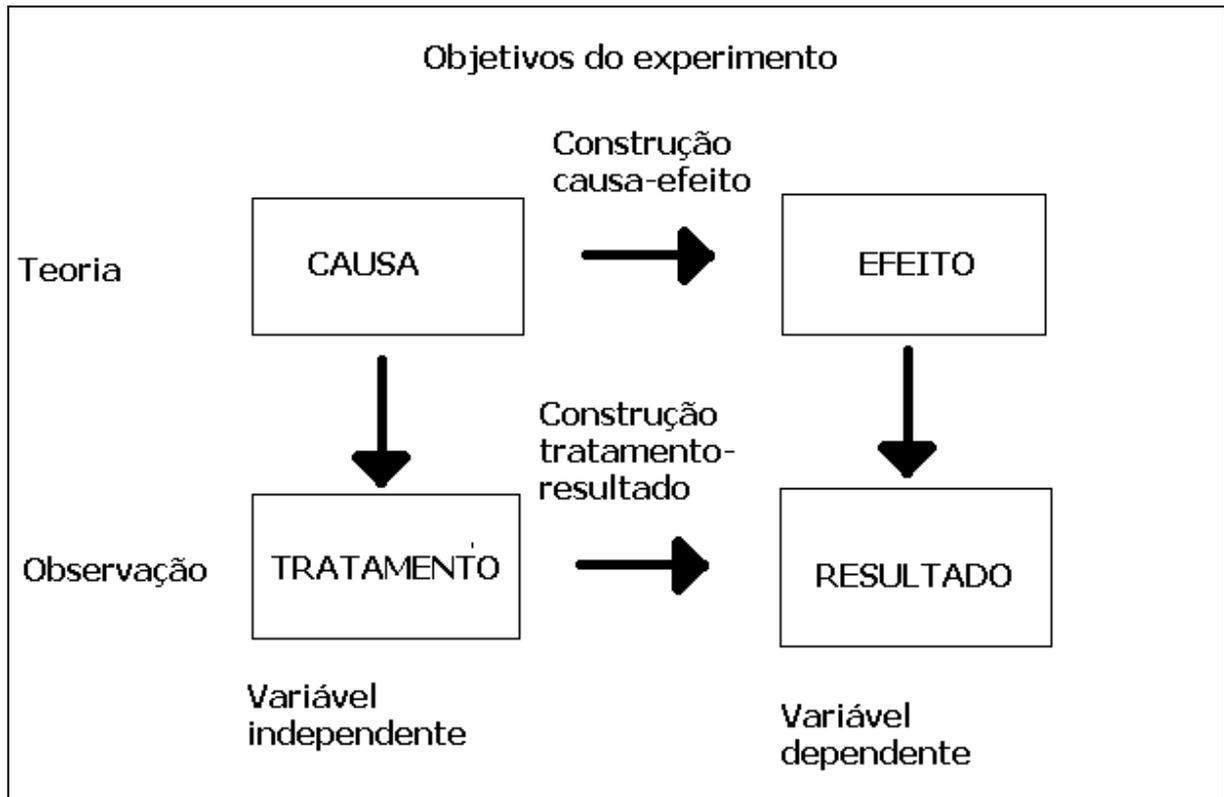
Segundo Travassos (2002, p. 7), “Os elementos principais do experimento são as variáveis, os objetos, os participantes, o contexto do experimento, hipóteses,[sic] e o tipo de projeto do experimento.” Estes elementos podem ser descritos como (TRAVASSOS, 2002):

- a) variáveis: são valores indefinidos no projeto, como por exemplo, o número de funcionalidades adicionais de um determinado cadastro, que irão contribuir para o resultado final do experimento;
- b) objetos: compõe a instrumentação do experimento;
- c) participantes: são os indivíduos que conduzem o experimento;
- d) contexto do experimento: é composto das condições em que o experimento está sendo executado;
- e) hipóteses: suposição de algum resultado que se deseja alcançar;
- f) tipo de projeto do experimento: determina a maneira como um experimento será conduzido.

Com estes elementos, tem-se por objetivo a realização de estudos que possam

comprovar a melhora de algum processo de desenvolvimento. Esta melhora está relacionada á verificação de teorias, formuladas através das hipóteses do experimento.

Existem dois tipos de variáveis: as independentes que são os valores de entrada do experimento, chamados de fatores, e as dependentes que são os valores de saída do experimento, chamados de resultado. Na figura 1 pode-se ter uma idéia de como funciona o tratamento de variáveis.



Fonte: Travassos (2002, p. 7).

Figura 1 - Os conceitos de um experimento

A figura 1 ilustra as variáveis envolvidas em um experimento. As variáveis independentes são representadas com a causa do experimento, que vão ocasionar os resultados para as variáveis dependentes. As variáveis dependentes possuem os dados do resultado do experimento. Segundo Travassos (2002, p. 7), “O próprio valor de uma variável dependente se chama ‘resultado’”, então quando se fala em valor de variáveis significa que a variável independente apresenta a causa enquanto a variável dependente apresenta o resultado das informações do experimento.

Os participantes são os indivíduos selecionados para conduzir o experimento. Eles são os responsáveis por informar parâmetros para o experimento como o valor das variáveis.

O contexto do experimento é o modo em que o experimento está inserido no foco do projeto, qual a equipe que vai executar o experimento, qual o tamanho do problema que está sendo analisado e se os resultados são válidos para o contexto.

A hipótese significa a suposição de algum valor. Inicialmente pode-se ter uma hipótese nula, mas o que se procura fazer é eliminar a hipótese nula baseando-se em uma ou mais hipóteses alternativas que tenham relação estatísticas de causa-efeito.

O tipo de projeto do experimento determina, segundo Travassos (2002, p. 9), “[...] a maneira como um experimento será conduzido. A decisão sobre alocação dos objetos e dos participantes é feita nesse momento. Também a maneira como os tratamentos serão aplicados aos objetos é definida.”

### 2.2.2 Organização do experimento

Segundo Travassos (2002, p. 9): “Os princípios gerais da organização do experimento são a aleatoriedade, o agrupamento(*blocking*),[sic] e o balanceamento.” A aleatoriedade é utilizada para evitar que um valor interfira no valor de outro e para a seleção dos participantes do experimento. O agrupamento é utilizado quando existe um valor não esperado no experimento que está influenciando o resultado gerando informações incorretas. Segundo Travassos (2002, p. 10): “O agrupamento sistematicamente elimina o efeito indesejado durante a comparação dos tratamentos. Dentro do bloco o efeito indesejado é indiferente e pode ser eliminado da consideração. O agrupamento aumenta a precisão do experimento.” O balanceamento contribui para a melhora da análise estatística pois consiste em ter um número igual de informações para cada participante.

### 2.2.3 A medição do experimento

Para avaliar o resultado de uma pesquisa, é preciso obter medidas a partir dos dados coletados, para verificar as hipóteses levantadas e ter um dimensionamento do problema. Segundo Travassos (2002, p. 10): “A medição é a parte central de um estudo experimental. A medição é definida como o mapeamento do mundo experimental para o mundo formal ou relacional. O objetivo principal deste mapeamento é caracterizar e manipular os atributos das entidades empíricas de maneira formal.”

Os dados extraídos a partir das medidas são chamados de métricas e são de quatro tipos: nominal, ordinal, intervalo e razão. Segundo Travassos (2002, p. 11): “A escala nominal apresenta o atributo de uma entidade como o nome ou símbolo. A classificação das

entidades pode ser feita a partir dos atributos nominalmente mapeados. A escala ordinal ordena as entidades segundo um critério definido. Nesse caso, as afirmações como ‘maior do que...’ ou ‘mais complexo do que...’ podem ser feitas. A escala do intervalo ordena os valores da mesma forma que a escala ordinal, mas acrescenta a noção da distância relativa entre as entidades. Na escala razão existe o valor zero significativo e a razão entre medidas é significativa. A possibilidade de produzir as afirmações significativa, chama também ‘a potência da escala’ cresce da escala nominal à escala da razão.”

As métricas são adquiridas geralmente a partir de questionários. Uma melhor descrição sobre aquisição de métricas pode ser encontrada na próxima seção.

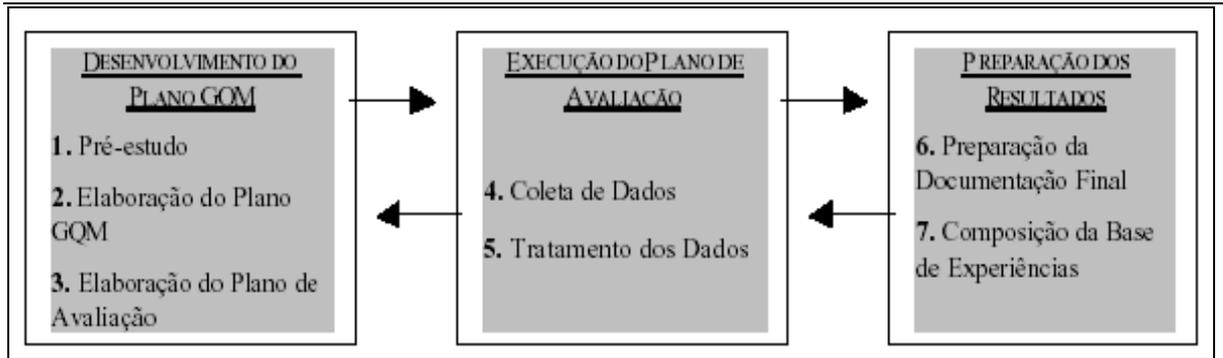
#### 2.2.4 GOAL QUESTION METRIC (GQM)

Para a realização de experimentos em Engenharia de Software pode-se utilizar o GQM para facilitar na adoção de princípios sistemáticos. O GQM baseia-se nos princípios da “Avaliação orientada a objetivos” que, segundo Gladcheft, Sanches e Silva (2001, p. 3), “possui o objetivo de servir como uma metodologia genérica para orientar a elaboração e execução de programas de avaliação da qualidade de processos na área de Engenharia de Software”, e tem importância na obtenção de métricas para o experimento.

Existem três tópicos dessa abordagem que são:

- a) paradigma: onde se deve ter os objetivos documentados, que são as metas que se deseja alcançar com o desenvolvimento do software;
- b) modelo: onde são definidos os objetos de estudo, como por exemplo a seleção dos participantes que vão contribuir no resultado da pesquisa;
- c) método: que segundo Gladcheft, Sanches e Silva (2001, p. 3), “inclui planejamento, execução e empacotamento das experiências obtidas durante o programa.”

Estas três etapas podem ser melhores visualizadas na figura 2.



Fonte: Gladcheft, Sanches e Silva (2001, p. 3).

Figura 2 - Etapas e fases do processo GQM

O pré-estudo, elaboração do plano GQM e elaboração do plano de avaliação correspondem ao paradigma. A coleta de dados e o tratamento dos dados correspondem ao modelo. A preparação da documentação final e a composição da base de experiências correspondem ao método. Segundo Gross (2001): “Durante o pré-estudo, os objetivos da avaliação são definidos levando-se em conta os seguintes aspectos: objeto, propósito, foco de qualidade, ponto de vista e ambiente.” Os respectivos aspectos são apresentados no quadro 2.

Objetivo		
Objeto	Análise de	processos, produtos e recursos de software
Propósito	Com o propósito de	Caracterização, evolução, monitoramento, controle, melhoramento
Foco de qualidade	Com relação a	custo, correção, defeitos, mudanças, confiabilidade, facilidade de uso, etc.
Ponto de vista	Sob o ponto de vista do	usuário, gerente senior, gerente de projeto, desenvolvedor, etc.
Ambiente	no seguinte contexto	organização, projeto, problema, processos, etc.

Fonte: Gross (2001).

Quadro 2 - Esquematização do objetivo

Existem ainda os objetivos, global e de medição. O objetivo global define uma descrição geral do objetivo do experimento e o objetivo da medição indica qual a meta que se deseja alcançar com a realização das perguntas.

É importante ressaltar que na avaliação de um software, não só aspectos técnicos são analisados, mas também a funcionalidade comercial do software. Um exemplo seria um sistema educacional, onde além de estar enquadrado aos itens da figura 2, deve estar também dentro das normas educacionais previstas pela lei.

Na elaboração das questões, a técnica “Entrevista Estruturada” é um modo do especialista da área expressar os seus conhecimentos para contribuir com o software em questão, destacando de forma mais específica o vocabulário da área de interesse. Segundo Gladcheft, Sanches e Silva (2001, p. 5), “Para a elaboração das questões desenvolvidas durante a fase de preparação do plano GQM, a técnica “Entrevista Estruturada” (Aquisição de Conhecimento Explícito) é incorporada junto ao processo de desenvolvimento do instrumento de avaliação.”

A parte de entrevista estruturada possui cinco etapas:

- a) planejamento: coleta todas as informações com o especialista da área;
- b) começo: motiva os participantes a uma comunicação ativa;
- c) corpo: baseia-se em um formulário com perguntas previamente selecionadas para evitar as perguntas aleatórias;
- d) fechamento: possibilita o especialista a revisar os pontos da entrevista;
- e) *follow up*: fase de tradução das informações em um formulário útil que será utilizado no desenvolvimento do software.

O foco maior deste trabalho é utilizar as métricas geradas pelos questionários para a verificação das hipóteses do experimento, e para servir de base para a criação de gráficos e cálculos estatísticos.

### 2.2.5 Validades dos dados do experimento

Após a execução do experimento, é necessário verificar se os dados gerados pelo estudo são válidos. Para isso, existem quatro tipos de validades: validade de conclusão, validade interna, validade de construção e a validade externa.

A validade de conclusão deve-se ao fato de verificar se os dados gerados pelo tratamento (variáveis independentes) são de alguma importância para o experimento. Segundo Travassos (2002, p. 12): “Durante a avaliação da validade de conclusão é necessário considerar os conceitos como a escolha do teste estatístico, a escolha do tamanho do conjunto dos participantes, a confiabilidade das medidas,[sic] e a confiabilidade da implementação dos tratamentos.”

A validade interna está mais voltada para os participantes, pois verifica se as respostas dadas não foram influenciadas por algum fator de confusão, ou seja, algum fator que pode distorcer o resultado do experimento. Segundo Travassos (2002, p. 13): “A validade interna

define se o relacionamento observado entre o tratamento e o resultado é causal, e não é o resultado da influência de outro fator que não é controlado ou mesmo não foi medido.”

A validade de construção verifica se a parte teórica do experimento foi formulada de forma correta. Segundo Travassos (2002, p. 13): “A validade de construção considera os relacionamentos entre a teoria e a observação, ou seja, se o tratamento reflete a causa bem e o resultado reflete o efeito bem.”

A validade externa representa a importância do resultado do experimento para o mundo real, segundo Travassos (2002, p. 13): “A validade externa define as condições que limitam a habilidade de generalizar os resultados de um experimento para a prática industrial.”

#### 2.2.6 Fases do processo de experimentação

Existem cinco fases para o processo de experimentação: definição, planejamento, operação, análise e interpretação e apresentação e empacotamento dos dados (TRAVASSOS, 2002).

Na fase de definição, são apresentados os objetivos, os parâmetros para a criação das métricas, no caso, as questões e a definição das hipóteses.

Na fase de planejamento são apresentados: a descrição da instrumentação, a seleção do contexto, os cadastros de projetos no caso de um objetivo de melhoria de processo e o cadastro dos participantes.

Na fase de operação são descritos a preparação e recursos utilizados e a elaboração do questionário e suas respectivas respostas. Segundo Travassos (2002, p. 23): “O aspecto mais importante da fase de execução é que a parte humana entra em jogo nesse momento. Os participantes devem ser preparados para a experimentação do ponto de vista moral e metodológica para evitar os resultados errôneos devido ao mal entendido ou falta de interesse.”

Na fase de análise e interpretação são realizados: a estatística descritiva, o teste de hipótese, as análises quantitativa e qualitativa e a descrição das validades.

Na fase de apresentação e empacotamento é realizada a apresentação dos dados obtidos no experimento através de relatórios ou gráficos e também o empacotamento dos dados. O empacotamento dos dados é importante para servir de suporte à repetição do experimento onde sua importância pode ser verificada segundo afirmação de Travassos (2002,

p. 23): “Com a repetição os pesquisadores adquirem o conhecimento adicional a respeito dos conceitos estudados, e recebem os resultados que são iguais ou diferentes dos resultados do experimento original.”

Pode-se observar que na fase de análise e interpretação dos dados, são abordados os termos de estatística descritiva e teste de hipótese, com o objetivo de um maior esclarecimento sobre estes assuntos, a próxima seção foi criada.

## 2.3 ESTATISTICA DIRECIONADA A EXPERIMENTOS

A realidade de um experimento em Engenharia de Software é a mesma para qualquer outro experimento que venha coletar dados, ou seja, existe a necessidade da redução destes dados para que a análise possa ser feita. Na literatura estatística, segundo Bereson, Levine e Stephan (2000, p. 119): “Em qualquer análise e/ou interpretação, várias medidas descritivas representando as propriedades de tendência central, variação e formato podem ser utilizadas para extrair e resumir as principais características do conjunto de dados”, porém, este trabalho utiliza as medidas de tendência central como a média aritmética, a mediana e a moda, e a medida de variação chamada de desvio padrão. Para a descrição da análise dos dados, a estatística descritiva é utilizada, e para a verificação das hipóteses, são utilizados os testes de hipótese. Optou-se por essas medidas por serem as mais utilizadas em ESE (TRAVASSOS, 2002).

### 2.3.1 Medidas de tendência central

Quando se fala em medida de tendência central, significa dizer que dentre um conjunto de dados, tem-se o objetivo de resumi-los em um ponto central para que eles possam ser trabalhados. Segundo Berenson, Levine e Stephan (2000, p. 119): “[...] para um conjunto de dados, em particular, geralmente se torna possível selecionar um valor típico ou **média** para descrever todo o conjunto. Tal valor descritivo típico é uma medida de **localização** ou **tendência central**.” A média aritmética, a mediana e a moda são medidas de tendência central.

### 2.3.2 Medidas de variação

Medidas de variação indicam o quanto um conjunto de dados diverge de outros. Segundo Berenson, Levine e Stephan (2000, p. 132): “**Varição** é a quantidade de **dispersão** ou *spread* nos dados. Dois conjuntos de dados podem divergir tanto na tendência central como na variação;”

### 2.3.3 Estatística descritiva

A Estatística descritiva nada mais é do que analisar os dados com base nos cálculos estatísticos e descrever o que se está observando. Segundo Berenson, Levine e Stephan (2000, p. 5): “A **estatística descritiva** pode ser definida como os métodos que envolvem a coleta, a apresentação e a caracterização de um conjunto de dados de modo a descrever apropriadamente as várias características deste conjunto.”

### 2.3.4 Teste de hipótese

O teste de hipótese serve para verificar se uma determinada hipótese é aceita ou rejeitada. Segundo Berenson, Levine e Stephan (2000, p. 330): “O teste de hipótese se inicia com alguma teoria, demanda ou afirmativa sobre determinado parâmetro de uma população.”

Em síntese, existem dois tipos de hipóteses, a hipótese nula e a hipótese alternativa. A hipótese nula é aquela que sempre é testada, ou seja, se caso esta hipótese for rejeitada, a hipótese alternativa será aceita, caso contrário, a hipótese nula será aceita.

Na realização de um teste de hipótese, são criadas regiões de rejeição e de não-rejeição, ou seja, se o resultado do teste de hipótese estiver entre a região de não-rejeição, a hipótese nula será aceita, caso contrário, ela será rejeitada. Em um teste de hipótese, dois erros podem ocorrer. O erro do tipo I quando a hipótese nula for rejeitada, onde na realidade deveria ser aceita, e o erro do tipo II quando a hipótese nula não for rejeitada, onde na realidade deveria ser rejeitada. O nível de significância indica a probabilidade de cometer um erro do tipo I, e o risco simbolizado pela letra grega  $\beta$ , é a probabilidade de se cometer um erro do tipo II. Segundo Berenson, Levine e Stephan (2000, p. 333): “Uma vez que o nível de

significância é especificado antes de o teste de hipóteses ser realizado, o risco de cometer um erro do tipo I,  $\alpha$ , está diretamente sob o controle do indivíduo que está realizando o teste.” Em cálculos estatísticos, costuma-se trabalhar com o nível de significância no valor de 0,05 ou menos.

É necessário também, se obter o número das amostras que estão sendo testadas, que dão origem ao grau de liberdade. Informações como o grau de liberdade e o grau de significância são necessários para a criação das regiões de rejeição e de não rejeição, a partir de uma tabela encontrada nas literaturas de estatística. Esta tabela pode ser visualizada na tabela 1.

Tabela 1 - Tabela do teste de hipótese

$\alpha \backslash P$	0,9	0,8	0,7	0,6	0,5	0,4	0,3	0,2	0,1	0,05	0,02	0,01	0,001
1	0,158	0,325	0,51	0,727	1	1,376	1,963	3,078	6,314	12,706	31,821	63,657	636,619
2	0,142	0,289	0,445	0,617	0,816	1,061	1,386	1,886	2,92	4,303	6,965	9,925	31,598
3	0,137	0,277	0,424	0,584	0,765	0,978	1,25	1,638	2,353	3,182	4,541	5,541	12,924
4	0,134	0,271	0,414	0,569	0,741	0,941	1,19	1,533	2,132	2,776	3,747	4,604	8,61
5	0,132	0,267	0,408	0,559	0,727	0,92	1,156	1,476	2,015	2,571	3,365	4,032	6,869
6	0,131	0,265	0,404	0,553	0,718	0,906	1,134	1,44	1,943	2,447	3,143	3,707	5,959
7	0,13	0,263	0,402	0,549	0,711	0,896	1,119	1,415	1,895	2,365	2,896	3,499	5,408
8	0,13	0,262	0,399	0,546	0,706	0,889	1,108	1,397	1,86	2,306	2,896	3,355	5,041
9	0,129	0,261	0,398	0,543	0,703	0,883	1,1	1,383	1,833	2,262	2,821	3,25	4,781
10	0,129	0,26	0,397	0,542	0,7	0,879	1,093	1,372	1,812	2,228	2,764	3,169	4,587
11	0,129	0,26	0,396	0,54	0,697	0,876	1,088	1,363	1,796	2,201	2,718	3,106	4,437
12	0,128	0,259	0,395	0,539	0,695	0,873	1,083	1,356	1,782	2,179	2,681	3,055	4,318
13	0,128	0,259	0,394	0,538	0,694	0,87	1,079	1,35	1,771	2,16	2,65	3,012	4,221
14	0,128	0,258	0,393	0,537	0,692	0,868	1,076	1,345	1,761	2,145	2,624	2,977	4,14
15	0,128	0,258	0,393	0,536	0,691	0,866	1,074	1,341	1,753	2,131	2,602	2,947	4,073
16	0,128	0,258	0,392	0,535	0,69	0,865	1,071	1,337	1,746	2,12	2,583	2,921	4,015
17	0,128	0,257	0,392	0,534	0,689	0,863	1,069	1,333	1,74	2,11	2,567	2,898	3,965
18	0,127	0,257	0,392	0,534	0,688	0,862	1,067	1,33	1,734	2,101	2,552	2,878	3,922
19	0,127	0,257	0,391	0,533	0,688	0,861	1,066	1,328	1,729	2,093	2,539	2,861	3,883
20	0,127	0,257	0,391	0,533	0,687	0,86	1,064	1,325	1,725	2,086	2,528	2,845	3,85
21	0,127	0,257	0,391	0,532	0,686	0,859	1,063	1,323	1,721	2,08	2,518	2,831	3,819
22	0,127	0,256	0,39	0,532	0,686	0,858	1,061	1,321	1,717	2,074	2,508	2,819	3,792
23	0,127	0,256	0,39	0,532	0,685	0,858	1,06	1,319	1,714	2,069	2,5	2,807	3,767
24	0,127	0,256	0,39	0,531	0,685	0,857	1,059	1,318	1,711	2,064	2,492	2,797	3,745
25	0,127	0,256	0,39	0,531	0,684	0,856	1,058	1,316	1,708	2,06	2,485	2,787	3,726
26	0,127	0,256	0,39	0,531	0,684	0,856	1,058	1,315	1,706	2,056	2,479	2,779	3,707
27	0,127	0,256	0,389	0,531	0,684	0,856	1,057	1,314	1,703	2,052	2,473	2,771	3,69
28	0,127	0,256	0,389	0,53	0,683	0,856	1,056	1,313	1,701	2,048	2,467	2,763	3,674
29	0,127	0,256	0,389	0,53	0,683	0,854	1,055	1,311	1,699	2,045	2,462	2,756	3,659
30	0,127	0,256	0,389	0,53	0,683	0,854	1,055	1,31	1,697	2,042	2,457	2,75	3,646
40	0,126	0,255	0,388	0,529	0,681	0,851	1,05	1,303	1,684	2,021	2,423	2,704	3,551
60	0,126	0,254	0,387	0,527	0,679	0,848	1,046	1,296	1,671	2	2,39	2,66	3,46
120	0,126	0,254	0,386	0,526	0,677	0,845	1,041	1,289	1,658	1,98	2,358	2,617	3,373
1	0,126	0,253	0,385	0,524	0,674	0,842	1,036	1,282	1,645	1,96	2,326	2,576	3,291

Fonte: Berenson, Levine e Stephan (2000, p. 738).

A primeira coluna da tabela é correspondente ao grau de liberdade e a primeira linha da tabela corresponde ao grau de significância. No caso de um grau de liberdade de 23 e um grau de significância de 5%, a região de rejeição seria de -2,069 e de +2,069, ou seja, a hipótese nula será aceita se o resultado do teste estatístico ficar entre os valores -2,069 e +2,069.

## 2.4 TRABALHOS CORRELATOS

Um trabalho que pode ser relatado é o de Gross (2001), que mostra toda a

funcionalidade e potencial da abordagem GQM, relatando o suporte à definição e implantação de metas. Este trabalho está direcionado para pesquisas de qualidade de software, onde foi desenvolvido um protótipo de um software de apoio que auxilia as etapas de desenvolvimento e execução do plano GQM.

Seguindo a área de GQM, pode ser relacionado o estudo da avaliação da qualidade de software em Kasburg (2001), no qual é abordada uma forma de avaliação de produtos de software de gestão integrada. O trabalho usa como base as normas técnicas de qualidade e o GQM. O protótipo foi desenvolvido para demonstrar a aplicação da avaliação de software.

Ainda sobre GQM que é um dos pilares do desenvolvimento deste trabalho, pode ser citada a ferramenta MedPlan, que é uma ferramenta para elaboração de planos de medição. Segundo Schnaider et al. (2004): “A proposta da ferramenta *MedPlan* é apoiar a elaboração dos Planos de Medição da Organização e do projeto. Baseando-se no método GQM, esta ferramenta disponibiliza ao usuário o conhecimento sobre objetivos, questões, métricas e procedimentos de coleta, armazenamento e análise de dados a serem utilizados.” Uma tela da ferramenta pode ser visualizada na figura 3.

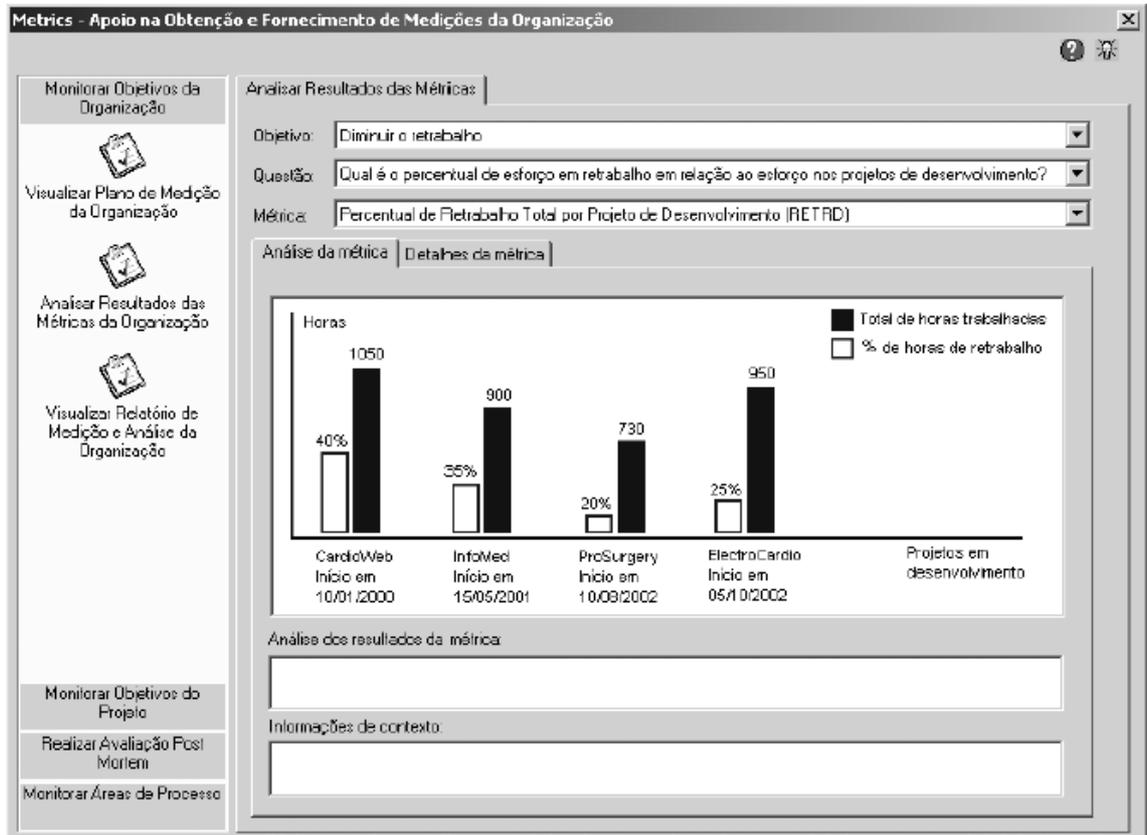
The screenshot shows a window titled "MedPlan - Apoio no Planejamento de Medição da Organização". The interface is divided into two main sections. On the left is a vertical sidebar with a list of menu items, each accompanied by a small icon of a document with a checkmark. The menu items are: "Elaborar Plano de Medição da Organização", "Definir Plano de Medição da Organização", "Identificar Objetivos de Medição da Organização", "Identificar Questões de Medição da Organização", "Identificar Métricas da Organização", and "Visualizar Plano de Medição da Organização". The main area on the right is titled "Definir Plano de Medição da Organização" and contains several input fields: "Objeto de análise:" (a large text box), "Propósito:" (a dropdown menu currently showing "Conhecer"), "Com respeito:" (a text box), "Ponto de vista:" (a text box), and "Contexto:" (a text box).

Fonte: Schnaider et al. (2004).

Figura 3 - Tela do sistema *MedPlan*

A ferramenta *Metrics* também pode ser relacionada como ferramenta correlata. Porém,

o que se difere da ferramenta *MedPlan*, é que ela, além de obter os dados, trabalha com os dados coletados. Segundo Schnaider et al. (2004): “O objetivo da ferramenta *Metrics* é apoiar a obtenção e o fornecimento dos resultados das medições realizadas de acordo com as necessidades e objetivos dos Planos de Medição da Organização e dos projetos.” Uma tela do *Metrics* pode ser visualizada na figura 4.



Fonte: Schnaider et al. (2004).

Figura 4 - Tela do sistema Metrics

Pode-se verificar que a maioria das ferramentas apresentadas, são de utilização do plano GQM, para obtenção de métricas, e manipulação dos dados coletados, que podem ser relacionadas a alguns tópicos deste trabalho como por exemplo a seção 2.2.4. Diretamente, como suporte a experimentos em Engenharia de Software, não foram encontrados trabalhos correlatos descrevendo ferramentas.

### 3 DESENVOLVIMENTO DO TRABALHO

Este capítulo apresenta os requisitos funcionais, a especificação, a implementação e os resultados e discussões.

#### 3.1 REQUISITOS PRINCIPAIS DO PROBLEMA A SER TRABALHADO

Os requisitos da ferramenta desenvolvida neste trabalho foram elaborados com base na fundamentação teórica sobre Engenharia de Software Experimental encontrada na seção 2.2 e sobre estatística direcionada a experimentos na seção 2.3. Optou-se por focar apenas requisitos de experimentos direcionados a melhoria de processo de software. Os requisitos funcionais identificados são:

- a) registro de experimentos: este requisito foi criado a partir da necessidade de empacotamento dos dados do experimento;
- b) cadastro de grupo de projetos: como a ferramenta desenvolvida neste trabalho está direcionada á melhoria de processo, este requisito foi criado para possibilitar o cadastro dos grupos de projetos de uma empresa;
- c) cadastro de projetos: o cadastro de projetos possibilita o cadastro dos projetos ou casos de uso de uma empresa. Este requisito foi criado com base na melhoria de processo, partindo deste princípio, pode-se deduzir que existam projetos dentro de um processo de desenvolvimento de uma empresa. Os casos de uso foram citados pelo motivo do experimentador poder escolher casos de uso de uma determinada parte de um sistema para avaliar, ao invés do processo como um todo;
- d) definição dos objetivos: este requisito foi criado para possibilitar o registro dos objetivos do experimento, para posterior apresentação dos dados;
- e) registro de questões e métricas: como abordado na seção 2.2.4, a elaboração das questões são muito importantes para a aquisição das métricas. Partindo deste princípio que este requisito foi criado;
- f) definição das hipóteses: as hipóteses de um experimento são definidas para que no final do estudo, se possa verificar se a teoria formulada pela hipótese foi aceita,

- para atender esta necessidade, este requisito foi criado;
- g) descrição da instrumentação: este requisito foi criado para atender a necessidade da descrição do modo como o experimento foi conduzido;
  - h) seleção do contexto: este requisito foi criado com base na fundamentação teórica descrita na seção 2.2.1, que explica a necessidade da seleção de um contexto em um experimento;
  - i) cadastro de participantes: partindo do princípio de que os participantes são os responsáveis pela resposta do questionário, este requisito foi criado para fornecer ao experimentador, algumas informações adicionais sobre os participantes do experimento, como por exemplo, o grau de experiência que determinado participante tem na sua área de atuação;
  - j) resposta das questões: para viabilizar a respostas dos participantes através de formulários de respostas, este requisito foi criado;
  - k) registro da descrição das validades: este requisito foi criado com base na fundamentação teórica da seção 2.2.5;
  - l) descrição da preparação e dos recursos utilizados: esta é mais uma fase do experimento, onde o experimentador indica o modo como o estudo foi conduzido. É uma funcionalidade descritiva, porém, importante para a documentação do experimento;
  - m) estatística descritiva: este requisito foi criado a partir da fundamentação teórica encontrada na seção 2.3.3, que indica a necessidade do uso de cálculos estatísticos com a função de redução dos dados do experimento;
  - n) teste de hipótese: para a verificação de uma teoria, é preciso realizar o teste de hipótese para testar se uma determinada hipótese foi aceita ou rejeitada, com este propósito que este requisito foi criado. Maiores detalhes sobre o teste de hipótese podem ser encontrados na seção 2.3.4;
  - o) análise quantitativa: este requisito foi criado para suprir a necessidade da análise do volume dos dados do experimento, no enfoque quantitativo;
  - p) análise qualitativa: segue o mesmo princípio do tópico o, porém no âmbito qualitativo;
  - q) resultados em forma de gráficos e relatórios: a seção 2.2.6, apresenta a fase final do processo de experimentação, que é a apresentação dos dados do experimento. Este requisito foi criado para oferecer ao experimentador, a visão geral dos resultados do experimento.

## 3.2 ESPECIFICAÇÃO

Nesta seção do trabalho são apresentados os diagramas utilizados para a especificação da ferramenta, dentre os quais, foram escolhidos: o diagrama de casos de uso, o diagrama de atividades, o diagrama de classes e os diagramas conceitual e físico do banco de dados.

As ferramentas utilizadas para a especificação foram o *Enterprise Architect 4.5*, para a criação dos diagramas da UML e a ferramenta *Power Designer* para a modelagem do banco de dados.

A partir da seção 3.2.1, estes diagramas podem ser visualizados com maior detalhamento.

### 3.2.1 Diagramas de caso de uso

Na figura 5 é apresentado o diagrama de casos de uso. Os atores que fazem parte deste diagrama são o usuário e o participante. As descrições de cada caso de uso são apresentadas no apêndice A deste trabalho.

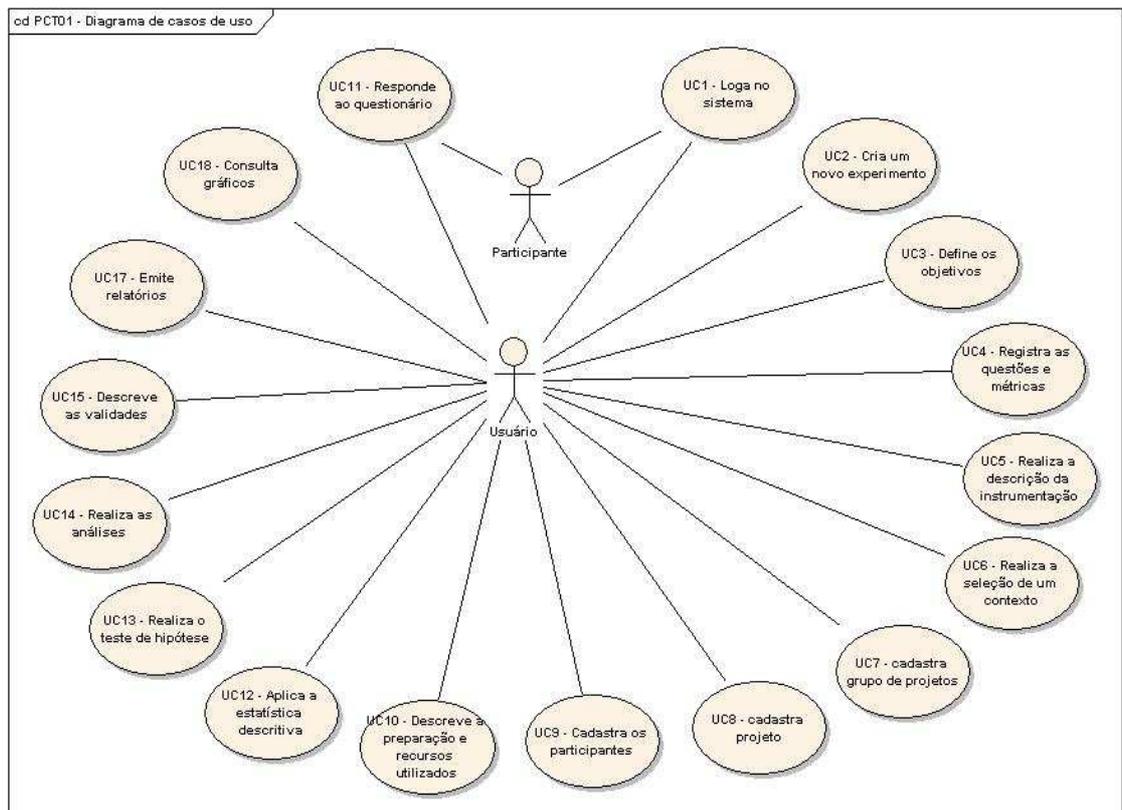


Figura 5 - Diagrama de casos de uso

## 3.2.2 Diagrama de atividades

Na figura 6, é apresentado o diagrama de atividades do processo, que mostra passo a passo as atividades executadas no processo de experimentação.

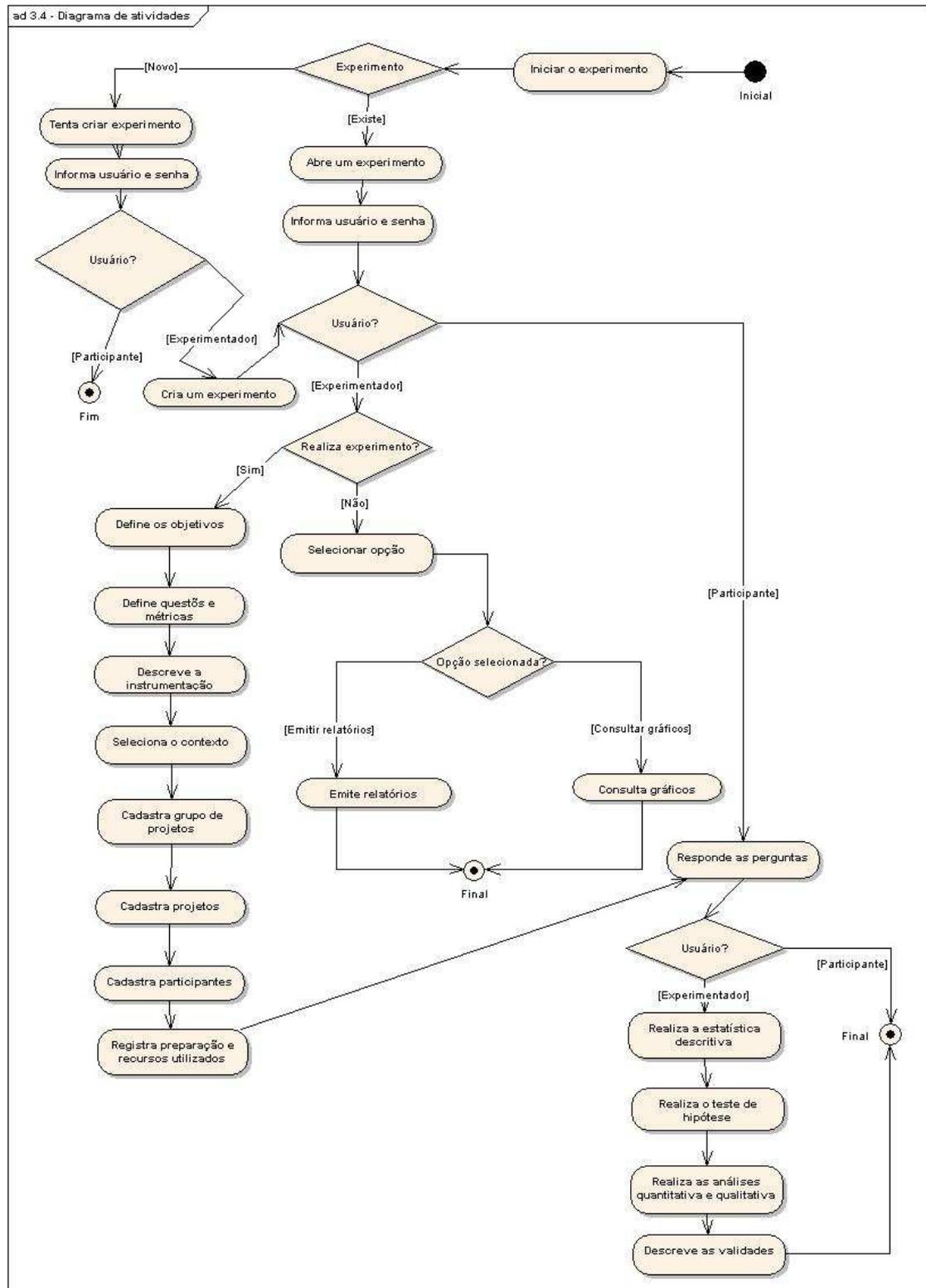


Figura 6 - Diagrama de atividades

O sistema começa com a necessidade do usuário criar ou abrir um novo experimento. No caso da criação de um novo experimento, somente um usuário do tipo experimentador pode executar esta operação. No caso da opção por abrir um experimento, se o usuário for do tipo participante, ele somente poderá responder as perguntas formuladas pelo experimentador, no caso do usuário for do tipo experimentador, ele poderá executar os passos da definição dos objetivos até a resposta das perguntas ou optar por emitir um relatório ou consultar um gráfico. No caso do experimentador optar por realizar o experimento, pode realizar ainda, a estatística descritiva, o teste de hipótese, as análises e as validades para finalmente o fluxo terminar.

### 3.2.3 Diagrama de classes

Na figura 7, é apresentado o diagrama de classes da ferramenta. Este diagrama possui os métodos que são invocados pelas rotinas de interface com o usuário.

Como pode ser observado, existem dois métodos comuns a todas as classes, que são:

- a) *Create*: serve para criar uma instância da classe;
- b) *Destroy*: serve para liberar a instância da classe.

O atributo “loOnAdiciona” também é utilizado em várias classes, e tem o mesmo objetivo: armazena o evento chamado no momento em que o usuário invoca uma rotina de inserção de dados. Como estes métodos e atributos são comuns em várias classes, não serão descritos a partir da seção 3.2.3.1.

Para acesso a dados, foram desenvolvidas classes próprias, porém, não foram ilustradas na figura 7 por não serem classes de regras de negócio, sendo assim, algumas classes não tiveram ligações e foram retiradas do diagrama.

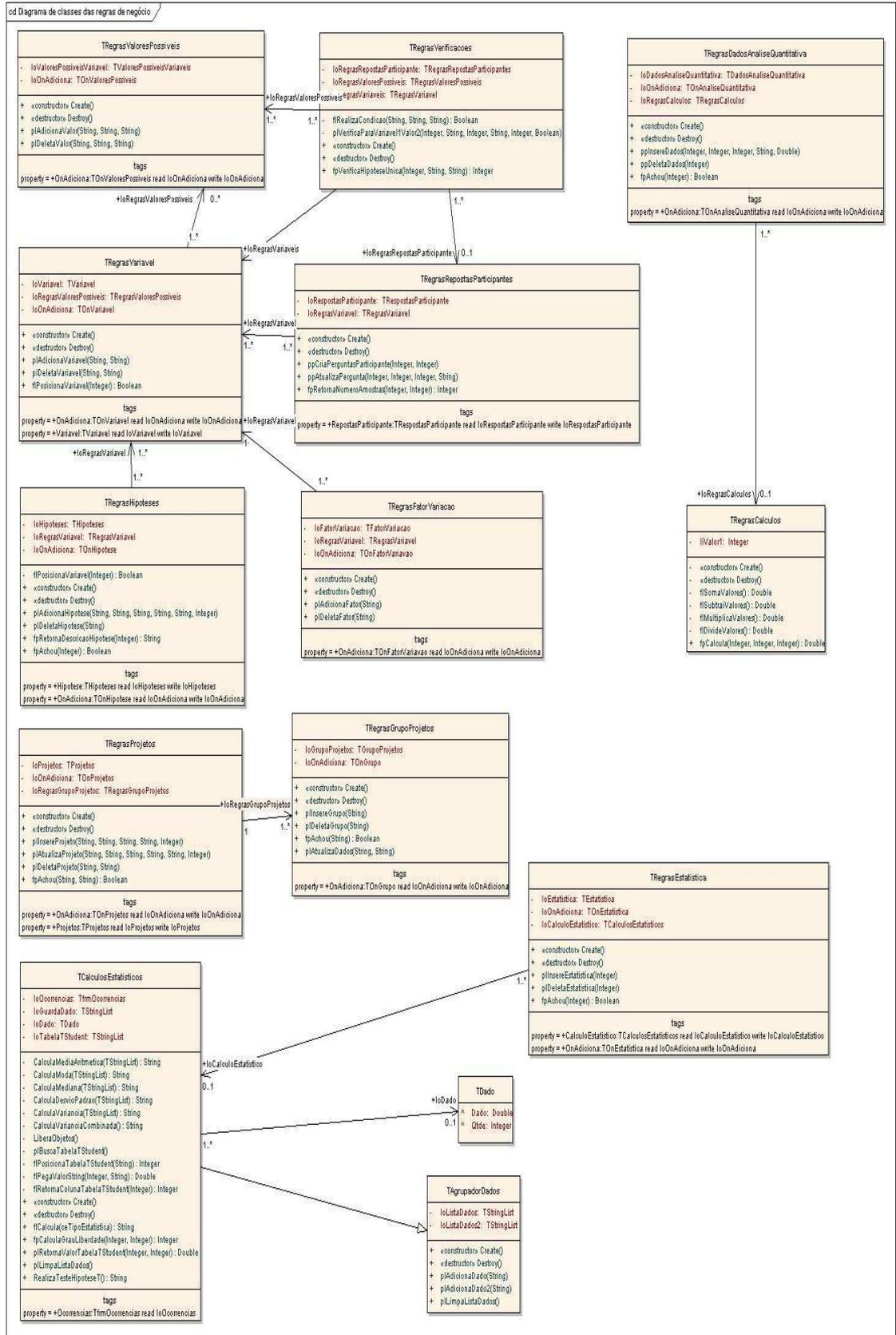


Figura 7 - Diagrama de classes

### 3.2.3.1 Descrição da classe “TRegrasFatorVariacao”

A classe “TRegrasFatorVariacao” esta relacionada a classe “TRegrasVariavel” pois ela dá subsídios para a criação das variáveis no sistema. A classe “TRegrasFatorVariacao” pode ter várias variáveis ligadas a ela, e a sua estrutura pode ser visualizada no quadro 3.

<b>Classe TRegrasFatorVariacao</b>	
<b>Atributos</b>	<b>Descrição</b>
loFatorVariacao : TFatorVariacao	Contém o acesso a tabela FatorVariacao
loRegrasVariavel : TRegrasVariavel	Regras de negócio da variável
<b>Métodos</b>	<b>Descrição</b>
plAdicionaFator(String)	Adiciona um fator de variação na base de dados
plDeletaFator(String)	Deleta o fator de variação passado como parâmetro da base de dados

Quadro 3 - Atributos e métodos da classe "RegrasFatorVariacao"

### 3.2.3.2 Descrição da classe “TRegrasVariavel”

Esta classe é responsável pela criação das variáveis (perguntas) do sistema, e pode ser visualizada no quadro 4.

<b>Classe TRegrasVariavel</b>	
<b>Atributos</b>	<b>Descrição</b>
loVariavel	Acessa a tabela Variavel
loRegrasValoresPossiveis : TRegrasValoresPossiveis	Possíveis valores da variável
<b>Métodos</b>	<b>Descrição</b>
plAdicionaVariavel(String, String)	Adiciona uma variável na base de dados
plDeletaVariavel(String, String)	Deleta a variável da base
flPosicionaVariavel(Integer) : Boolean	Efetua um select no banco

Quadro 4 - Atributos e métodos da classe "TRegrasVariavel"

### 3.2.3.3 Descrição da classe “TRegrasValoresPossiveis”

A classe “TRegrasValoresPossiveis” representa os possíveis valores que uma classe pode assumir, o uso desta classe não é obrigatório e pode ser visualizado no quadro 5.

Classe TRegrasValoresPossiveis	
Atributos	Descrição
loValoresPossiveisVariavel : TValoresPossiveisVariavel	Acessa a tabela ValoresPossiveisVariavel
Métodos	Descrição
plAdicionaValor(String, String, String)	Adiciona um possível valor na base de dados
plDeletaValor(String, String, String)	Deleta um possível valor da base de dados

Quadro 5 - Métodos e atributos da classe "TRegrasValoresPossiveisVariavel"

### 3.2.3.4 Descrição da classe “TRegrasVerificações”

A classe “TRegrasVerificações” possibilita a verificação de algumas condições feitas pelo usuário, quando o mesmo objetiva a busca de uma quantidade de dados. A descrição desta classe pode ser vista no quadro 6.

Classe TRegrasVerificacoes	
Atributos	Descrição
loRegrasRespostasParticipante : TRegrasRespostasParticipante	Regras das respostas dos participantes
loRegrasValoresPossiveis : TRegrasValoresPossiveis	Valores possíveis da variável
loRegrasVariavel : TRegrasVariavel	regras das variáveis do sistema
Métodos	Descrição
flRealizaCondição(String, String, String) : Boolean	Realiza uma condição
plVerificaparaVariavel1Valor2(Integer, String, Integer, String, Integer, Boolean)	Faz uma verificação de condição entre uma variável e um valor
fpVerificaHipoteseUnica(integer, String, String)	Verifica uma condição

Quadro 6 – Atributos e métodos da classe “TRegrasVerificações”

### 3.2.3.5 Descrição da classe “TRegrasRepostasParticipante”

A classe “TRegrasRespostaParticipante” é responsável pela manipulação das respostas dadas pelos participantes, e a sua descrição pode ser visualizada no quadro 7.

<b>Classe TRegrasRespostasParticipante</b>	
<b>Atributos</b>	<b>Descrição</b>
loRespostasParticipante : TRespostasParticipante	Acessa a tabela RespostasParticipante
loRegrasVariavel : TRegrasVariavel	regras das variáveis do sistema
<b>Métodos</b>	<b>Descrição</b>
ppCriaPerguntasParticipante(Integer, Integer)	cria as perguntas para o participante responder
ppAtualizaPergunta(Integer, Integer, Integer, String)	Atualiza a resposta do participante na base de dados
fpRetornaNumeroAmostras(Integer, Integer) : Integer	Verifica uma condição através de uma hipótese

Quadro 7 - Atributos e métodos da class "TRegrasRepostasParticipante "

### 3.2.3.6 Descrição da classe “TRegrasHipoteses”

A classe “TRegrasHipoteses” está relacionada a classe “TRegrasVariavel”, por que na formulação das hipóteses, o usuário informa duas variáveis para que o sistema possa fazer o teste de hipótese em cima delas. A descrição desta classe pode ser visualizada no quadro 8.

<b>Classe TRegrasHipoteses</b>	
<b>Atributos</b>	<b>Descrição</b>
loHipoteses : thipoteses	Acessa a tabela Hipoteses
loRegrasVariavel : tregrasVariavel	regras das variáveis do sistema
<b>Métodos</b>	<b>Descrição</b>
flPosicionaVariavel(Integer) : Boolean	Realiza um select no banco
plAdicionaHipotese(String, String, String)	Adiciona uma hipótese na base
plDeletaHipotese(String)	Deleta uma hipótese da base
fpRetornaDescricaoHipotese(integer) : String	diz se a hipótese foi aceita
fpAchou(Integer) : Boolean	Procura uma hipótese no banco

Quadro 8 - Atributos e métodos da classe “TRegrasHipoteses”

### 3.2.3.7 Descrição da classe “TRegrasProjetos”

A classe “TRegrasProjetos” é a classe responsável pela manipulação dos projetos ou casos de uso do sistema, e a sua descrição pode ser visualizada no quadro 9.

<b>Classe TRegrasProjetos</b>	
<b>Atributos</b>	<b>Descrição</b>
loProjetos : Tprojetos	Acessa a tabela projetos
loregrasGrupoProjetos : TRegrasgrupoProjetos	Regras dos grupos de projetos
<b>Métodos</b>	<b>Descrição</b>
plInsereProjeto(String, String, String, String, Integer)	Insere um projeto no banco
plAtualizaProjeto(String, String, String, String, Integer)	Atualiza dados do projeto
plDeletaProjeto(String, String)	Deleta o projeto do banco
fpAchou(String, String) : Boolean	Procura um projeto no banco

Quadro 9 - Atributos e métodos da classe “TRegrasProjetos”

### 3.2.3.8 Descrição da classe “TRegrasGruposProjetos”

A classe “TRegrasGruposProjetos” é responsável pelo agrupamento dos projetos e pode ser visualizada no quadro 10.

<b>Classe TRegrasGruposProjetos</b>	
<b>Atributos</b>	<b>Descrição</b>
loGrupoProjetos : TGupoprojetos	Acessa a tabela Grupoprojetos
<b>Métodos</b>	<b>Descrição</b>
plInsereGrupo(String)	Insere um grupo no banco
plDeletaGrupo(String)	Deleta um grupo do banco
fpAchou(String) : Boolean	procura um grupo no banco
plAtualizaDados(String, String)	Atualiza os dados do grupo

Quadro 10 - Atributos e métodos da classe “TRegrasGruposProjetos”

### 3.2.3.9 Descrição da classe “TRegrasExperimento”

A classe “TRegrasExperimento” é a classe que manipula os dados gerais do

experimento, como as descrições dos objetivos, e a sua descrição pode ser encontrada no quadro 11.

<b>Classe TRegrasExperimento</b>	
<b>Atributos</b>	<b>Descrição</b>
loExperimento : tExperimento	Acessa a tabela Experimento
loLoginUsuario : tLoginUsuario	Acessa a tabela LoginUsuario
<b>Métodos</b>	<b>Descrição</b>
plCriaObjetos	Cria as classes
plDestroyObjetos	Libera a criação das classes
DeletaExperimento(integer, string) : Boolean	Deleta o experimento do banco
InserelInformacoesExperimento(Integer, String, String, String, String, String, Boolean)	Inserel informações gerais do experimento

Quadro 11 - Atributos e métodos da classe “TRegrasExperimento”

### 3.2.3.10 Descrição da classe “TRegrasContexto”

A classe “TRegrasContexto” manipula os valores do contexto selecionado para o experimento e a sua descrição pode ser visualizada no quadro 12.

<b>Classe TRegrasContexto</b>	
<b>Atributos</b>	<b>Descrição</b>
loContexto	Acessa a tabela Contexto
<b>Métodos</b>	<b>Descrição</b>
InserelInformacoesContexto(Integer, Integer, Integer, Integer)	Inserel a seleção do contexto

Quadro 12 - Atributos e métodos da classe “TRegrasContexto”

### 3.2.3.11 Descrição da classe “TRegrasDadosAnaliseQualitativa”

A descrição da classe “TRegrasDadosAnaliseQualitativa” pode ser vista no quadro 13.

<b>Classe TRegrasDadosAnaliseQualitativa</b>	
<b>Atributos</b>	<b>Descrição</b>
DadosAnaliseQualitativa: TDadosAnaliseQualitativa	Acessa a tabela DadosAnaliseQualitativa
<b>Métodos</b>	<b>Descrição</b>
ppInsererDados(String, Integer)	Inserer os dados da análise qualitativa
ppDeletarDados(Integer)	Deleta os dados da análise qualitativa
fpAchoo(Integer) : Boolean	Procura uma análise no banco

Quadro 13 - Atributos e métodos da classe “TRegrasDadosAnaliseQualitativa”

### 3.2.3.12 Descrição da classe “TRegrasDescricaoValidades”

A classe “TRegrasDescricaoValidades” é responsável pela pelos dados das descrições das validades e sua descrição pode ser encontrada no quadro 14.

<b>Classe TRegrasDescricaoValidades</b>	
<b>Atributos</b>	<b>Descrição</b>
loDescricaoValidades : tDescricaoValidades	Acessa a tabela DescricaoValidades
<b>Métodos</b>	<b>Descrição</b>
ppInsererDescricao(ceTipoValidade, String)	Inserer uma validade no sistema
ppAtualizaDescricao(ceTipovalidade, String)	Atualiza informações da validade
fpAchoo(ceTipoValidade) : Boolean	Procura uma validade no banco

Quadro 14 - Atributos e métodos da classe “TRegrasDescricaoValidades”

### 3.2.3.13 Descrição da classe “TRegrasTesteHipotese”

A tabela “TRegrasTesteHipotese” é responsável pela realização do teste de hipótese, para a verificação das hipóteses, e a sua descrição pode ser visualizada no quadro 15.

<b>Classe TRegrasTesteHipotese</b>	
<b>Atributos</b>	<b>Descrição</b>
loTesteHipotese : ttesteHipotese	Acessa a tabela TesteHipotese
<b>Métodos</b>	<b>Descrição</b>
plInsere(Integer, Integer, Integer, Integer, Integer, Double, Double, Double, String)	Insere um teste de hipótese no banco
plDeleta(Integer, Integer, Integer, Integer)	Deleta um teste de hipótese do banco
fpAchou(Integer, Integer, Integer, Integer) : Boolean	Procura um teste de hipótese

Quadro 15 - Atributos e métodos da classe “TRegrasTesteHipotese”

### 3.2.3.14 Descrição da classe “TRegrasParticipante”

Esta classe corresponde aos métodos de acesso aos dados dos participantes, e sua descrição pode ser encontrada no quadro 16.

<b>Classe TRegrasParticipante</b>	
<b>Atributos</b>	<b>Descrição</b>
loParticipante : Tparticipante	Acessa a tabela Participante
<b>Métodos</b>	<b>Descrição</b>
fpAchou(integer) : Boolean	Procura um participante no banco
plinsereParticipante(String, Integer, Integer, Integer, Integer, Integer)	Insere um participante no banco
pldeletaParticipante(Integer)	Deleta um participante do banco
plAtualizadadosParticipante(Integer, String, Integer, Integer, Integer, Integer)	Atualiza os dados do participante como os níveis de experiência

Quadro 16 - Atributos e métodos da classe “TRegrasParticipante”

### 3.2.3.15 Descrição da classe “TRegrasEstatistica”

A classe “TRegrasEstatistica” é responsável pelos valores de estatística e sua descrição pode ser encontrada no quadro 17.

<b>Classe TRegrasEstatistica</b>	
<b>Atributos</b>	<b>Descrição</b>
loEstatistica : Testatistica	Acessa a tabela Estatistica
loCalculoestatistico : tCalculoestatistico	Acessa os cálculos de estatística
<b>Métodos</b>	<b>Descrição</b>
pInsereEstatistica(Integer)	Insere uma estatística no banco
pDeletaEstatistica(integer)	Deleta uma estatística do banco
fpAchou(Integer) : Boolean	Acha uma estatística

Quadro 17 - Atributos e métodos da classe “TRegrasEstatistica”

### 3.2.3.16 Descrição da classe “TRegrasDescricaoAnalises”

Esta classe é responsável pela descrição da análise quantitativa e qualitativa e sua descrição pode ser encontrada no quadro 18.

<b>Classe TRegrasDescricaoAnalises</b>	
<b>Atributos</b>	<b>Descrição</b>
loDescricaoAnalises : TDescricaoAnalises	Acessa a tabela DescricaoAnalises
<b>Métodos</b>	<b>Descrição</b>
pInsereDescricao(ceTipoAnalise)	Insere uma análise no sistema
ppAtualizaDescricao(ceTipoAnalise)	Atualiza a descrição da análise no sistema
fpAchou(ceTipoAnalise) : Boolean	Acha uma análise no banco

Quadro 18 - Atributos e métodos da classe “TRegrasDescricaoAnalises”

### 3.2.3.17 Descrição da classe “TRegrasCalculos”

Esta classe é responsável pelos cálculos matemáticos do sistema, e sua descrição pode ser encontrada no quadro 19.

<b>Classe TRegrasCalculos</b>	
<b>Atributos</b>	<b>Descrição</b>
liValor1 : Integer	variável de auxílio para cálculos
<b>Métodos</b>	<b>Descrição</b>
fISomaValores : Double	Soma dois valores
fISubtraivalores : Double	Subtrai dois valores
fIMultiplicavalores : Double	Multiplca dois valores
fIDivideValores : Double	Divide dois valores
fpCalcula(Integer, Integer, Integer)	Método que chama os cálculos

Quadro 19 - Atributos e métodos da classe “TRegrasCalculos”

### 3.2.3.18 Descrição da classe “TRegrasDadosAnaliseQuantitativa”

Esta classe é responsável pelos dados da análise quantitativa e sua descrição pode ser encontrada no quadro 20.

<b>Classe TRegrasDadosAnaliseQuantitativa</b>	
<b>Atributos</b>	<b>Descrição</b>
lodadosAnaliseQuantitativa : TDadosAnaliseQuantitativa	Acessa a tabela DadosAnaliseQuantitativa
loregrasCalculos : Tregascalculos	Acessa os cálculos do sistema
<b>Métodos</b>	<b>Descrição</b>
ppInseredados(Integer, Integer, Integer, String, Double)	Insere uma analise quantitativa
ppDeletadados(Integer)	Deleta uma analise quantitativa
fpAchou(Integer) : Boolean	Procura uma analise quantitativa no banco

Quadro 20 - Atributos e métodos da classe “TRegrasDadosAnaliseQuantitativa”

### 3.2.3.19 Descrição da classe “TAggrupadordados”

Esta classe é responsável pelo agrupamento de dados para efeito de cálculos estatísticos e sua descrição pode ser encontrada no quadro 21.

<b>Classe TAgupadorDados</b>	
<b>Atributos</b>	<b>Descrição</b>
loListaDados : TStringList	Agrupador de dados da primeira amostra
loListadados2 : TStringList	Agrupador de dados da segunda amostra
<b>Métodos</b>	<b>Descrição</b>
plAdicionado(String)	Inserir um dado da primeira amostra
plAdicionado2(String)	insere um dado da segunda amostra
plLimpaListaDados	Limpa os valores da primeira e segunda amostra

Quadro 21 - Atributos e métodos da classe “TAgupadordados”

### 3.2.3.20 Descrição da classe “TCalculosEstatisticos”

Esta classe é responsável por todos os cálculos estatísticos utilizados no sistema e a sua descrição pode ser encontrada no quadro 22.

<b>Classe Tcalculosestatisticos</b>	
<b>Atributos</b>	<b>Descrição</b>
loOcorrencias : tfrmOcorrencias	Mostra ocorrências se houver
loguardadado : tStringList	Atributo de auxílio para os cálculos
lodado : Tdado	Atributo de auxílio para os cálculos
loTabelaTStudent : TStringList	Atributo que contém a tabela student
<b>Métodos</b>	<b>Descrição</b>
CalculaMediaAritmetica(TStringList) : String	Calcula a média
CalculaModa(tStringList) : String	Calcula a moda os valores
CalculaMediana(TStringList) : String	Calcula a mediana
CalculaDesvioPadrao(tStringList) : String	Calcula o desvio padrão
CalculaVariancia(TStringList) : String	Calcula a variância
CalculaVarianciaCombinada() : String	Calcula a variância combinada
liberaObjetos()	Libera os objetos alocados
plBuscaTabelaTStudent()	Busca a tabela tstudent
flPosicionatabelaTStudent(String) : Integer	Posiciona um valor da tabela Tstudent
flPegaValorString(integer, String) : Double	Pega um valor da coluna da tabela Tstudent
flRetornaColunatabelaTStudent(Integer) : Integer	Retorna o número de uma coluna da tabela
flCalcula(ceTipoEstatistica) : String	Calcula a estatística
fpCalculaGrauLiberdade(Integer, Integer) : Integer	Calcula o grau de liberdade conforme número de amostras
plRetornaValorTabelaTStudent(Integer, Integer) : Double	chama o método flPegaValorString
plLimpaListaDados()	Limpa as amostras
RealizatesteHipotese() : String	Realiza o teste para verificar se a hipótese é aceita ou não

Quadro 22 - Atributos e métodos da classe “TCalculosEstatisticos”

### 3.2.4 Diagrama de entidade e relacionamento conceitual

Na figura 8, é apresentado o diagrama de entidade e relacionamento conceitual, indicando os relacionamentos das tabelas da ferramenta.

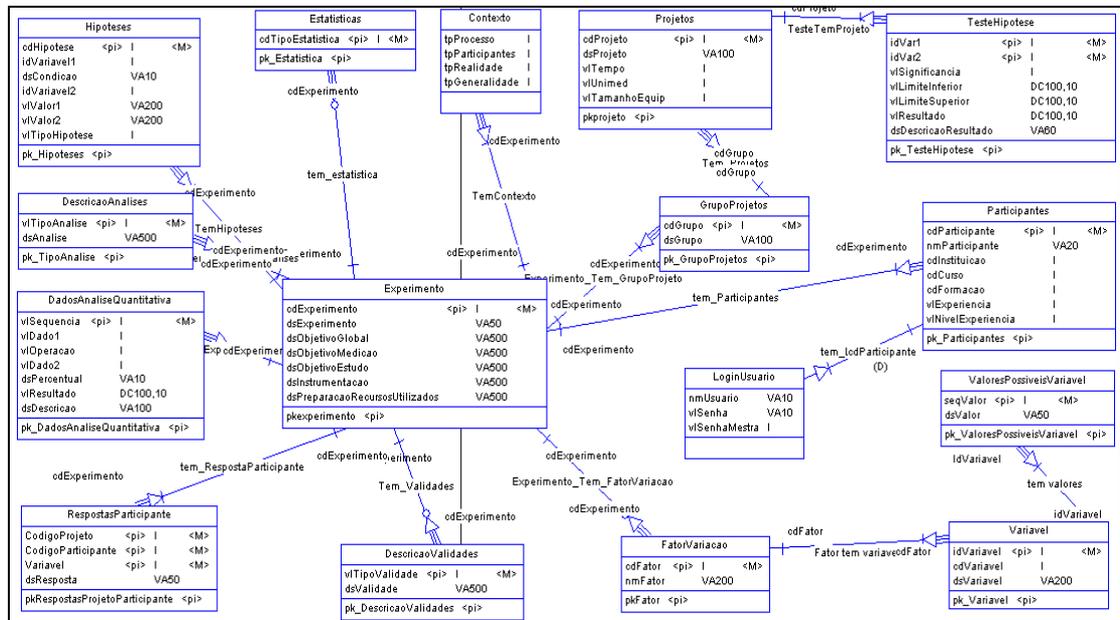


Figura 8 - Diagrama entidade relacionamento conceitual

A tabela experimento tem o maior número de relacionamentos, pois todas as tabelas do sistema, tem por obrigação ter o campo cdexperimento, para evitar duplicação de dados no caso da criação de mais experimentos. Existe a ligação do experimento aos projetos pelas tabelas grupoprojetos e projetos. Existe o relacionamento das variáveis do experimento através das tabelas fatorvariacao, variavel e valorespossiveisvariavel. A tabela repostasparticipante não foi relacionada a outras tabelas por motivo de alguns problemas ocasionados pelo relacionamento, um exemplo, foi a duplicação de alguns campos. A tabela participantes está relacionada a tabela experimento, no sistema, os valores dos campos cdInstituicao, cdCurso, cdFormacao, vlExperiencia e vlNivelExperiencia foram criados de forma fixa se baseando nos artigos de Engenharia de Software Experimental. A tabela loginusuario está relacionada a tabela participantes por um relacionamento de dependência para que a tabela loginusuario contenha os campos cdExperimento e cdParticipante como chave primária. A tabela testehipotese está relacionada à tabela projetos, pois os testes de hipótese vão ser feitos por projeto.

### 3.2.5 Diagrama de entidade e relacionamento físico

Na figura 9, pode ser visualizado o diagrama de entidade e relacionamento físico, que diferente do diagrama conceitual, mostra a estrutura definitiva do banco de dados.

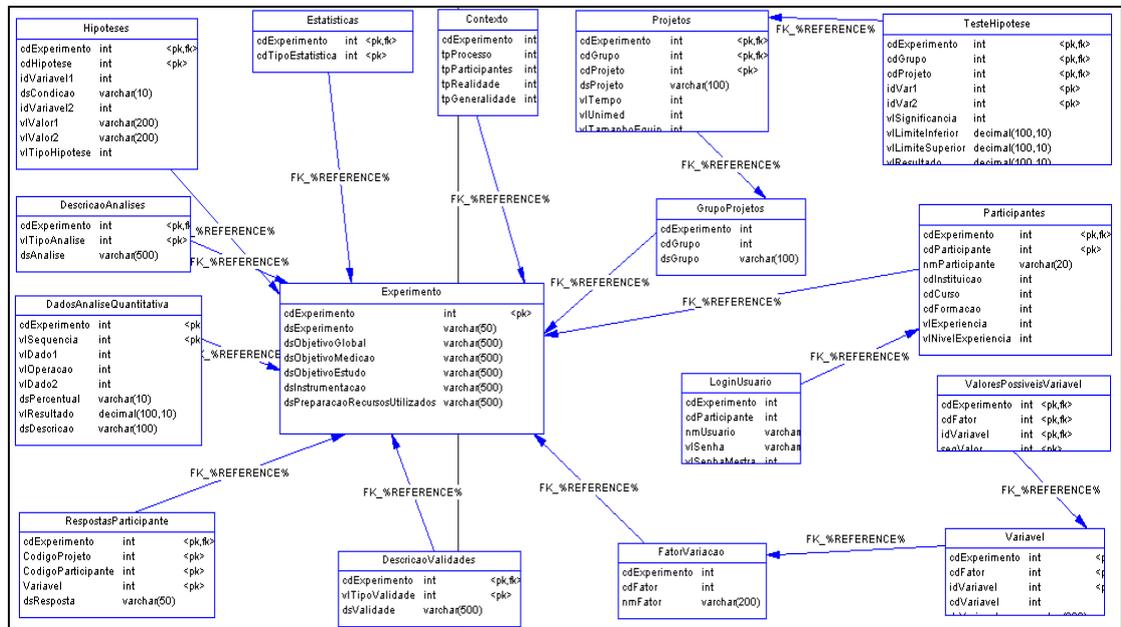


Figura 9 - Diagrama entidade relacionamento físico

A descrição das tabelas e de cada campo pode ser encontrada no dicionário de dados da seção 3.2.6.

### 3.2.6 Dicionário de dados

O dicionário de dados da ferramenta foi criado para oferecer uma maior documentação sobre a estrutura do banco de dados do sistema. Os tipos de dados de cada campo foram padronizados na seguinte forma:

- int: Integer, que caracteriza um valor numérico;
- varchar: lista de caracteres, onde seu tamanho é definido pelo valor entre parênteses;
- decimal: valores com casas decimais que recebe dois parâmetros, o primeiro que indica o tamanho do valor numérico e o segundo que indica o tamanho das casas decimais.

O dicionário de dados utiliza o seguinte padrão: nome da tabela, descrição do campo, nome do campo, tipo do campo, se faz parte da chave primária e se é obrigatório. Informações

como, se o campo faz parte da chave estrangeira, foram ocultados neste diagrama, pois podem ser visualizadas na figura 9. A tabela do dicionário de dados é apresentada no apêndice C deste trabalho.

### 3.3 IMPLEMENTAÇÃO

Nesta seção são apresentadas as ferramentas utilizadas e a operacionalidade da ferramenta.

#### 3.3.1 Técnicas e ferramentas utilizadas

Para um melhor desempenho do trabalho desenvolvido, foi utilizada a técnica de orientação a objetos (OO), que oferece uma melhor estruturação do código fonte para a implementação proposta. O ambiente de programação utilizado para o desenvolvimento foi a ferramenta Delphi 6.0, por possuir suporte a rápida criação de telas gráficas para interface com o usuário, e por disponibilizar um método de criação de gráficos e relatórios de forma rápida e precisa. O banco de dados utilizado foi o MY-Sql, por ser um banco de grande credibilidade no mercado e ser gratuito. A implementação foi realizada da seguinte maneira:

- a) para as classes e regras de negócio foram criadas duas *units* (unidades): a *unit* “uclasses.pas” e a *unit* “uregras.pas”;
- b) para os cálculos estatísticos, foi criada a *unit* “uregrasestatistica.pas”;
- c) para as funções de conversão de tipos de dados do formato Delphi para o formato MY-Sql a *unit* “ucontrole.pas” foi criada;
- d) a *unit* “uvariaveis.pas” possui todas as declarações globais de variáveis, constantes e *arrays*;
- e) para as ligações dos eventos do sistema, a *unit* “ueventos.pas” foi criada;
- f) para a conexão com a base de dados, a *unit* “ubase.pas” foi criada.

As demais *units* são as rotinas de telas do sistema.

Para ilustrar o código fonte, foram relacionadas partes importantes de algumas funcionalidades da ferramenta criada, que podem ser visualizadas a partir da seção 3.3.2.

3.3.2 Trecho de código do cálculo da moda da *unit* uregrasestatistica.pas

No quadro 23, é apresentado um trecho de código correspondente ao cálculo da moda.

```

function TCalculosEstatisticos.CalculaModa(Const coListaDados : TStringList): String;
Var
  liInd : Integer;
  ldDado : Double;
  liPosicao : Integer;
  liMaiorValor : Integer;
  lsModas : String;
  lbMudouValor : Boolean;
begin
  Try
    ldDado := 0;
    For liInd := 0 to coListaDados.Count - 1 Do
      Begin
        ldDado := StrtoFloat(coListaDados.Strings[liInd]);

        If Not loGuardaDado.Find(FloattoStr(ldDado), liPosicao) Then
          Begin
            loDado := TDado.Create;
            loDado.Dado := ldDado;
            loDado.Qtde := 1;
            loGuardaDado.AddObject(FloattoStr(loDado.Dado), loDado);
          End
        Else
          TDado(loGuardaDado.Objects[liPosicao]).Qtde :=
          TDado(loGuardaDado.Objects[liPosicao]).Qtde + 1;
        End;

        liMaiorValor := Low(Integer);
        lbMudouValor := False;
        lsModas := '';
        For liInd := 0 to loGuardaDado.Count - 1 Do
          Begin
            loDado := TDado(loGuardaDado.Objects[liInd]);

            If (liMaiorValor <> Low(Integer))
            ANd (loDado.Qtde <> liMaiorValor) Then
              lbMudouValor := True;

            If loDado.Qtde > liMaiorValor Then
              Begin
                liMaiorValor := loDado.Qtde;
                lsModas := FloattoStr(loDado.Dado);
              End
            Else
              If loDado.Qtde = liMaiorValor Then
                lsModas := lsModas + ' ' + FloattoStr(loDado.Dado);
            End;

            If Not lbMudouValor Then
              lsModas := '';

            Result := lsModas;
          Except
            On EConvertError Do
              loOcorrencias.AdicionaOcorrencia('Erro de conversão no valor ' + coListaDados.Strings[liInd]);
            On EAccessViolation Do
              loOcorrencias.AdicionaOcorrencia('Erro de conversão no valor ' + coListaDados.Strings[liInd]);
            Else
              loOcorrencias.AdicionaOcorrencia('Erro no valor ' + coListaDados.Strings[liInd]);
          End;
        end;
      end;
    end;
  end;
end;

```

Quadro 23 - Trecho de código do cálculo da moda

A moda indica os valores com maior incidência dentro de uma amostra, por tanto, o

primeiro laço serve para buscar todos os dados que estão repetindo, armazenando a quantidade de vezes que se repetem. O segundo laço serve para verificar as amostras que tem o maior número de aparições, no caso, se houver mais de uma moda, o sistema acumula as respectivas modas em uma string separando-as por ‘:’.

### 3.3.3 Trecho de código do cálculo da mediana da *unit* uregrasestatistica.pas

No quadro 24, é apresentado um trecho de código do cálculo da mediana.

```
function TCalculosEstatisticos.CalculaMediana(Const coListaDados : TStringList): String;
Var
  liInd : Integer;
  liPosicao : Integer;
  liAux : Integer;
  lbAchou : Boolean;
  ldDado : Double;
  ldCalc1,
  ldCalc2 : Double;
begin
  Try
    ldDado := 0;
    liind := coListaDados.Count;

    If liind Mod 2 <> 0 Then
      Begin
        liPosicao := Trunc((liInd + 1) / 2) - 1;

        If liPosicao <= coListaDados.Count - 1 Then
          ldDado := StrtoFloat(coListaDados.Strings[liPosicao]);
        End
      Else
        Begin
          liPosicao := Trunc(liInd / 2) - 1;

          If liPosicao <> -1 Then
            Begin
              ldCalc1 := StrtoFloat(coListaDados.Strings[liPosicao]);

              If liPosicao + 1 <= coListaDados.Count - 1 Then
                ldCalc2 := StrtoFloat(coListaDados.Strings[liPosicao + 1])
              Else
                ldCalc2 := ldCalc1;

              ldDado := (ldCalc1 + ldCalc2) / 2;
            End;
          End;
        End;

      Result := FloattoStr(ldDado);
    Except
      On EConvertError Do
        loOcorrencias.AdicionaOcorrencia('Erro de conversão no valor ' + coListaDados.Strings[liInd]);
      On EAccessViolation Do
        loOcorrencias.AdicionaOcorrencia('Erro de conversão no valor ' + coListaDados.Strings[liInd]);
      Else
        loOcorrencias.AdicionaOcorrencia('Erro no valor ' + coListaDados.Strings[liInd]);
    End;
  end;
```

Quadro 24 - Trecho de código do cálculo da mediana

A mediana é o ponto central de uma amostra. Se a amostra tiver um número ímpar de dados, a mediana é  $(n + 1) / 2$ , se tiver um número par, a mediana é a média entre os valores

$(n + 1) / 2$ , por exemplo, se o número de dados for 8, a mediana é a média dos valores 4 e 5, ou seja,  $(4 + 5) / 2 = 4,5$ .

O primeiro teste verifica se o número da amostra é ímpar, se for, o sistema já apresenta a mediana, se for par, o sistema faz o cálculo para fazer a média dos dois valores do ponto central.

### 3.3.4 Trecho de código do cálculo da variância da *unit* uregrasestatistica.pas

No quadro 25, é apresentado um trecho de código do cálculo da variância.

```
function TCalculosEstatisticos.CalculaVariancia(Const coListaDados : TStringList): String;
Var
  IdMedia,
  IdDado : Double;
  IdAcumulaValor : Double;
  liInd : Integer;
begin
  Try
    IdMedia := StrtoFloat(Self.CalculaMediaAritmetica(coListaDados));
  Except
    IdMedia := 0;
  End;

  IdAcumulaValor := 0;
  For liInd := 0 to coListaDados.Count - 1 Do
    Begin
      Try
        IdDado := StrtoFloat(coListaDados.Strings[liInd]);
      Except
        IdDado := 0;
      End;
      IdDado := Power((IdDado - IdMedia), 2); //Potência
      IdAcumulaValor := IdAcumulaValor + IdDado;
    End;

  If coListaDados.Count > 1 Then
    IdAcumulaValor := IdAcumulaValor / (coListaDados.Count - 1);

  IdAcumulaValor := ArredondaDouble(IdAcumulaValor, 2);
  Result := FloattoStr(IdAcumulaValor);
end;
```

Quadro 25 - Trecho de código do cálculo da variância

A variância é a quantidade de variação que os dados da amostra tem sobre a média, no trecho de código, pode-se visualizar que o primeiro cálculo é a média, chamada por uma sub rotina chamada calculamediaaritmetica, logo após, é feito um laço para percorrer todos os valores da amostra. Para cada valor, é feito o seguinte cálculo (Dado – Media) na potência 2, onde o resultado deste cálculo é acumulado numa variável local. Terminado o laço, o último cálculo á ser feito, é (AcumulaDado / Amostras), ou seja, os dados acumulados divididos pelo número da amostra. No caso do exemplo do código, se está dividindo o número da amostra menos 1, por que conforme a classe do Delphi chamada *TStringList*, os seus dados

acumulados começam do zero e não do um.

### 3.3.5 Operacionalidade da implementação

Nesta seção são apresentadas as funcionalidades da ferramenta desenvolvida. Para um melhor entendimento sobre o funcionamento dos passos da ferramenta, será apresentado um estudo experimental para verificar se o MODEST, que segundo SANTOS NETO, RESENDE E PADUA (2005, p. 5), “[...] é um método para automação dos testes de sistemas baseado nas especificações do SST.”, melhora a capacidade de uma equipe de programadores de software de capturar falhas nos seus programas. Este estudo foi criado comparando-se os resultados obtidos com o uso do MODEST dos resultados obtidos sem o seu uso. As funcionalidades mais importantes da ferramenta são mostradas neste capítulo.

#### 3.3.5.1 Tela principal do sistema

A tela principal do sistema pode ser visualizada na figura 10, onde estão as opções das funcionalidades do sistema.

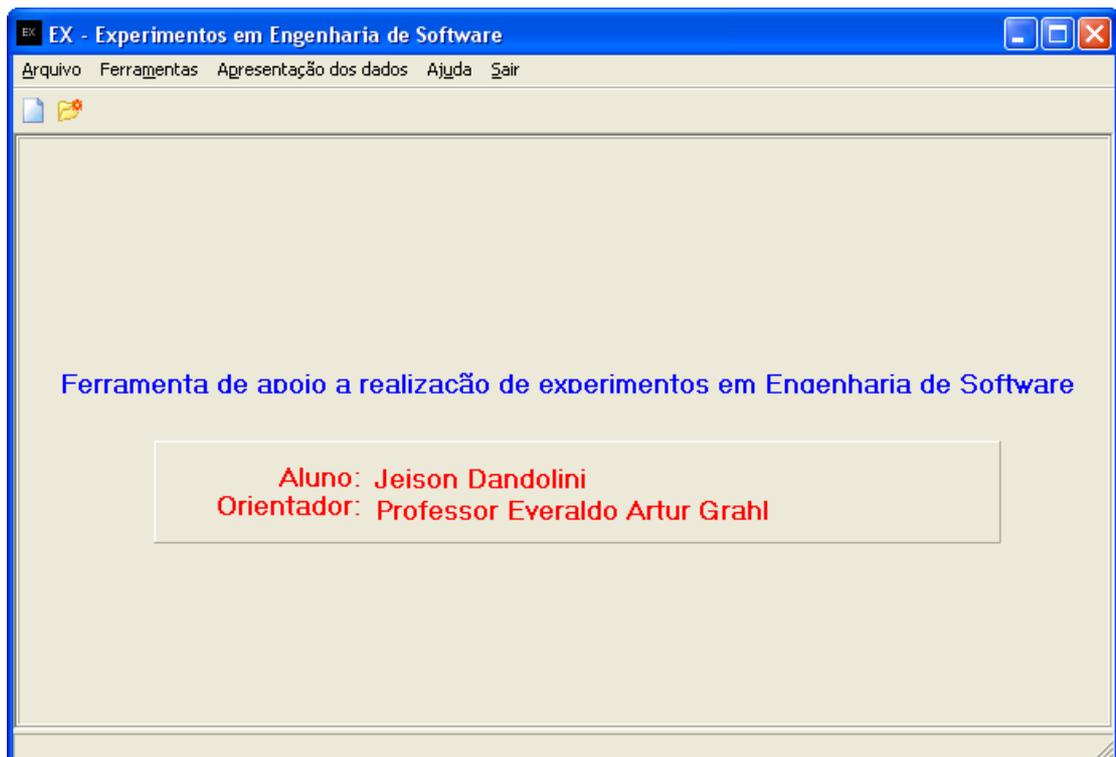


Figura 10 - Tela principal do sistema

No menu arquivo, estão as opções de criar e abrir um experimento, mesma funcionalidade oferecida pelo primeiro e segundo botão da barra de ferramentas.

No menu ferramentas está a opção de cadastrar uma senha para o participante, no caso de o mesmo for utilizar a ferramenta, e também a opção de agrupamento das variáveis, porém, esta última opção tem por requisito que um experimento esteja aberto na tela.

No menu Apresentação dos dados está localizada toda a parte de relatórios e gráficos do sistema, como por exemplo, o relatório de dados gerais, verificação das hipóteses, estatística descritiva, análise quantitativa e análise qualitativa e os seus respectivos gráficos, também existe o requisito de um experimento estar aberto na tela.

No menu ajuda, existem alguns tópicos de ajuda e a informações gerais sobre a autoria da ferramenta.

### 3.3.5.2 Tela de *Login*

A figura 11 ilustra a tela de *login*, que é apresentada no sistema sempre que o usuário ou participante for abrir ou criar um novo experimento, no caso de um novo experimento, só o usuário que possuir a senha mestre pode realizar esta operação.



Figura 11 - Tela de login

### 3.3.5.3 Tela de criação de um novo experimento

Na tela da figura 12, o usuário informa o nome do experimento que deseja criar.

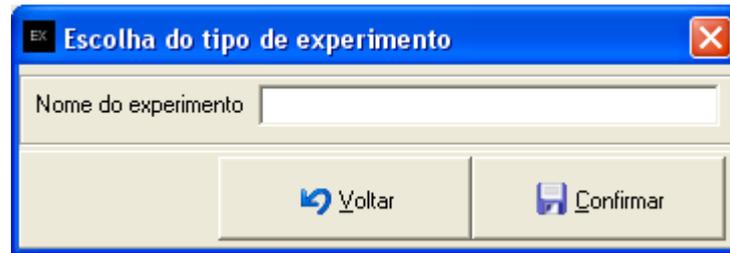


Figura 12 - Tela de criação de um novo experimento

#### 3.3.5.4 Tela principal do experimento

A tela da figura 13 oferece uma seqüência de passos a serem seguidos para a execução do experimento. Um melhor esclarecimento sobre o que envolve as fases de definição, planejamento, operação, análise e interpretação e apresentação dos dados pode ser visto na seção 2.2.6 nas fases do experimento.

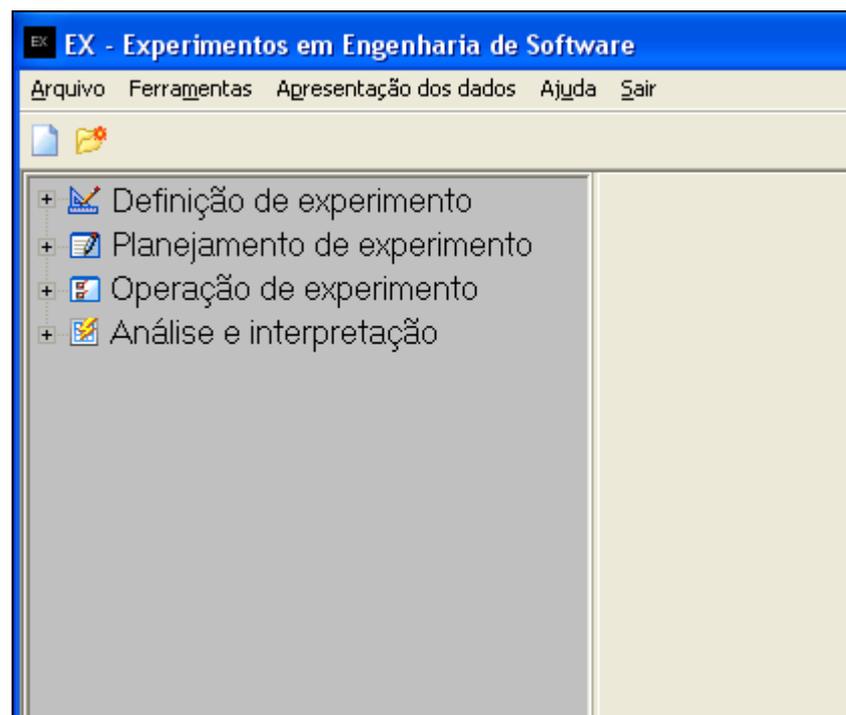


Figura 13 - Tela principal do experimento

#### 3.3.5.5 Tela de definição dos objetivos

Conforme pode ser observado na visualização da tela de definição dos objetivos da figura 14, o objetivo deste experimento se dá ao fato da verificação da utilidade de um método

de verificação de falhas chamado MODEST. As métricas do experimento são focadas no esforço despendido e a capacidade de detecção de falhas com e sem a presença do MODEST.

O objetivo do estudo está parametrizado conforme a definição dos objetivos do GQM relatado na seção 2.2.4, que é indicado por uma simples descrição, sem uma posterior verificação dos dados obtidos no experimento.

The screenshot shows a window with three vertically stacked text areas for defining objectives:

- Objetivo global:** O propósito desse estudo experimento é avaliar se o uso do MODEST no desenvolvimento de SI pode aumentar a capacidade de detecção de falhas dos testes gerados, e ao mesmo tempo reduzir os custos do desenvolvimento, em função da diminuição do esforço necessário na execução das atividades de teste.
- Objetivo da medição:** O foco da medição é o esforço(tempo) e a capacidade de detecção de falhas.
- Objetivo do estudo:**

<b>Analisar</b>	<utilização de um processo de software instanciado com ou sem o MODEST>
Com o propósito de	<avaliar>
Com respeito à	<esforço e capacidade de detecção de falhas>
Do ponto de vista do	<pesquisador>
No contexto dos	<alunos de graduação em Ciências da computação>

Figura 14 - Tela de definição dos objetivos

### 3.3.5.6 Tela de definição das questões e métricas

A tela da figura 15 apresenta a definição das questões para obtenção das métricas, e a definição das hipóteses, e está dividida em: fatores de variação, perguntas sobre os fatores, valores possíveis das variáveis e definição das hipóteses.

O fator de variação foi criado para agrupar variáveis que tenham alguma relação entre si, no caso, foram criados três fatores de variação: tempo de desenho, tempo de desenvolvimento e capacidade de detecção de falhas.

O tempo de desenho significa o tempo utilizado para desenhar a idéia da rotina que será desenvolvida, então a partir do fator de variação, foram criadas duas perguntas: “Qual o tempo de desenho utilizando o MODEST” e “Qual o tempo de desenho sem o MODEST”, o mesmo princípio foi utilizado para os outros dois fatores de variação. Estas perguntas criadas,

são posteriormente encaminhadas para os participantes.

Os valores possíveis das variáveis, possibilita ao experimentador, a opção de criar opções de resposta para o participante.

Por fim, foi feita a definição das hipóteses do experimento, que compõe as hipóteses nula e alternativa. A hipótese nula indica que a utilização do método MODEST não obteve resultados significativos e a hipótese alternativa indica que o método trouxe melhoria ao processo de desenvolvimento de software.

Fatores de variação

Nome do fator: Capacidade de detecção de falhas + Adicionar    X Excluir

Codigo	Nome do fator
1	Tempo de desenho
2	Tempo de desenvolvimento
3	Capacidade de detecção de falhas

Perguntas sobre os fatores

Descrição da pergunta: lhas(Número de falhas encontradas) sem o MODEST? + Adicionar    X Excluir

IdVariavel	CodigoFator	Descrição da pergunta
1	1	Qual o tempo de desenho utilizando o MODEST?
2	1	Qual o tempo de desenho sem o MODEST?

Valores possíveis das variáveis

Valor:  + Adicionar    X Excluir

IdVariavel	Sequencia	Valor da variável

Definição das hipóteses

Variável Valor 1: 5    Condição: <>    + Adicionar    X Excluir

Variável Valor 2: 6    Tipo hipótese: Alternativa

Código	Id variável 1	Condição	Id variável 2	Valor 1	Valor 2	Tipo Hipótese
1	1	=	2			Nula
2	1	<>	2			Alternativa
3	3	=	4			Nula
4	3	<>	4			Alternativa
5	5	=	6			Nula

Figura 15 - Tela de definição de questões e métricas

### 3.3.5.7 Tela de descrição da instrumentação

Na tela da figura 16, todo o método de instrumentação do experimento é descrito.

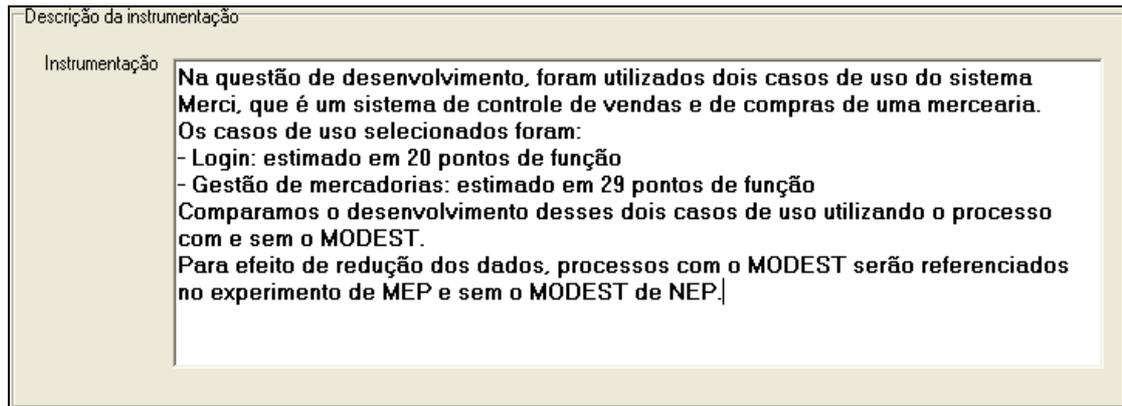


Figura 16 - Tela de descrição da instrumentação

### 3.3.5.8 Tela de seleção do contexto

A tela da figura 17, apresenta o contexto do experimento, que está no processo *On-line* por ser conduzido dentro de um ambiente controlado, para os alunos da graduação em Ciências da Computação, sobre uma realidade de um problema real e sobre um método específico de desenvolvimento.

As opções de processo são: *On-line* e *off-line*, as de participantes são: alunos e profissionais, as de realidade são: de um problema real ou de um problema modelado e as opções de generalidade são: específico e geral. Estas opções foram criadas objetivando experimentos para melhoria de processo.

Seleção do contexto

Processo: On-line

Participantes: Alunos

Realidade: Um problema real

Generalidade: Específico

Figura 17 - Tela de seleção do contexto

### 3.3.5.9 Tela de cadastro de grupo de projetos

O cadastro da figura 18 serve para agrupar projetos. No caso, foi cadastrado o grupo MODEST, para agrupar todos os casos de uso em um único grupo de projetos. É importante

ressaltar que projetos não possuem nenhuma ligação com experimentos, pois são utilizados dentro de um experimento para separação dos dados.

Código	Nome do grupo
1	Grupo MODEST

Figura 18 - Tela de cadastro de grupo de projetos

### 3.3.5.10 Tela de cadastro de projetos

O cadastro de projetos da figura 19, foi desenvolvido tendo uma visão mais genérica para melhoria de processo, por tal motivo, além de projetos, atende também a necessidade de cadastrar casos de uso ao invés de projetos. Neste caso, foram cadastrados os casos de uso *login* e gestão de mercadorias. O campo código é um número seqüencial que é gerado pelo sistema.

Código	Nome do projeto	Grupo	Tempo previsto	Unidade	Tamanho da equipe
1	Login	1	20	Pontos de função	12
2	Gestão de mercadorias	1	29	Pontos de função	12

Figura 19 - Tela de cadastro de projetos

### 3.3.5.11 Tela de cadastro de participantes do experimento

O cadastro de participantes da figura 20, serve para armazenar informações importantes de um participante como: formação, tempo de experiência de desenvolvimento e, se o participante estiver estudando, o curso que está matriculado. Estas informações auxiliam o experimentador, no que se diz respeito à qualidade do experimento.

Participantes

Código

Nome  Instituição

Formação  Curso

Experiência(Anos)  Experiência(Nível)

Lista de participantes cadastrados

Codigo	Nome do participante	Instituição	Curso	Formação	Experiência	Experiência(Nível)
1	Jeison	Pública	Informática	Técnico	Menos de 6 Meses	Baixo
2	Carlos	Pública	Informática	Universitária	De 6 Meses à 2 Anos	Baixo
3	Roberto	Pública	Informática	Pós Graduação	De 6 Meses à 2 Anos	Baixo
4	Fábio	Pública	Informática	Universitária	De 2 Anos à 4 Anos	Baixo
5	Ricardo	Pública	Informática	Universitária	De 2 Anos à 4 Anos	Baixo
6	Jefferson	Pública	Informática	Universitária	De 2 Anos à 4 Anos	Baixo
7	Juliano	Pública	Informática	Técnico	De 2 Anos à 4 Anos	Baixo
8	Fabiano	Pública	Informática	Técnico	Menos de 6 Meses	Baixo
9	Renata	Particular	Informática	Técnico	De 6 Meses à 2 Anos	Baixo
10	Catia	Particular	Informática	Técnico	De 2 Anos à 4 Anos	Baixo
11	César	Particular	Informática	Pós Graduação	Menos de 6 Meses	Baixo
12	Alexandre	Particular	Informática	Pós Graduação	De 2 Anos à 4 Anos	Baixo
13	Ronaldo	Pública	Informática	Técnico	Menos de 6 Meses	Baixo

Figura 20 - Tela de cadastro de participantes do experimento

A tela da figura 21, apresenta o gráfico do perfil dos participantes, onde o experimentador pode ter uma melhor visualização dos dados de cada participante.

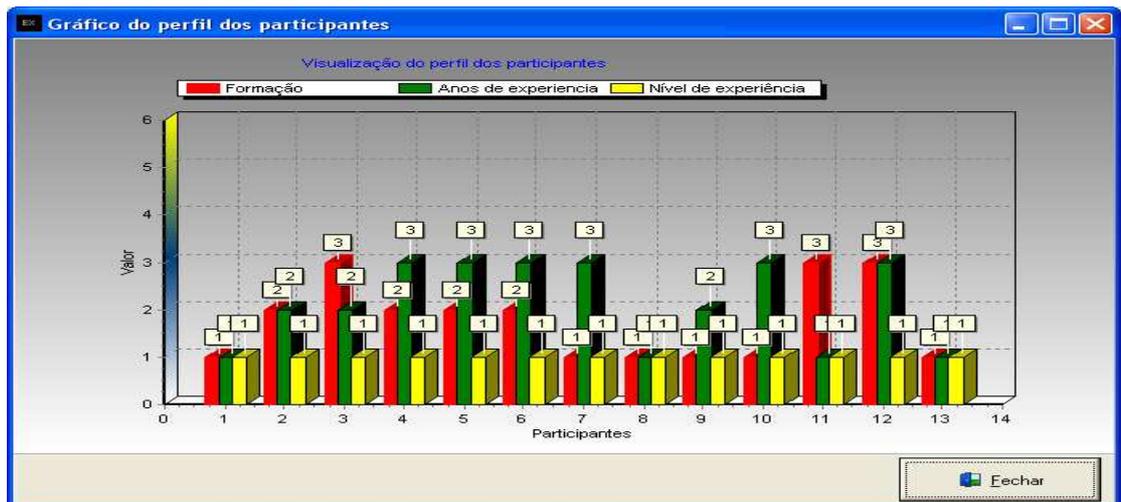


Figura 21 - Gráfico do perfil dos participantes

### 3.3.5.12 Tela de preparação e recursos utilizados

No cadastro da figura 22, são descritos todos os recursos utilizados na preparação do experimento. No caso, o perfil dos participantes ajudou a perceber que alguns participantes não tinham preparação adequada para participar do experimento, exigindo assim, algum treinamento de geração de testes, aprimorando o conhecimento dos participantes, e assim contribuindo para um melhor sucesso dos resultados do experimento.

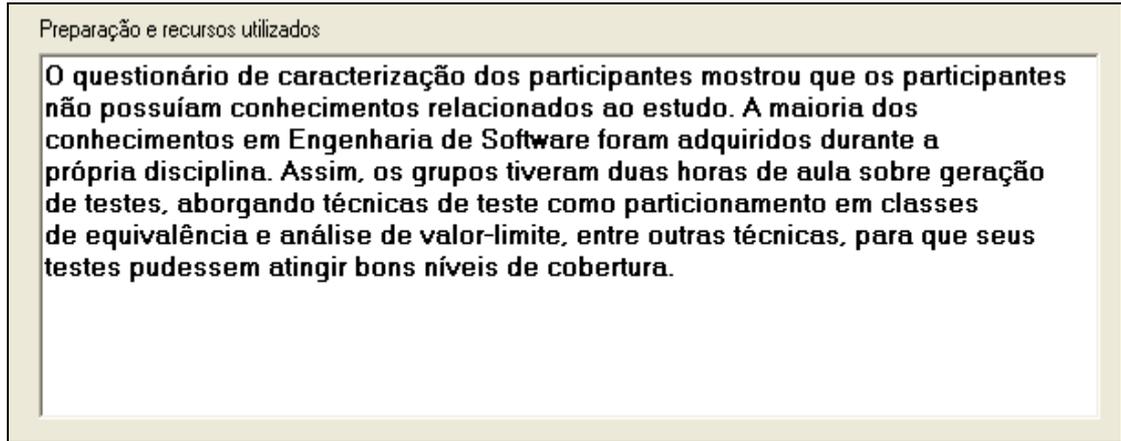


Figura 22 - Tela de preparação e recursos utilizados

### 3.3.5.13 Tela do questionário

Na tela da figura 23, as perguntas são respondidas pelo experimentador ou pelo participante do experimento. No caso do participante estar logado no sistema, somente o seu nome aparecerá no cadastro. As respostas das respectivas perguntas servem para a obtenção dos dados do experimento. O número de participantes indica o número de amostras estatisticamente falando.

Nome	Partic.	Nome
Login	1	Jeison
Gestão de mercador	2	Carlos
	3	Roberto
	4	Fábio
	5	Ricardo
	6	Jefferson
	7	Juliano
	8	Fabiano
	9	Renata
	10	Cátia
	11	Cézar
	12	Alexandre

Respostas		
Variável	Descrição da pergunta	
<input type="text"/>		
<input type="button" value="Criar perguntas"/> <input type="button" value="Editar pergunta"/> <input type="button" value="Atualizar dados"/> <input type="button" value="Visualizar"/>		
Resposta		
<input type="text"/>		
V..	Descrição	Resposta
1	Qual o tempo de desenho utilizando o MODEST?	
2	Qual o tempo de desenho sem o MODEST?	
3	Qual o tempo de desenvolvimento (desenho + Testes) utilizando o MODEST?	
4	Qual o tempo de desenvolvimento (desenho + Testes) sem o MODEST?	
5	Qual a capacidade de detecção de falhas(Número de falhas encontradas) utilizando o MODEST?	
6	Qual a capacidade de detecção de falhas(Número de falhas encontradas) sem o MODEST?	

Figura 23 - Tela do questionário

O usuário (experimentador) poderá obter um questionário para impressão clicando no botão visualizar, para entregar aos participantes. O referido questionário pode ser visualizado na figura 24.

EX - Experimentos em Engenharia de Software	
<b>Questionário para obtenção de métricas</b>	
Nome do participante :	Data
Descrição da pergunta	
Qual o tempo de desenho utilizando o MODEST?	
<b>Resposta :</b>	
Qual o tempo de desenho sem o MODEST?	
<b>Resposta :</b>	
Qual o tempo de desenvolvimento (desenho + Testes) utilizando o MODEST?	
<b>Resposta :</b>	
Qual o tempo de desenvolvimento (desenho + Testes) sem o MODEST?	
<b>Resposta :</b>	
Qual a capacidade de detecção de falhas(Número de falhas encontradas) utilizando o MODEST?	
<b>Resposta :</b>	
Qual a capacidade de detecção de falhas(Número de falhas encontradas) sem o MODEST?	
<b>Resposta :</b>	

Figura 24 - Impressão do questionário

#### 3.3.5.14 Tela da estatística descritiva

Na tela da estatística descritiva da figura 25, é realizada a análise dos dados apresentados, servindo de fundamentação para a análise quantitativa a qualitativa. Pode-se tomar como exemplo o desvio padrão da variável 1, que indica o valor 3,14, como o nível de variação dos dados da amostra em relação a média, ou seja, pode-se verificar que alguns dados variaram acima da média e outros dados variaram abaixo da média. O bom aproveitamento desta funcionalidade do sistema vai depender da experiência do experimentador de analisar dados estatísticos.

Tabela de estatísticas						
Estat. Variáveis	1	2	3	4	5	6
Média Aritmética	9,76	6,54	9,76	22,37	29	26,17
Moda		5,9:7,8		25,3		28:29
Mediana	9,7	6,1	9,7	22,8	29	28
Desvio Padrão	3,14	2,07	3,14	5,22	0	2,86

Figura 25 - Tela da estatística descritiva

### 3.3.5.15 Tela do teste de hipótese

A tela da figura 26, é muito importante para o experimento, pois verifica se as hipóteses formuladas podem ser aceitas ou não. O teste é feito por projeto, ou como no exemplo, pelo caso de uso *Login*. O teste do exemplo foi feito utilizando as variáveis 1 e 2, ou seja, tempo de desenho utilizando o MODEST e tempo de desenho sem utilizá-lo. O número de amostras é o número de respostas para cada variável, e o grau de liberdade é um cálculo estatístico obtido pela soma das duas amostras menos 2. O grau de significância delimita o limite inferior e o limite superior do teste. Estes limites servem para indicar se a hipótese foi rejeitada ou não, pois se o resultado do teste de hipótese ficar entre os limites inferior e superior, indica que a hipótese nula foi aceita, caso contrário, a hipótese nula foi rejeitada.

Resultado							
Var 1	Var 2	Projeto	Significância	Limite inferior	Limite superior	Resultado	Descrição do resultado
1	2	1	5	-2,069	+2,069	-7,38	A hipótese nula 1 = 2 foi rejeitada !

Figura 26 - Tela do teste de hipótese

### 3.3.5.16 Tela da análise quantitativa e qualitativa

Para apoiar a análise quantitativa e qualitativa, além da estatística descritiva, foi criada a tela de gráfico de variáveis, para que o experimentador possa visualizar através de um gráfico, os dados do experimento e tirar as suas conclusões. A referida tela pode ser visualizada na figura 27.

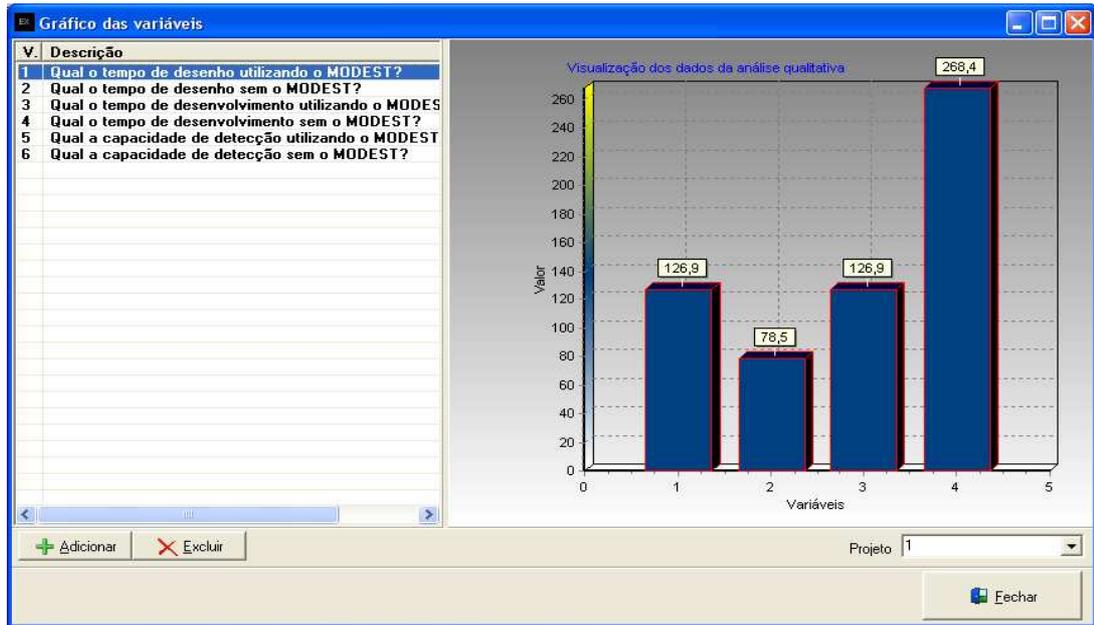


Figura 27 - Gráfico das variáveis

O gráfico da figura 28, apresenta os valores das variáveis 5 e 6, que são respectivamente, a capacidade de detecção de falhas utilizando o MODEST e a capacidade de detecção de falhas sem utilizá-lo..

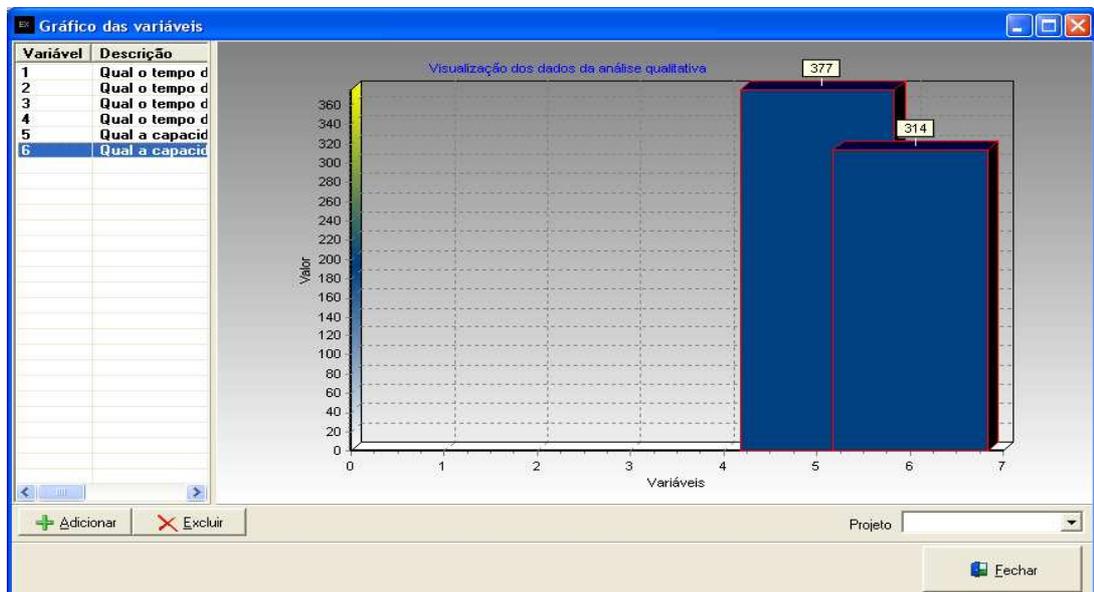


Figura 28 - Gráfico das variáveis

Pode-se verificar que na tela da figura 29, além da descrição da análise quantitativa, a ferramenta disponibiliza mais funcionalidades, como por exemplo, a opção do gráfico das variáveis e a inserção de cálculos de porcentagem para uma visão de percentuais dos dados. Pode-se verificar que 38,46% das amostras coletadas do tempo de desenho utilizando o MODEST ficaram abaixo da média 9,76 e 61,54% das amostras ficaram acima da média. Esta funcionalidade da ferramenta é descritiva e pode ser visualizada em forma de relatório.

Pode-se verificar na figura 29, que o tempo de desenho utilizando o MODEST foi maior do que o tempo sem utilizá-lo, porém o tempo de desenvolvimento utilizando o MODEST foi bem menor do que o tempo sem utilizá-lo. Na figura 30, pode-se verificar que a capacidade de detecção de falhas utilizando o MODEST foi maior do que sem utilizá-lo. Estas análises podem ser descritas no campo descrição da análise quantitativa da figura 30.

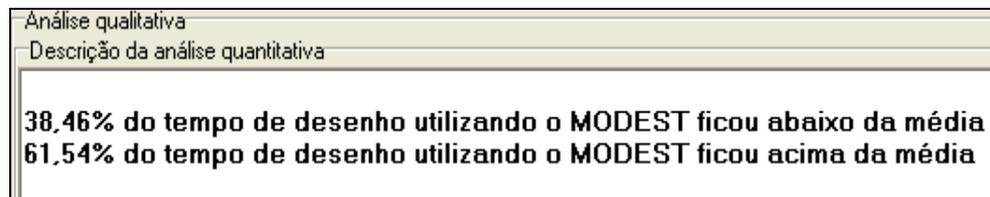


Figura 29 - Tela da análise qualitativa

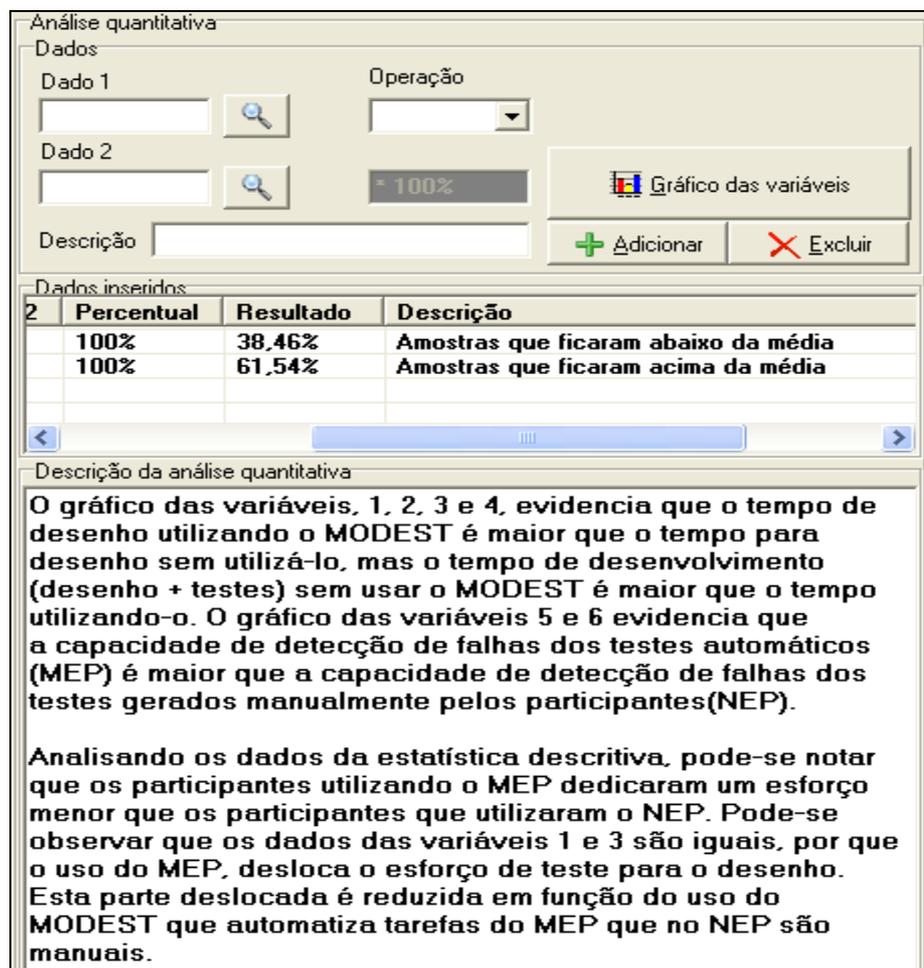


Figura 30 - Tela da análise quantitativa

### 3.3.5.17 Tela da descrição das validades

A funcionalidade do sistema que pode ser visualizada na figura 31, é a parte mais descritiva, onde o experimentador descreve as validades do experimento, ou seja, a justificativa para que o resultado do experimento seja considerado válido. A validade interna, como já visto neste trabalho, deve ser mais focada para os dados internos do experimento, ou seja, a garantia de que não haja fatores que possam distorcer os dados do experimento. A validade externa está na realidade externa ao experimento, no exemplo, o experimentador se preocupou em destacar que os participantes tinham pouca experiência em Engenharia de Software apesar de estarem cursando tal disciplina. A validade de conclusão relata uma descrição geral do experimento, dos dados, e possíveis indicações para futuros experimentos. A validade de construção verifica se os métodos utilizados no experimento foram válidos, no exemplo, o experimentador destaca que as medidas são bastante conhecidas no contexto de Engenharia de Software e por isso foram de bastante usabilidade no experimento.

<p>Validade de interna</p> <p><b>Para se evitar o problema da maturação, utilizou-se dois casos de uso com características bem distintas. Com isso, procurou-se evitar uma influência positiva na utilização dos tratamentos, visto que eles tinham um pouco em comum. Acredita-se que isso também ajudou a evitar influências negativas, como o cansaço e tédio, na repetição de uma atividade muito similar.</b></p> <p><b>As falhas foram selecionadas com a ajuda de um testador experiente na área, para garantir que as falhas fossem as mais reais possíveis.</b></p>
<p>Validade de externa</p> <p><b>O experimento envolveu alunos de Engenharia de Software do último ano, e como a maioria deles já trabalhavam na área, porém somente com desenvolvimento, acredita-se que eles tem o perfil de um desenvolvedor jr. com pouca experiência no mercado.</b></p> <p><b>O exemplo utilizado, dois casos de uso de um sistema para controle de mercearias, embora seja pequeno, possui características da realidade do mundo real, e este experimento pode servir de base para outros estudos nesta área.</b></p>
<p>Validade de conclusão</p> <p><b>A verificação da hipótese foi feita utilizando o teste t. A potência deste teste é considerada alta. No intuito de se minimizar os efeitos associados a imprecisão das medidas, relacionadas ao esforço empregado nos passos do experimento, foi solicitado o envio diário dos resultados intermediários e do formulário com o registro de horas relativos a esses resultados. Esses formulários eram analisados para evitar uma imprecisão dos dados coletados. Além disso, foi acompanhado cada passo do estudo, em função desse acompanhamento descartou-se participantes cuja qualidade dos registros não foi considerada adequada para o experimento.</b></p>
<p>Validade de construção</p> <p><b>As idéias utilizadas no estudo já tinham sido utilizadas em alguns estudos preliminares anteriores, sendo aperfeiçoadas para o estudo atual. Assim acredita-se que elas foram suficientemente definidas antes de serem traduzidas em tratamentos e medidas. Essas medidas são bastante claras e amplamente conhecidas no contexto de Engenharia de Software e, além disso, desde os primeiros estudos se mostraram adequadas. Planejou-se o estudo de forma que os participantes não tivessem conhecimento das hipóteses a serem verificadas, de forma a evitar comportamentos positivos ou negativos, pois sabia a existência de participantes defensores dos processos ágeis.</b></p>

Figura 31 - Tela da descrição das validades

### 3.3.5.18 Apresentação dos dados

A apresentação dos dados é composta por relatórios e gráficos, que espelham os dados obtidos no experimento, e pode ser encontrada no apêndice B deste trabalho.

## 3.4 RESULTADOS E DISCUSSÃO

Para que os resultados da implementação da ferramenta ficassem mais evidentes, a utilização de um caso real, onde o experimentador verifica a melhoria da capacidade de detecção de falhas utilizando um método de testes chamado MODEST, foi necessária. Esta utilização contribuiu para a avaliação do resultado final gerado pela ferramenta, que provou suportar grande parte dos passos necessários para a realização do experimento.

Os resultados mais importantes obtidos na execução deste trabalho foram:

- a) suporte á definição de questões e métricas;
- b) visualização do perfil dos participantes: pois indica ao experimentador o nível de qualidade dos dados que se poderá obter durante o experimento;
- c) a geração do questionário para as respostas dos participantes;
- d) a estatística descritiva dos dados;
- e) o teste de hipótese: que serve para a verificação das hipóteses formuladas;
- f) as análises dos dados obtidos;
- g) as validades;
- h) a geração de gráficos para uma melhor visualização dos dados do experimento.

O quadro 26 possui uma comparação da ferramenta desenvolvida com os trabalhos correlatos.

Trabalho	Definição dos objetivos	Definição de questões	Cálculos estatísticos	Geração de gráficos
EX	Possui	Possui	Possui	Possui
GROSS (2001)	Possui	Possui	Não possui	Não possui
MedPlan	Possui	Possui	Não possui	Não possui
Metrics	Não possui	Não possui	Não possui	Possui

Quadro 26 - Relação entre trabalhos correlatos

O método para avaliar os resultados obtidos, foi o de comparar os resultados gerados pela ferramenta desenvolvida neste trabalho com os dados do experimento realizado para a verificação da utilidade do método MODEST na detecção de falhas.

Pode-se verificar que a ferramenta desenvolvida neste trabalho atende todas as colunas da tabela, enquanto o protótipo desenvolvido em GROSS (2001) e a ferramenta *MedPlan*, atenderam somente as colunas 1 e 2, que dizem respeito a definição dos objetivos e a definição das questões. A ferramenta *Metrics* atendeu somente a última coluna.

## 4 CONCLUSÕES

Experimentos em Engenharia de Software são importantes para a verificação de teorias formuladas. Neste aspecto, observou-se que a ferramenta desenvolvida oferece suporte á formulação de questões e a obtenção das mesmas, a definição de hipóteses e a sua verificação através do teste de hipótese.

Os conceitos de Engenharia de Software Experimental foram aplicados observando-se todas as fases necessárias para a realização de um experimento. Inicialmente, tinha-se o objetivo de se desenvolver a ferramenta direcionada para melhoria de processo e de produto, mais conforme o andamento, optou-se por focar apenas para a melhoria de processo. Utilizou-se parcialmente a técnica GQM, para a realização e obtenção de medidas do experimento. Foram adotadas técnicas de estatística para a análise dos dados do experimento, desde a definição das hipóteses nula e alternativa, até a verificação das mesmas no teste de hipótese.

Alguns dados fixos criados na ferramenta, foram baseados em artigos de Experimentos em Engenharia de Software direcionados a melhoria de processo. Para uma versão melhorada estes campos deveriam ser flexíveis.

As tecnologias utilizadas para a construção da ferramenta foram de grande utilidade, principalmente no que se diz respeito á elaboração de gráficos, formulação de telas gráficas e criação do banco de dados.

Esta ferramenta pode ser utilizada em empresas que utilizam métodos de Engenharia de Software e também, por alunos de Engenharia de Software para fins acadêmicos.

### 4.1 EXTENSÕES

A sugestão para trabalhos futuros seria a criação de funcionalidades mais diversificadas sobre cálculos estatísticos para a manipulação e análise dos dados do experimento e um maior aprofundamento sobre técnicas GQM.

## REFERÊNCIAS BIBLIOGRÁFICAS

BERENSON, M. L.; LEVINE, D. M.; STEPHAN, D. **Estatística: teoria e aplicações usando microsoft excel em português**. Rio de Janeiro: LTC, 2000.

GLADCHEFT, A. P.; SANCHES, R.; SILVA, D. M. **Um instrumento de avaliação de qualidade de software educacional: como elaborá-lo**. [S.l.], [2001]. Disponível em: <<http://www.ime.usp.br/dcc/posgrad/teses/anapaula/artigoWQS.PDF>>. Acesso em: 30 set. 2005.

GROSS, J. C. **Protótipo de um software de apoio à utilização do GQM (Goal-Question-Metric)**. 2001. 60 f. Trabalho de Conclusão de Curso (Bacharelado em Ciências da Computação) – Centro de Ciências Exatas e Naturais, Universidade Regional de Blumenau, Blumenau.

KASBURG, A. **Avaliação da qualidade de software de gestão integrada utilizando as normas ISO/IEC 14598-1**. 2001. 76 f. Trabalho de Conclusão de Curso (Bacharelado em Ciências da Computação) – Centro de Ciências Exatas e Naturais, Universidade Regional de Blumenau, Blumenau.

PFLEEGER, S. L. **Engenharia de software: teoria e prática**. São Paulo: Pearson, 2004.

SANTOS NETO, P.; RESENDE, R.; PADUA, C. **Requisitos para automação dos testes de sistemas de informação**. [S.l.], [2005]. Disponível em: <<http://homepages.dcc.ufmg.br/~pasn/Arquivos/sbsi2005.pdf>>. Acesso em: 15 fev. 2007.

SCHNAIDER, L. et al. **Uma abordagem para medição e análise em projetos de desenvolvimento de software**. In: SIMPÓSIO BRASILEIRO DE QUALIDADE DE SOFTWARE, 3., 2004, Brasília. **Anais...** Brasília: UCB-SBC, 2004. Não paginado. Disponível em: <<http://www.sbc.org.br/bibliotecadigital/download.php?paper=259>>. Acesso em: 22 out. 2006.

SOMMERVILLE, I. **Engenharia de software**. São Paulo: Addison Wesley, 2003.

TRAVASSOS, G. H. **Introdução à engenharia de software experimental**. [S.l.], [2002]. Disponível em: <<http://cronos.cos.ufrj.br/publicacoes/reltec/es59002.pdf>>. Acesso em: 18 set. 2005.

## APÊNDICE A – Descrição dos casos de uso

UC1 – Loga no sistema
<b><i>public UseCase:</i></b> Permite o acesso ao sistema
<b><i>Constraints</i></b>
▪ <i>Approved Pré-condição</i> . Deve ter a senha cadastrada.
▪ <i>Approved Pós-condição</i> . Login efetuado com sucesso.
<b><i>Cenários</i></b>
<u>Loga no sistema</u> {Principal}.
1 - O usuário informa o seu usuário e senha e indica se é usuário mestre ou não 2 – O usuário clica em Confirmar 3 - O sistema valida as informações e loga o usuário
<u>Os dados estão incorretos</u> {Exceção}.
No passo 3, se os dados estiverem incorretos apresentar a mensagem: "Usuário ou senha incorretos"

Quadro 27 - Loga no sistema

UC2 – Cria um novo experimento
<b><i>public UseCase:</i></b> Criação de um novo experimento
<b><i>Constraints</i></b>
▪ <i>Approved Pré-condição</i> . O usuário deve estar logado no sistema.
▪ <i>Approved Pós-condição</i> . Experimento criado com sucesso.
<b><i>Cenários</i></b>
<u>Cria um novo experimento</u> {Principal}.
1 - O sistema apresenta a tela de criação do experimento; 2 – O usuário informa os dados necessários para a criação do experimento 3 - O sistema valida as informações e cria o experimento 4 - O sistema apresenta a tela principal;

Quadro 28 - Cria um novo experimento

UC3 – Define os objetivos
<b><i>public UseCase:</i></b> Registra os objetivos do experimento
<b><i>Constraints</i></b>
<ul style="list-style-type: none"> <li>▪ <i>Approved Pré-condição</i> . O usuário deve estar logado no sistema.</li> <li>▪ <i>Approved Pós-condição</i> . Objetivos definidos com sucesso.</li> </ul>
<b><i>Cenários</i></b>
<u>Usuário define os objetivos</u> {Principal}.
<p>1 - O sistema apresenta a tela de registro de objetivos;</p> <p>2 - O usuário informa os dados da definição dos objetivos global, da medição e do estudo;</p> <p>3 - O sistema valida e grava as informações do objetivo</p>
<u>Usuário altera os dados do objetivo</u> {Alternativo}.
<p>No passo 3, o usuário opta por alterar os dados do objetivo</p> <p>2.1 - O sistema localiza e apresenta a tela para edição, com as informações do objetivo;</p> <p>2.2 - O usuário preenche as informações e confirma os dados</p> <p>2.3 - Retorna ao passo 3;</p>

Quadro 29 - Define os objetivos

UC4 – Registra as questões e métricas
<b><i>public UseCase:</i></b> Permite o cadastro das questões e métricas
<b><i>Constraints</i></b>
<ul style="list-style-type: none"> <li>▪ <i>Approved Pré-condição</i> . O usuário deve estar logado no sistema.</li> <li>▪ <i>Approved Pós-condição</i> . Questões e métricas registradas com sucesso.</li> </ul>
<b><i>Cenários</i></b>
<u>O usuário define as questões e métricas</u> {Principal}.
<p>1 - O sistema apresenta a tela para o cadastro das questões e métricas.</p> <p>2 - O usuário informa o nome do fator de variação e o sistema valida as informações</p> <p>3 - O usuário informa a pergunta(Variável) e o sistema valida as informações</p> <p>4 - O usuário informa os valores possíveis da pergunta e o sistema valida as informações</p> <p>5 - O usuário define as hipóteses e o sistema valida as informações</p>
<u>O usuário não informou as informações antes de adicionar uma hipótese</u> {Exceção}.
No passo 5, se o usuário não informar os valores, o sistema deve apresentar a seguinte mensagem "Você deve informar pelo menos um valor"
<u>O usuário não selecionou o fator de variação quando foi adicionar uma pergunta</u> {Exceção}.
No passo 2, se o usuário não selecionar um fator de variação, apresentar a mensagem "Você deve selecionar primeiro um fator"
<u>O usuário não selecionou uma pergunta quando foi indicas os valores possíveis</u> {Exceção}.
No passo 4, se o usuário não selecionar a pergunta(variável), o sistema deve apresentar a seguinte mensagem "Você deve selecionar uma variável"
<u>O usuário opta por excluir as informações</u> {Alternativo}.
<p>1 - O usuário selecionar a informação que deseja excluir e clica no botão excluir</p> <p>2 - O sistema exclui a informação e apresenta uma mensagem indicando que a informação foi excluída com sucesso.</p>

Quadro 30 - Registra as questões e métricas

UC5 – Realiza a descrição da instrumentação
<b><i>public UseCase:</i></b> Descreve a instrumentação
<b><i>Constraints</i></b>
<ul style="list-style-type: none"> <li>▪ <i>Approved Pré-condição</i> . O usuário deve estar logado no sistema.</li> <li>▪ <i>Approved Pós-condição</i> . Descrição da instrumentação gerada com sucesso.</li> </ul>
<b><i>Cenários</i></b>
<u>Usuário descreve a instrumentação {Principal}</u> .
<ol style="list-style-type: none"> <li>1 – O sistema apresenta a tela de descrição da instrumentação;</li> <li>2 - O usuário descreve a instrumentação</li> <li>3 - O usuário clica em gravar</li> <li>4 – O sistema valida e grava as informações</li> </ol>
<u>Usuário altera os dados da instrumentação {Alternativo}</u> .
<p>No passo 2, o usuário opta por alterar os dados</p> <ul style="list-style-type: none"> <li>- O usuário altera os dados e clica em gravar</li> <li>- O sistema valida e grava as informações</li> </ul>

Quadro 31 - Realiza a descrição da instrumentação

UC6 – Realiza a seleção de um contexto
<b><i>public UseCase:</i></b> Define o contexto do experimento
<b><i>Constraints</i></b>
<ul style="list-style-type: none"> <li>▪ <i>Approved Pré-condição</i> . O usuário estar logado no sistema.</li> <li>▪ <i>Approved Pós-condição</i> .O contexto do experimento foi definido.</li> </ul>
<b><i>Cenários</i></b>
<u>O usuário seleciona o contexto do experimento {Principal}</u> .
<ol style="list-style-type: none"> <li>1 - O sistema apresenta a tela de seleção do contexto;</li> <li>2 - O usuário seleciona as informações desejadas, como o tipo de processo do experimento, o tipo de participantes, a realidade e a generalidade do experimento.</li> <li>3 - O usuário clica em gravar</li> <li>4 - O sistema valida e grava as informações</li> </ol>
<u>O usuário altera as informações do contexto {Alternativo}</u> .
<p>No passo 2, o usuário opta por alterar as informações do contexto.</p> <ol style="list-style-type: none"> <li>1 - O usuário altera as informações do contexto</li> <li>2 - O usuário clica em gravar</li> <li>3 - O sistema valida e grava as informações</li> </ol>

Quadro 32 - Realiza a seleção de um contexto

UC7 – Cadastra grupo de projetos
<b><i>public UseCase:</i></b> Permite o cadastro de grupos de projetos
<b><i>Constraints</i></b>
<ul style="list-style-type: none"> <li>▪ <i>Approved Pré-condição</i> . O usuário deve estar logado no sistema.</li> <li>▪ <i>Approved Pós-condição</i> . Grupo de projeto cadastrado com sucesso.</li> </ul>
<b><i>Cenários</i></b>
<u>O usuário cadastra um grupo de projeto</u> {Principal}.
<ol style="list-style-type: none"> <li>1 - O usuário informa o nome do grupo e clica em adicionar</li> <li>2 - O sistema valida e grava as informações</li> </ol>
<u>Na hora da exclusão, o usuário não selecionou o grupo</u> {Exceção}.
No passo da exclusão, se o usuário não selecionar um grupo, apresentar a mensagem "Você deve selecionar um grupo para exclusão"
<u>O usuário edita as informações</u> {Alternativo}.
No passo 1, o usuário opta por editar as informações <ol style="list-style-type: none"> <li>1 - O usuário clica em editar e altera o nome do grupo</li> <li>2 - O sistema grava as informações</li> </ol>
<u>O usuário exclui um grupo</u> {Alternativo}.
NO passo 1, o usuário opta por excluir um grupo <ol style="list-style-type: none"> <li>1 - O usuário seleciona o grupo</li> <li>2 - O usuário clica em excluir</li> <li>3 - O sistema valida e exclui o grupo</li> </ol>

Quadro 33 - cadastra grupo de projetos

UC8 – Cadastra projeto
<b><i>public UseCase:</i></b> Permite o cadastro de projetos
<b><i>Constraints</i></b>
<ul style="list-style-type: none"> <li>▪ <i>Approved Pré-condição</i> . O usuário deve estar logado no sistema.</li> <li>▪ <i>Approved Pré-condição</i> . Pelo menos um grupo de projeto deve ter sido cadastrado.</li> <li>▪ <i>Approved Pós-condição</i> . Projeto cadastrado com sucesso.</li> </ul>
<b><i>Cenários</i></b>
<u>Usuário cadastra projeto</u> {Principal}.
<ol style="list-style-type: none"> <li>1 - O sistema apresenta a tela de cadastro de projeto;</li> <li>2 - O usuário informa o nome do projeto, grupo, tempo previsto, unidade de medida e tamanho da equipe e clica em Adicionar;</li> <li>3 - O sistema valida as informações e grava.</li> </ol>
<u>O usuário edita as informações</u> {Alternativo}.
No passo 2, o usuário opta por editar uma informação
<ol style="list-style-type: none"> <li>1 - O usuário seleciona o projeto</li> <li>2 - O usuário atualiza o nome do projeto, grupo, tempo previsto, unidade de medida e tamanho da equipe e clica em adicionar</li> <li>3 - O sistema valida e grava as informações</li> </ol>
<u>O usuário exclui um projeto</u> {Alternativo}.
No passo 2, o usuário opta por excluir um projeto
<ol style="list-style-type: none"> <li>1 - O usuário seleciona um projeto e clica em Excluir</li> <li>2 - O sistema valida as informações e grava</li> </ol>
<u>Usuário não selecionou um projeto</u> {Exceção}.
No passo de edição e exclusão de um projeto, se o usuário não selecionar um projeto o sistema apresenta a seguinte mensagem "Você deve selecionar um projeto"

Quadro 34 - cadastra projeto

UC9 – Cadastra os participantes
<b><i>public UseCase:</i></b> Permite o cadastro de participantes
<b><i>Constraints</i></b>
<ul style="list-style-type: none"> <li>▪ <i>Approved Pré-condição</i> . O usuário deve estar logado no sistema.</li> <li>▪ <i>Approved Pós-condição</i> . Participante cadastrado com sucesso.</li> </ul>
<b><i>Cenários</i></b>
<u>Usuário cadastra os participantes</u> {Principal}.
<ol style="list-style-type: none"> <li>1 - o sistema apresenta a tela de cadastro de participantes;</li> <li>2 - o usuário informa o nome do participante, a instituição, a formação, o curso, a experiência em anos e a experiência em nível e clica em Adicionar;</li> <li>3 - o sistema valida e grava as informações.</li> </ol>
<u>O usuário exclui um participante</u> {Alternativo}.
<p>No passo 2, o usuário opta por excluir um participante</p> <ol style="list-style-type: none"> <li>1 - o usuário seleciona um participante e clica em excluir;</li> <li>2 - o sistema valida as informações e exclui.</li> </ol>
<u>O usuário não selecionou um participante</u> {Exceção}.
Nos passos de edição e exclusão, se o usuário não informar um participante, o sistema apresenta a seguinte mensagem "Você deve selecionar um participante".
<u>Usuário edita participantes</u> {Alternativo}.
<p>No passo 2, o usuário opta por editar as informações dos participantes.</p> <ol style="list-style-type: none"> <li>1 - o usuário seleciona um participante e clica em editar;</li> <li>2 - o sistema valida as informações e apresenta na tela;</li> <li>3 - o usuário informa o nome do participante, o a instituição, a formação, o curso, a experiência em anos e a experiência em nível e clica em adicionar;</li> <li>4 - o sistema valida as informações e grava.</li> </ol>

Quadro 35 - Cadastra os participantes

UC10 – Descreve a preparação e recursos utilizados
<b><i>public UseCase:</i></b> Permite o registro da preparação e recursos utilizados
<b><i>Constraints</i></b>
▪ <i>Approved Pré-condição</i> . O usuário deve estar logado no sistema.
▪ <i>Approved Pós-condição</i> . Registro efetuado com sucesso.
<b><i>Cenários</i></b>
<u>Usuário descreve a preparação e recursos utilizados</u> {Principal}.
1 - o sistema apresenta a tela de preparação e recursos utilizados; 2 - o usuário informa a descrição da preparação e recursos utilizados e clica em gravar; 3 - o sistema grava as informações.
<u>O usuário altera os dados</u> {Alternativo}.
No passo 2, o usuário opta por alterar os dados da preparação e recursos utilizados; 1 - o usuário informa a descrição da preparação e recursos utilizados e clica em gravar; 2 - o sistema grava as informações.

Quadro 36 - Descreve a preparação e recursos utilizados

UC11 – Responde ao questionário
<b><i>public UseCase:</i></b> Permite responder o questionário
<b><i>Constraints</i></b>
<ul style="list-style-type: none"> <li>▪ <i>Approved Pré-condição</i> . Usuário ou participante devem estar logados no sistema.</li> <li>▪ <i>Approved Pré-condição</i> . As questões devem ter sido formuladas.</li> <li>▪ <i>Approved Pós-condição</i> . Questionário respondido com sucesso.</li> </ul>
<b><i>Cenários</i></b>
<u>Reposta do questionário</u> {Principal}.
<ol style="list-style-type: none"> <li>1 - o sistema apresenta a tela para respostas do questionário;</li> <li>2 - o usuário ou participante seleciona o projeto, e o participante;</li> <li>3 - o usuário ou participante clica em editar pergunta;</li> <li>4 - o usuário ou participante responde a pergunta e clica em atualizar dados;</li> <li>5 - o sistema valida e grava as informações.</li> </ol>
<u>O usuário ou participante não selecionou um participante</u> {Exceção}.
No passo das respostas, se o usuário ou participante não selecionar um participante, o sistema apresenta a mensagem "Você deve selecionar um participante".
<u>O usuário ou participante não selecionou uma pergunta</u> {Exceção}.
No passo das respostas das perguntas, se o usuário ou participante não selecionar a pergunta, o sistema deve apresentar a mensagem "Você deve selecionar uma pergunta".
<u>Se não existir perguntas.</u> {Alternativo}.
No passo da resposta da pergunta, se caso não existir pergunta
<ol style="list-style-type: none"> <li>1 - o usuário ou participante clica em Criar perguntas;</li> <li>2 - o sistema cria as perguntas e apresenta na tela;</li> <li>3 - volta ao passo 2.</li> </ol>
<u>Usuário ou participante não selecionou um projeto</u> {Exceção}.
No passo das respostas, se o usuário ou participante não selecionar um projeto no caso de melhoria de processo, o sistema deve apresentar a mensagem "Você deve selecionar um projeto".

Quadro 37 - Responde ao questionário

UC12 – Aplica a estatística descritiva
<b><i>public UseCase:</i></b> Permite realizar a estatística descritiva
<b><i>Constraints</i></b>
<ul style="list-style-type: none"> <li>▪ <i>Approved Pré-condição</i> . O usuário deve estar logado no sistema.</li> <li>▪ <i>Approved Pré-condição</i> . O questionário deve ter sido respondido.</li> <li>▪ <i>Approved Pós-condição</i> . Aplicou a estatística descritiva com sucesso.</li> </ul>
<b><i>Cenários</i></b>
<u>Usuário aplica a estatística descritiva</u> {Principal}.
<ol style="list-style-type: none"> <li>1 - o sistema apresenta a tela da estatística descritiva;</li> <li>2 - o usuário escolhe um tipo de estatística e clica no botão adicionar;</li> <li>3 - o sistema realiza o cálculo estatístico e mostra na tela.</li> </ol>
<u>O usuário atualiza a estatística</u> {Alternativo}.
<p>No passo 1, o usuário opta por atualizar os dados da estatística</p> <ol style="list-style-type: none"> <li>1 - usuário clica no botão atualizar;</li> <li>2 - o sistema atualiza os dados da estatística descritiva na tela.</li> </ol>
<u>O usuário não escolheu um tipo de estatística</u> {Exceção}.
No passo 2, se o usuário não escolher um tipo de estatística, o sistema apresenta a mensagem "Você deve escolher um tipo de estatística".
<u>Usuário escolheu uma estatística já existente</u> {Exceção}.
No passo 2, se o usuário escolher uma estatística já existente, o sistema apresenta a mensagem "Esta estatística já foi inserida".

Quadro 38 - Aplica a estatística descritiva

UC13 – Realiza o teste de hipótese
<b><i>public UseCase:</i></b> Permite realizar o teste de hipótese
<b><i>Constraints</i></b>
<ul style="list-style-type: none"> <li>▪ <i>Approved Pré-condição</i> . O usuário deve estar logado no sistema.</li> <li>▪ <i>Approved Pré-condição</i> . O questionário deve ter sido respondido.</li> <li>▪ <i>Approved Pós-condição</i> . Teste de hipótese realizado com sucesso.</li> </ul>
<b><i>Cenários</i></b>
<u>Usuário realiza o teste de hipótese</u> {Principal}.
<p>1 - o sistema apresenta a tela do teste de hipótese;</p> <p>2 - o usuário informa o número do projeto e clica em confirmar;</p> <p>3 - o usuário informa a variável 1, a variável 2, o grau de significância e clica no botão executar teste;</p> <p>4 - o sistema valida as informações e realiza o teste.</p>
<u>O usuário cancela a geração do teste</u> {Alternativo}.
<p>NO passo 3, o usuário opta por cancelar a geração do teste</p> <p>1 - o usuário clica no botão cancelar;</p> <p>2 - o sistema volta ao passo 1.</p>
<u>O usuário não informou os dados corretos</u> {Exceção}.
No passo 3, se o usuário não informar os dados corretos mostrar mensagem acusando o erro.
<u>O usuário não selecionou o teste</u> {Exceção}.
No momento da exclusão do teste, se o usuário não selecionar um teste o sistema apresenta a mensagem "Você deve selecionar um teste".
<u>Usuário exclui um teste</u> {Alternativo}.
<p>No passo 3, o usuário opta por excluir um teste</p> <p>1 - o usuário seleciona o teste e clica no botão excluir;</p> <p>2 - o sistema valida e exclui o teste.</p>

Quadro 39 - Realiza o teste de hipótese

UC14 – Realiza as análises
<b><i>public UseCase:</i></b> Realizar as análises quantitativa e qualitativa
<b><i>Constraints</i></b>
<ul style="list-style-type: none"> <li>▪ <i>Approved Pré-condição</i> . O usuário deve estar logado no sistema.</li> <li>▪ <i>Approved Pré-condição</i> . O questionário deve ter sido respondido.</li> <li>▪ <i>Approved Pós-condição</i> . Análises efetuadas com sucesso.</li> </ul>
<b><i>Cenários</i></b>
<u>Usuário realiza as análises</u> {Principal}.
<p>1 - o sistema apresenta a tela da análise quantitativa e qualitativa;</p> <p>2 - o usuário informa descrições da análise quantitativa, qualitativa, insere dados da análise quantitativa e clica em adicionar;</p> <p>3 - o sistema valida e grava os dados.</p>
<u>O usuário exclui um dado inserido</u> {Alternativo}.
<p>No passo 2, o usuário opta por excluir um dado inserido</p> <p>1 - o usuário seleciona um dado e clica no botão excluir;</p> <p>2 - o sistema valida as informações e exclui.</p>
<u>O usuário não informou os dados</u> {Exceção}.
No passo 2, se o usuário não informar os dados, apresentar mensagem.
<u>Usuário não escolheu um dado inserido</u> {Exceção}.
No passo de exclusão de um dado inserido, se o usuário não escolher um dado, o sistema apresenta a mensagem "Você deve selecionar um dado".

Quadro 40 - Realiza as análises

UC15 – Descreve as validades
<b><i>public UseCase:</i></b> Permite descrever as validades
<b><i>Constraints</i></b>
<ul style="list-style-type: none"> <li>▪ <i>Approved Pré-condição</i> . O usuário deve estar logado no sistema.</li> <li>▪ <i>Approved Pós-condição</i> . Validades descritas com sucesso.</li> </ul>
<b><i>Cenários</i></b>
<u>Usuário descreve as validades</u> {Principal}.
<p>1 - o sistema apresenta a tela para a descrição das validades;</p> <p>2 - o usuário insere das descrições e clica no botão gravar;</p> <p>3 - o sistema grava as informações.</p>
<u>Usuário altera as descrições</u> {Alternativo}.
<p>No passo 2, o usuário opta por alterar as descrições</p> <p>1 - o usuário alterar as descrições e clica no botão gravar;</p> <p>2 - o sistema grava as informações.</p>

Quadro 41 - Descreve as validades

UC17 – Emite relatórios
<b><i>public UseCase:</i></b> Emitir os relatórios do sistema
<b><i>Constraints</i></b>
▪ <i>Approved Pré-condição</i> . O usuário deve estar logado no sistema.
▪ <i>Approved Pré-condição</i> . Deve existir um experimento aberto.
▪ <i>Approved Pós-condição</i> . Relatórios emitidos com sucesso.
<b><i>Cenários</i></b>
<u>Emite relatórios</u> {Principal}.
1 - O sistema apresenta a tela principal; 2 - O usuário seleciona o menu Apresentação de dados e a opção desejada; 3 - O sistema apresenta o relatório.
<u>Nenhum experimento está aberto</u> {Exceção}.
No passo 2, se nenhum experimento estiver aberto, o sistema apresenta a mensagem "Para esta operação, um experimento deve estar aberto".

Quadro 42 - Emite relatórios

UC18 – Consulta gráficos
<b><i>public UseCase:</i></b> Consultar os gráficos do sistema
<b><i>Constraints</i></b>
▪ <i>Approved Pré-condição</i> . O usuário deve estar logado no sistema.
▪ <i>Approved Pré-condição</i> . Um experimento deve estar aberto.
▪ <i>Approved Pós-condição</i> . Gráficos foram consultados.
<b><i>Cenários</i></b>
<u>Consulta gráficos</u> {Principal}.
1 - O sistema apresenta a tela principal; 2 - O usuário seleciona o menu Apresentação dos dados - Gráficos e escolhe a opção desejada; 3 - O sistema mostra o gráfico na tela.
<u>Nenhum experimento está aberto</u> {Exceção}.
No passo 2, se nenhum experimento estiver aberto, o sistema mostra a mensagem "Para esta operação, um experimento deve estar aberto".

Quadro 43 - Consulta gráficos

## APÊNDICE B – Relatórios e gráficos do sistema

Na figura 32, é apresentado o relatório dos dados do experimento, indicando os dados como nome do participante, instituição, formação, curso, experiência em anos e experiência em nível.

<div style="display: flex; align-items: center;"> <div style="background-color: black; color: white; padding: 2px 5px; margin-right: 10px;">EX</div> <div> <p style="margin: 0;"><b>DADOS GERAIS DO EXPERIMENTO: 1</b></p> <p style="margin: 0; text-align: center;"><b>Participantes</b></p> </div> </div>					
Nome do participante	Instituição	Formação	Curso	Experiência(Anos)	Experiência(Nível)
Jeison	Pública	Técnico	Informática	Menos de 6 Meses	Baixo
Carlos	Pública	Universitária	Informática	De 6 Meses à 2 Anos	Baixo
Roberto	Pública	Pós Graduação	Informática	De 6 Meses à 2 Anos	Baixo
Fábio	Pública	Universitária	Informática	De 2 Anos à 4 Anos	Baixo
Ricardo	Pública	Universitária	Informática	De 2 Anos à 4 Anos	Baixo
Jefferson	Pública	Universitária	Informática	De 2 Anos à 4 Anos	Baixo
Juliano	Pública	Técnico	Informática	De 2 Anos à 4 Anos	Baixo
Fabiano	Pública	Técnico	Informática	Menos de 6 Meses	Baixo
Renata	Particular	Técnico	Informática	De 6 Meses à 2 Anos	Baixo
Catia	Particular	Técnico	Informática	De 2 Anos à 4 Anos	Baixo
César	Particular	Pós Graduação	Informática	Menos de 6 Meses	Baixo
Alexandre	Particular	Pós Graduação	Informática	De 2 Anos à 4 Anos	Baixo
Ronaldo	Pública	Técnico	Informática	Menos de 6 Meses	Baixo

Figura 32 - Relatório dos participantes

Na figura 33, é apresentado o relatório dos objetivos, instrumentação, preparação e recursos utilizados e o contexto do experimento.

<b>Objetivos</b>	
<b>Global</b>	O propósito desse estudo experimento é avaliar se o uso do MODEST no desenvolvimento de SI pode aumentar a capacidade de detecção de falhas dos testes gerados, e ao mesmo tempo reduzir os custos do desenvolvimento, em função da diminuição do esforço necessário na execução das atividades de teste.
<b>Medição</b>	O foco da medição é o esforço(tempo) e a capacidade de detecção de falhas.
<b>Estudo</b>	Analisar <utilização de um processo de software instanciado com ou sem o MODEST> Com o propósito de <avaliar> Com respeito à <esforço e capacidade de detecção de falhas> Do ponto de vista do <pesquisador> No contexto dos <alunos de graduação em Ciências da computação>
<b>Instrumentação</b>	
Na questão de desenvolvimento, foram utilizados dois casos de uso do sistema Mercú, que é um sistema de controle de vendas e de compras de uma mercearia. Os casos de uso selecionados foram: - Login: estimado em 20 pontos de função - Gestão de mercadorias: estimado em 29 pontos de função	
<b>Preparação e recursos utilizados</b>	
O questionário de caracterização dos participantes mostrou que os participantes não possuíam conhecimentos relacionados ao estudo. A maioria dos conhecimentos em Engenharia de Software foram adquiridos durante a própria disciplina. Assim, os grupos tiveram duas horas de aula sobre geração de testes, abordando técnicas de teste como particionamento em classes	
<b>Contexto</b>	
<b>Processo</b>	On-line
<b>Participantes</b>	Alunos
<b>Realidade</b>	Um problema real
<b>Generalidade</b>	Específico

Figura 33 - Dados gerais do estudo

Na figura 34, é apresentado o relatório das análises feitas no experimento.

<b>EX</b>		<b>DADOS ESPECÍFICOS DO EXPERIMENTO: 1</b>			
<b>Análise Quantitativa e qualitativa</b>					
Sequência	Dado 1	Operação	Dado 2	Percentual	Resultado
1	5	/	13	100%	38,46%
3	8	/	13	100%	61,54%
<b>Descrição da análise quantitativa</b>					
O gráfico das variáveis, 1, 2, 3 e 4, evidencia que o tempo de desenho utilizando o MODEST é maior que o tempo para desenho sem utilizá-lo, mas o tempo de desenvolvimento (desenho + testes) sem usar o MODEST é maior que o tempo utilizando-o. O gráfico das variáveis 5 e 6 evidencia que					
<b>Descrição da análise qualitativa</b>					
38,46% do tempo de desenho utilizando o MODEST ficou abaixo da média 61,54% do tempo de desenho utilizando o MODEST ficou acima da média					

Figura 34 - Dados específicos do estudo

Na figura 35, é apresentado o gráfico da análise quantitativa, onde a coluna representa a seqüência das análises e a linha representa o percentual.

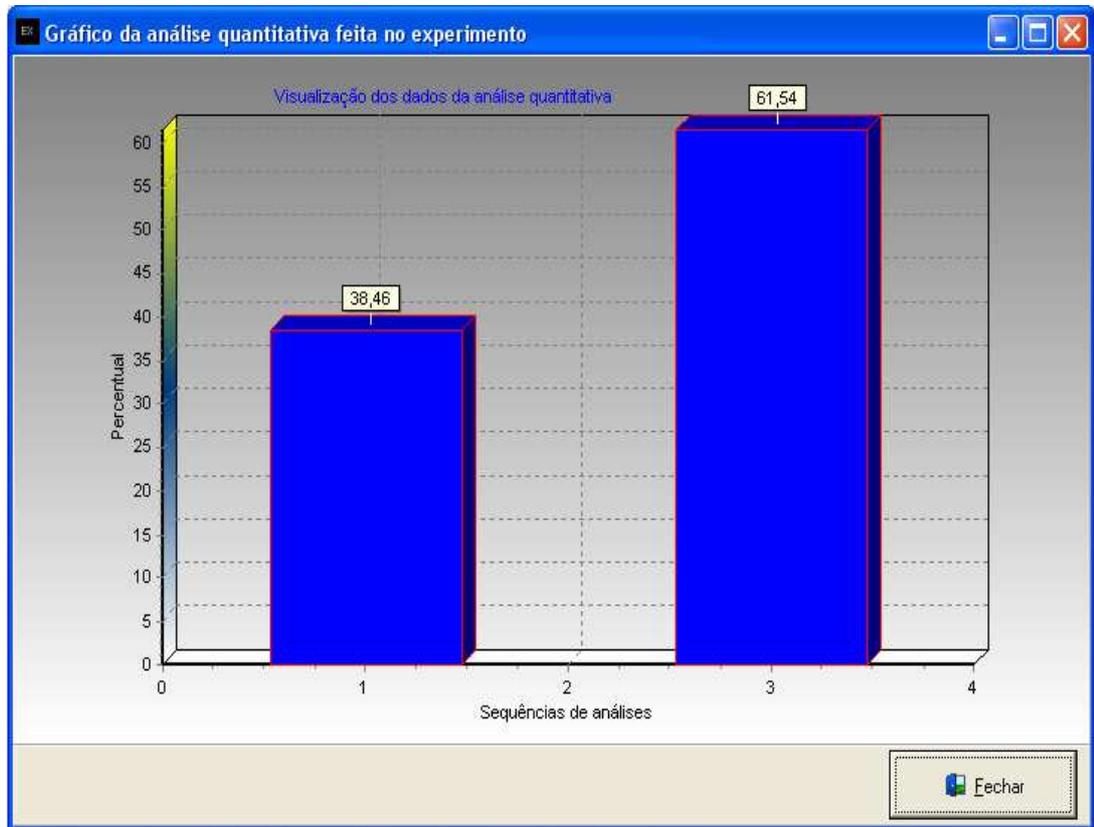


Figura 35 - Gráfico da análise quantitativa

Na figura 36, é apresentado o gráfico dos cálculos de estatística.

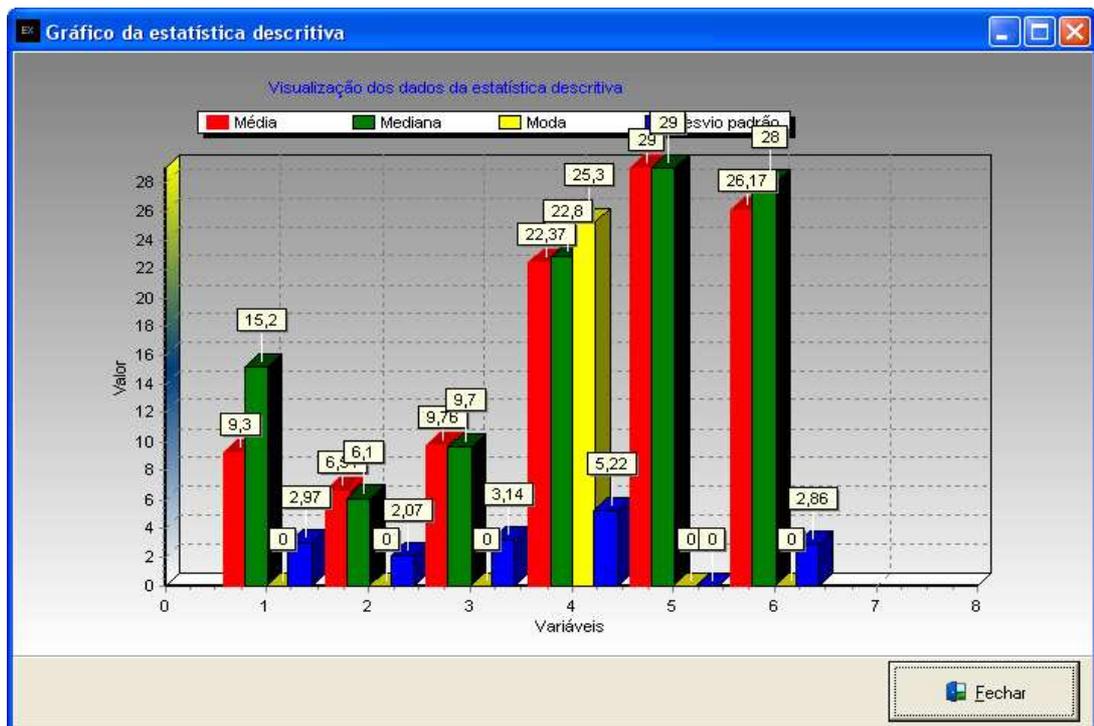


Figura 36 - Gráfico da estatística descritiva

## APÊNDICE C – Quadro do dicionário de dados

No dicionário de dados, o campo “cdexperimento” existe em todas as tabelas, pois indica que todas as tabelas fazem parte de um experimento. Esta ligação da tabela experimento com as demais tabelas do sistema, possibilita o empacotamento dos dados de um experimento por completo. Alguns campos como “vValor1” da tabela hipóteses, foram criados com o tipo “varchar”, para não limitar o sistema somente a um tipo de dado.

No quadro 44, pode ser visualizado o dicionário de dados do sistema.

<b>Hipóteses</b>				
Descrição do campo	Campo	Tipo	pk	Obrigatório
Código do experimento	cdExperimento	int	sim	sim
Código da hipótese	cdHipotese	int	sim	não
Identificação da variável 1	idVariavel1	int	não	não
Descrição condição	dsCondicao	varchar(10)	não	não
Identificação da variável 2	idVariavel2	int	não	não
Valor 1	vValor1	varchar(200)	não	não
Valor 2	vValor2	varchar(200)	não	não
Tipo da hipótese	vTipoHipotese	int	não	não
<b>Estatísticas</b>				
Descrição do campo	Campo	Tipo	pk	Obrigatório
Código do experimento	cdExperimento	int	sim	sim
Tipo da estatística	cdTipoEstatistica	int	sim	sim
<b>DescricaoAnalises</b>				
Descrição do campo	Campo	Tipo	pk	Obrigatório
Código do experimento	cdExperimento	int	sim	sim
Tipo de análise	vTipoAnalise	int	sim	sim
Descrição análise	dsAnalise	varchar(500)	não	não
<b>DadosAnaliseQuantitativa</b>				
Descrição do campo	Campo	Tipo	pk	Obrigatório
Código do experimento	cdExperimento	int	sim	sim
Valor sequência	vSequencia	int	sim	sim
Valor dado 1	vDado1	int	não	não
Valor operação	vOperacao	int	não	não
Valor dado 2	vDado2	int	não	não
Descrição percentual	dsPercentual	varchar(10)	não	não

Valor resultado	vIResultado	decimal(100, 10)	não	não
Descrição análise	dsDescricao	varchar(100)	não	não
<b>RespostasParticipante</b>				
Descrição do campo	Campo	Tipo	pk	Obrigatório
Código do experimento	cdExperimento	int	sim	sim
Código do projeto	CodigoProjeto	int	sim	sim
Código participante	CodigoParticipante	int	sim	sim
Identificação da variável	Variavel	int	sim	sim
Descrição resposta	dsResposta	varchar(50)	não	não
<b>Experimento</b>				
Descrição do campo	Campo	Tipo	pk	Obrigatório
Código do experimento	cdExperimento	int	sim	sim
Descrição experimento	dsExperimento	varchar(50)	não	não
Descrição do Objetivo Global	dsObjetivoGlobal	varchar(500)	não	não
Descrição do Objetivo Medição	dsObjetivoMedicao	varchar(500)	não	não
Descrição do Objetivo Estudo	dsObjetivoEstudo	varchar(500)	não	não
Descrição da Instrumentação	dsInstrumentacao	varchar(500)	não	não
Descrição da preparação	dsPreparacao	varchar(500)	não	não
<b>DescricaoValidades</b>				
Descrição do campo	Campo	Tipo	pk	Obrigatório
Código do experimento	cdExperimento	int	sim	sim
Tipo validade	vITipoValidade	int	sim	sim
Descrição validade	dsValidade	varchar(500)	não	não
<b>GrupoProjetos</b>				
Descrição do campo	Campo	Tipo	pk	Obrigatório
Código do experimento	cdExperimento	int	sim	sim
Código grupo	cdGrupo	int	sim	sim
Descrição grupo	dsGrupo	varchar(100)	não	não
<b>Projetos</b>				
Descrição do campo	Campo	Tipo	pk	Obrigatório
Código do experimento	cdExperimento	int	sim	sim
Código grupo	cdGrupo	int	sim	sim
Código projeto	cdProjeto	int	sim	sim
Descrição projeto	dsProjeto	varchar(100)	não	não
Tempo do projeto	vITempo	int	não	não

Unidade de medida	vUnimed	int	não	não
Tamanho da equipe	vITamanhoEquip	int	não	não
<b>TesteHipotese</b>				
Descrição do campo	Campo	Tipo	pk	Obrigatório
Código do experimento	cdExperimento	int	sim	sim
Código grupo	cdGrupo	int	sim	sim
Código projeto	cdProjeto	int	sim	sim
Identificação da variável 1	idVar1	int	sim	sim
Identificação da variável 2	idVar2	int	sim	sim
Grau de significância	vISignificancia	int	não	não
Limite inferior	vLimiteInferior	decimal(100, 10)	não	não
Limite superior	vLimiteSuperior	decimal(100, 10)	não	não
Resultado do teste	vIResultado	decimal(100, 10)	não	não
Descrição resultado	dsDescricaoResultado	varchar(60)	não	não
<b>Contexto</b>				
Descrição do campo	Campo	Tipo	pk	Obrigatório
Código do experimento	cdExperimento	int	sim	sim
Tipo processo	tpProcesso	int	não	não
Tipo de participantes	tpParticipantes	int	não	não
Tipo realidade	tpRealidade	int	não	não
Tipo generalidade	tpGeneralidade	int	não	não
<b>FatorVariacao</b>				
Descrição do campo	Campo	Tipo	pk	Obrigatório
Código do experimento	cdExperimento	int	sim	sim
Código do fator	cdFator	int	sim	sim
Nome do fator	nmFator	varchar(200)	não	não
<b>Variavel</b>				
Descrição do campo	Campo	Tipo	pk	Obrigatório
Código do experimento	cdExperimento	int	sim	sim
Código do fator	cdFator	int	sim	sim
Identificação da variável	idVariavel	int	sim	sim
Código da variável	cdVariavel	int	não	não
Descrição variável	dsVariavel	varchar(200)	não	não
<b>AgrupamentoVariaveis</b>				

Descrição do campo	Campo	Tipo	pk	Obrigatório
Código do experimento	cdExperimento	int	não	sim
Código do fator	cdFator	int	não	sim
Identificação da variável	idVariavel	int	não	sim
Motivo agrupamento	dsMotivoAgrupamento	varchar(100)	não	não
<b>ValoresPossiveisVariavel</b>				
Descrição do campo	Campo	Tipo	pk	Obrigatório
Código do experimento	cdExperimento	int	sim	sim
Código do fator	cdFator	int	sim	sim
Identificação da variável	idVariavel	int	sim	sim
Sequência valor	seqValor	int	sim	sim
Descrição valor	dsValor	varchar(50)	não	não
<b>Participantes</b>				
Descrição do campo	Campo	Tipo	pk	Obrigatório
Código do experimento	cdExperimento	int	sim	sim
Código do participante	cdParticipante	int	sim	sim
Nome do participante	nmParticipante	varchar(20)	não	não
Código instituição	cdInstituicao	int	não	não
Código curso	cdCurso	int	não	não
Código formação	cdFormacao	int	não	não
Valor experiência	vlExperiencia	int	não	não
Valor nível experiência	vlNivelExperiencia	int	não	não
<b>LoginUsuario</b>				
Descrição do campo	Campo	Tipo	pk	Obrigatório
Código do experimento	cdExperimento	int	não	sim
Código do participante	cdParticipante	int	não	sim
Usuário	nmUsuario	varchar(10)	não	não
Senha	vlSenha	varchar(10)	não	não
E senha mestra	vlSenhaMestra	int	não	não

Quadro 44 - Dicionário de dados do sistema