



UFCG – UNIVERSIDADE FEDERAL DE CAMPINA GRANDE
CEEI – CENTRO DE ENGENHARIA ELÉTRICA E INFORMÁTICA
UASC – UNIDADE ACADÊMICA DE SISTEMAS E COMPUTAÇÃO
CCC – CURSO DE CIÊNCIA DA COMPUTAÇÃO

RELATÓRIO FINAL DE ESTÁGIO

Estágio Integrado – Fábrica de Componentes da Politec (FC-CG)

José Antão Beltrão Moura
Orientador Acadêmico

Luciano de Lima Logrado
Orientador Técnico

Eduardo Santiago Moura
Estagiário

Campina Grande – Paraíba
Maio – 2007



FOLHA DE ASSINATURAS

José Antão Beltrão Moura
Orientador Acadêmico

Luciano de Lima Logrado
Orientador Técnico

Eduardo Santiago Moura
Estagiário



BANCA EXAMINADORA

José Antão Beltrão Moura
Professor / UASC / UFCG

Marcelo Alves de Barros
Professor / UASC / UFCG

Bruno Correia da Nóbrega Queiroz
Chefe do DSC/ Professor / UASC / UFCG



SUMÁRIO

Lista de Abreviaturas e Siglas	05
Seção I	07
1. Introdução	08
1.1 Descrição do Problema	08
1.2 Objetivos	09
1.3 Cronograma de Atividades	09
1.4 Estrutura do Relatório	10
Seção II	11
2. Ambiente de Estágio	12
2.1 A Empresa	12
2.2 Produtos e Serviços	13
2.3 Local de Desenvolvimento	14
2.4 Hardware	14
2.5 Software	15
2.5.1 Sistema Operacional	15
2.5.2 Aplicativos	15
2.5.3 Plataformas	15
Seção III	16
3. Atividades desenvolvidas	17
3.1 Criação de Scripts de Teste de Aceitação	17
3.2 Execução de testes segundo o Controle de Qualidade	19
3.3 Avaliação da qualidade do código produzido	19
3.4 Re-teste dos testes dos demais testadores para detecção de falhas	20
3.5 Fazer reparos no código	20
Seção IV	21
4. Considerações finais	22
Referências Bibliográficas	23
Apêndices	25
Anexos	27



LISTA DE ABREVIATURAS E SIGLAS

ASP: Active Server Pages – tecnologia desenvolvida pela Microsoft que faz parte da plataforma .Net. São páginas web que possuem conteúdo dinâmico, podendo fazer acessos às informações contidas no servidor. Concorrente direta da tecnologia JSP da Sun Microsystems.

CMMI: Capability Maturity Model Integration - é um processo de qualidade de desenvolvimento de software reconhecido mundialmente, de forma a gerar visibilidade em âmbitos nacional e internacional. O nível máximo que uma empresa pode atingir no CMMI é o 5.

DAO: Data Access Object – Padrão de projeto de software onde uma entidade fica responsável pelo acesso ao banco de dados tornando isso transparente às demais camadas do software.

TO: Transfer Object – Padrão de projeto de software em que uma entidade é representada por um objeto que é transferido entre as camadas do software, desde a camada de persistência até a de apresentação.

EJB: Enterprise Java Bean.

FC-CG: Fábrica de Componentes de Campina Grande.

ISO 9001: A série ISO 9000 é um conjunto de normas que formam um modelo de gestão da qualidade para organizações que podem, se desejarem, certificar seus sistemas de gestão através de organismos de certificação.

I18N: Padrão de internacionalização.

SGBD: Um Sistema Gerenciador de Banco de Dados, é o conjunto de programas (softwares) que compõe a camada responsável pelo armazenamento, e recuperação dos dados no Sistema de Informação. O objetivo principal é retirar da camada da Aplicação a responsabilidade dessas tarefas provendo um ambiente mais seguro, mais fácil de manter-se e mais confiável.

JSP: Java Server Pages – tecnologia desenvolvida pela Sun Microsystems de natureza aberta, que faz parte da plataforma J2EE. São páginas web que possuem conteúdo dinâmico, podendo fazer acessos às informações contidas no servidor. Concorrente direta da tecnologia Asp da Microsoft.

JavaScript: JavaScript é uma linguagem de programação criada pela Netscape - em 1995.

J2EE: Java 2 Enterprise Edition – Plataforma de desenvolvimento de software desenvolvida pela Sun Microsystems de natureza aberta e voltada para o ambiente web. É a concorrente direta da plataforma .Net da Microsoft.

MVC: Model View Controller – Padrão de projeto arquitetural de software que proporciona separação e modularização das camadas de interface gráfica (View), lógica de negócios (Model) e fluxo de informações (Controller).

M2M: A M2M Digital é uma empresa de Tecnologia e Serviços para a área de Gestão de Compras.

REsp: Repositório de Especificações – Ferramenta de análise, desenvolvimento e testes de software desenvolvida pela Fábrica de Software de Campina Grande.



SEÇÃO I
INTRODUÇÃO



1. INTRODUÇÃO

O presente relatório descreve as atividades e os resultados de um estágio integrado, desenvolvido na Politec [1] (Fábrica de Componentes) de Campina Grande-PB. O estágio teve como objetivo principal a execução dos testes de controle de qualidade e criação de scripts de teste de aceitação durante a conversão do sistema de leilões e licitações públicas *e-Compras* da empresa M2M Digital [2], que atualmente utiliza a plataforma proprietária ASP, para a plataforma Java utilizando a tecnologia J2EE.

Cada especificação dos problemas será subdividida em tarefas denominadas de “Casos de Uso” (UC). Estes casos serão distribuídos entre os desenvolvedores e cada um terá um prazo para o cumprimento da tarefa, ao concluí-lo o desenvolvedor passará o código resultante para a equipe de testes, que por sua vez será responsável por avaliar a qualidade do código produzido e realizar testes de acordo com os critérios definidos pelo “Controle de Qualidade” (CQ) da empresa.

Para o desenvolvimento dos “Casos de uso”, foram utilizadas as mais novas tecnologias, com o propósito de atingir as necessidades especificadas pelos clientes. Dentre as tecnologias e ferramentas que foram utilizadas, temos: A plataforma J2EE [3], EJB, Struts [4], JSP, Hibernate [5], JavaScript, SGBD(Oracle, Informix, SQLServer, etc), EasyAccept [6], CVS, Xplanner [7] e o REsp (Repositório de especificações) que é um plugin para o Eclipse desenvolvido pela Politec (FC-CG).

1.1 Descrição do Problema

A Politec foi contratada para converter o sistema de leilões e licitações públicas *e-Compras* da empresa M2M Digital, que atualmente utiliza a plataforma proprietária ASP, para a plataforma Java utilizando a tecnologia J2EE, devido às melhorias fornecidas pela última, tais como: portabilidade e facilidade de manutenção de código.

Além dessa conversão, a Politec está modernizando a ferramenta para melhor atender ao seu cliente. A ferramenta possui durante a conversão um banco de dados com aproximadamente 600 tabelas e cerca de 30 páginas escritas em JSP, 300 classes Java e mais 30 arquivos de configuração do projeto.

A conversão emprega tecnologias conhecidas do mundo J2EE, tais como: os frameworks

O cronograma executado apresenta uma diferença de um mês em relação ao cronograma que foi planejado durante o plano de estágio, pois devido a problemas de prazos internos na Politec, o gerente considerou ser melhor o início das atividades do estágio no início do ano de 2007.

Em relação ao cronograma acima, cada célula representa uma semana de trabalho e cada semana corresponde a 20 horas que trabalho. Assim, teve-se um total 320 horas de trabalho durante o estágio.

O acompanhamento das tarefas em ordem de execução pode ser feito consultando-se o Apêndice A.

1.4 Estrutura do Relatório

A Politec recebe muitos estagiários da UFCG/DSC, os quais relatam suas atividades, no final do estágio, com um relatório em formato relativamente padrão - principalmente no que se refere à descrição da empresa e suas atividades. Este relatório não é diferente. Seu conteúdo baseia-se em parte nos relatórios apresentados pelos colegas de trabalho durante o respectivo estágio [9].

Este relatório encontra-se estruturado em quatro seções, da seguinte forma:

A **Seção II** apresenta as informações referentes à EMPRESA POLITEC, local onde o estágio foi realizado, assim como as informações a respeito do hardware e do software utilizado no teste do sistema.

A **Seção III** aborda as atividades de testes desenvolvidas para o sistema e-Compras, que consistem em *execução dos testes de controle de qualidade e criação de scripts de teste de aceitação*.

Na **Seção IV**, são relatadas as *Considerações Finais* relativas ao alcance dos objetivos do trabalho onde serão destacadas as experiências profissionais e pessoais adquiridas durante a execução do estágio.

Posteriormente, são apresentadas as *Referências* dos artigos e ferramentas que deram suporte teórico ao trabalho apresentado.

E por fim, apresenta-se o *Apêndice* seguido dos *Anexos*

SEÇÃO II
AMBIENTE DE ESTÁGIO



2. AMBIENTE DE ESTÁGIO

2.1 A Empresa

A Politec foi constituída em 1970, em Goiânia – GO, tendo como objetivo a prestação de serviços em processamento de dados, na forma de bureau de serviços, para folha de pagamentos, contabilidade, faturamento, controle de estoque de mercadorias, etc.

Chegou a ter em seu cadastro cerca de 1.000 clientes e uma infra-estrutura instalada de cinco computadores de grande porte, distribuídos entre Goiânia e Brasília.

Com o tempo surgem demandas para outros tipos de serviço em decorrência de tecnologias emergentes. São desenvolvidos hardwares mais potentes, microprocessadores mais velozes e os protocolos de comunicação passam a ser mais consistentes. Surgem padrões de mercado para diversas ferramentas de desenvolvimento e a padronização de componentes de hardware. Vão surgindo metodologias estruturadas para desenvolvimento de sistemas, modelagem de dados, orientação a objetos, etc. O mercado continuamente passa por transformações em razão das novas tecnologias.

Com as novas modalidades de administrar os recursos humanos praticados pelas empresas e órgãos governamentais, surge a possibilidade de alocar técnicos para trabalhar nessas organizações, sem vínculo empregatício com as mesmas.

Com base no "know-how" adquirido, após diversos projetos e serviços executados, a empresa passou a atender um maior número de clientes, independente do seu porte e localização, oferecendo serviços com mais tecnologia agregada. A Politec também investiu e adotou uma nova estrutura organizacional, a partir de sua adesão ao processo da Qualidade Total (certificação com base nas normas da família ISO 9001). Com isso, a Politec colocou-se em condições de satisfazer às exigências crescentes de organizações que estabelecem a certificação ISO 9000 como condição para contratação de serviços.

Com o passar dos anos a Politec expandiu-se, sendo hoje uma das grandes empresas privada de serviços de Tecnologia da Informação do Brasil, destacando-se no desenvolvimento, manutenção e produção de sistemas de alta tecnologia. Sediada em Brasília, a Politec mantém escritórios e Fábricas em várias cidades do país como: São Paulo, Rio de Janeiro, Salvador, Aracaju, Belo Horizonte, Curitiba, Goiânia, João Pessoa, Campina Grande, Florianópolis, Recife,

Taquaritinga, Fortaleza e entre outras. A Politec de Campina Grande, denominada Fábrica de Componentes, foi inaugurada em julho de 2005. A Politec tem ainda escritórios fora do país localizado em Reston, VA – EUA e Tóquio - Japão cumprindo assim um papel fundamental na conexão entre a Politec e seus parceiros internacionais, provendo soluções também para o mercado Internacional.

A Politec tem hoje mais de 6.000 funcionários atualizados em seus ambientes de trabalho, tem muitos clientes de renome e importância como: Eletronorte, CHESF, Natura, Ministério da Justiça – MT, Empresa Brasileira de correios e Telégrafos, Caixa Econômica Federal, Banco Real, Bando do Brasil, entre muitos outros, principalmente dos setores públicos brasileiros.

No ano de 2006, a Politec conquistou o selo de maturidade CMMI 5 (Capability Maturity Model Integration), que qualifica o serviço de engenharia de software das empresas. A conquista do nível máximo do CMMI é resultado dos investimentos constantes que a empresa tem feito em qualidade, diminuição do retrabalho e qualificação pessoal através de treinamentos, para promover melhoria nos serviços oferecidos ao Brasil e ao exterior.

2.2 Produtos e Serviços

Os produtos e serviços que a Politec desenvolve são:

- **Serviços de desenvolvimento de software:** para ambientes de servidor/cliente ou baseado na Web.
- **Serviços de Manutenção de Sistemas:** mesmo após sua construção, os sistemas requerem monitoramento contínuo para serem adaptados às mudanças dos ambientes de negócio. Tipicamente, 70% dos orçamentos de TI das corporações são destinados a melhorias constantes, upgrades, soluções para problemas e suporte técnico para sistemas usados, demonstrando a importância da manutenção de sistemas em todo o ciclo de vida.
- **Outsourcing:** para se manterem atualizadas na evolução tecnológica e enfrentarem o aumento da competitividade, as organizações modernas procuram cada vez mais direcionar seus recursos para atividades de fins específicos, enquanto transferem outras atividades para companhias parceiras.
- **Serviços help desk:** são construídos sob os parâmetros do cliente, apoiados por um ponto único de contato para o diagnóstico e solução dos problemas do usuário com o hardware e aplicativos adequados para o negócio.

2.3 Local de Desenvolvimento

O estágio foi desenvolvido no Laboratório de Desenvolvimento da Fábrica de Componentes da Politec de Campina Grande - PB. O laboratório localiza-se no segundo andar do prédio situado à Rua Antenor Navarro, Prata, Cep. 58.101-320. Campina Grande - PB.

O laboratório é dividido em 4 células, onde cada célula possui um Líder, que possui o cargo de Sênior e 6 desenvolvedores, sendo 1 Pleno e os outros Juniores, conforme observado na Figura 01.

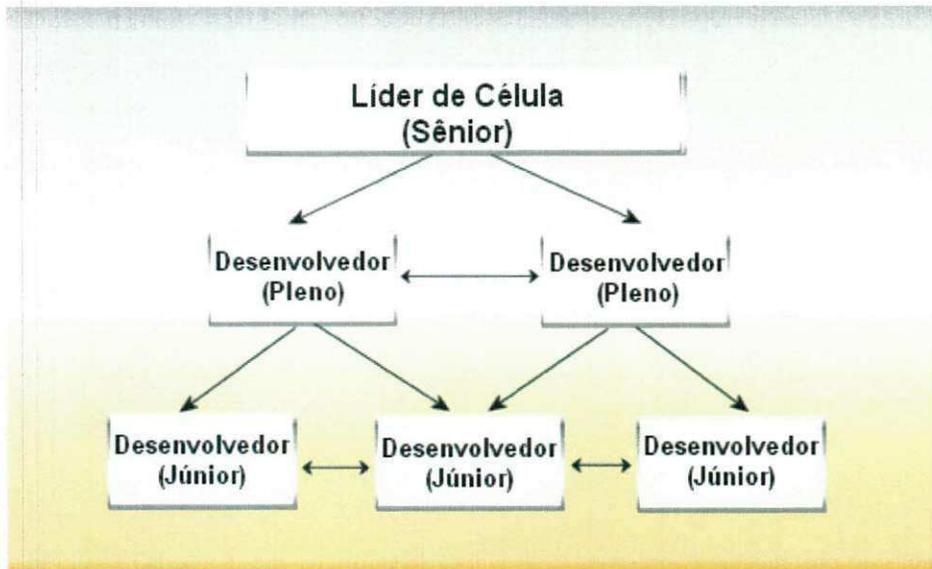


Figura 1: Estrutura das células internas da Politec - CG

2.4 Hardware

Existem 2 configurações de computadores no laboratório:

1. **Pentium4:** Microcomputador com processador Pentium4 de 2.26 GHz, 1 GB de memória RAM;
2. **Celeron:** Microcomputador com processador Celeron de 1.20 GHz, 512 MB de memória RAM;

As categorias acima contam com monitores de 15 polegadas, e placas, de vídeo com 16 MB, de som e de rede com 10/100 Mbits.

2.5 Software

2.5.1 Sistema Operacional

O sistema operacional instalado em todas as máquinas é o Microsoft Windows XP Professional.

2.5.2 Aplicativos

Os aplicativos que rodam nas máquinas para o desenvolvimento dos softwares são:

- **Eclipse:** ferramenta de Software Livre para desenvolvimento de aplicações Java.
- **Tomcat:** servidor de aplicação web Java da Apache versão 5.5. [7]
- **Aqua Data Studio:** ferramenta de Software Livre, para gestão de banco de dados MySQL, Oracle, SQL Server e Postgree, entre outros.
- **Xplanner:** ferramenta de Software Livre para o gerenciamento dos projetos.
- **CVS:** repositório para compartilhamento de versões do projeto.

2.5.3 Plataformas

- J2EE
- JDK 1.5.0_04

SEÇÃO III
ATIVIDADES DESENVOLVIDAS



3. ATIVIDADES DESENVOLVIDAS

Esta seção descreve as atividades desenvolvidas durante o estágio integrado. A proposta deste estágio foi baseada no teste dos “Casos de Uso”, ou seja, na criação e execução de scripts de teste de aceitação relacionados às tarefas especificadas abaixo.

Atividades	Horas estimadas para a realização de cada atividade/ semana
Criação de Scripts de Teste de Aceitação.	2
Execução de testes segundo o Controle de Qualidade.	10
Avaliação da qualidade do código produzido.	2
Re-teste dos testes dos demais testadores para detecção de falhas.	5
Fazer reparos no código.	1
Total	20

Tabela 2: Atividades desenvolvidas para cada Caso de Uso.

3.1 Criação de Scripts de Teste de Aceitação

A etapa de testes tem como meta garantir que a aplicação apresente atributos de qualidade previamente definidos, tal etapa consistiu na elaboração de scripts usando a ferramenta EasyAccept com o intuito de testar os requisitos funcionais do sistema expressados pela M2M Digital.

Para interligar a tecnologia J2EE e o framework EasyAccept foi desenvolvida uma mecanismo pela equipe da Politec denominado *StrutsTestCaseFacade*, que é uma fachada de ligação entre as tecnologias.

A arquitetura do mecanismo desenvolvido pode ser observada na Figura 05.

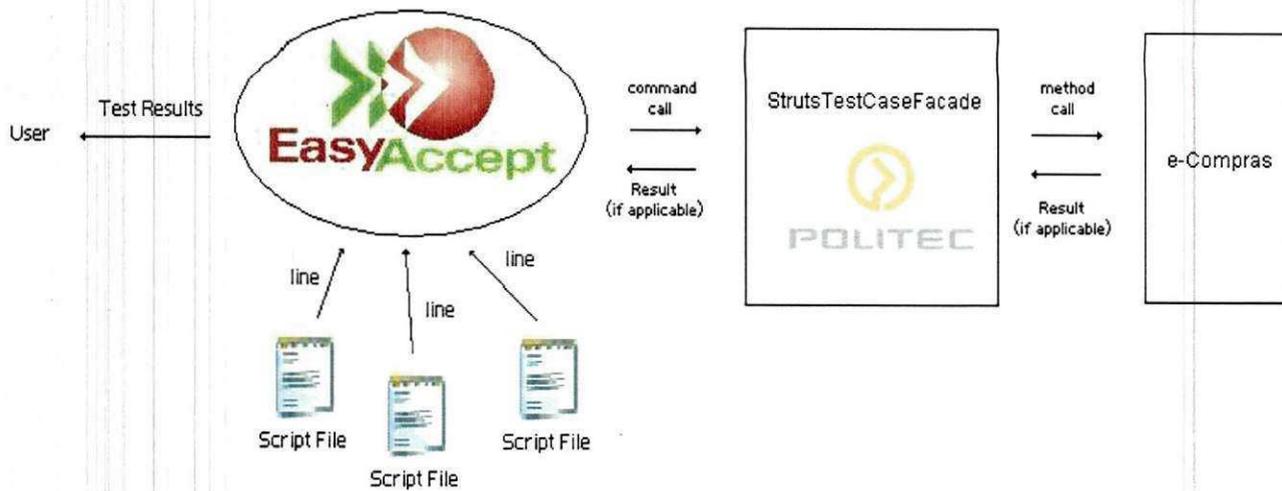


Figura 2: Mecanismo do StrutsTestCaseFacade.

Um trecho de teste de aceitação desenvolvido pode ser visto abaixo:

```
# M2M Digital (www.m2m.digital)
# Projeto de Conversao do E-Compras para Java
# Caso de Teste Basico Banco
# Autor: Eduardo Moura

setUp

#Login
setPropriedadeRequest nome="login" valor=admin
setPropriedadeRequest nome="senha" valor=admin
executaAcao entidade="login" acao="login"
setPropriedadeRequest nome="idCli" valor="1"
setModulo modulo="cadastro"
executaAcao entidade="empresasDisponiveis" acao="escolherEmpresa"
setNomelIdentificador nomelIdentificador="idBanco"

#Listar
executaAcao entidade="banco" acao="listar"
verificaExisteListagem nomeLista="bancoLista"

#Criar
executaAcao entidade="banco" acao="criar"
verificaEncaminhamento nome="cadastro"
setPropriedadeRequest nome="nmBanco" valor="Teste Nome Banco 1"
setPropriedadeRequest nome="nmSiglaBanco" valor="Teste Sigla Banco 1"
setPropriedadeRequest nome="nuBanco" valor="891"
executaAcao entidade="banco" acao="gravar"
verificaNaoHaErros
verificaEncaminhamento nome="principal"
verificaMensagens mensagens="msg.003.01"
limpaRequest
...
```

3.2 Execução de testes segundo o Controle de Qualidade

Os testes de controle de qualidade da empresa foram definidos após anos de experiência que resultaram em uma série de pontos onde o sistema pode falhar, para que isto não ocorra tais pontos devem ser executados exaustivamente para garantia da qualidade e integridade do sistema.

Alguns desses pontos são:

- Verificar regra de acesso à funcionalidade;
- As páginas não devem apresentar erros de JavaScript;
- Se a quantidade máxima de caracteres aceita pelo campo é a mesma descrita na especificação;
- Se, quando houver necessidade, a máscara de entrada de dados está correta e não permite a entrada de dados incorretos;
- Verificar se toda validação é feita via JavaScript no Browser;
- Se paginado, apresenta controles e dados da paginação;
- Todas as páginas e seus componentes estão no padrão (StyleSheet) de tela definido pela M2M;
- Verificar campos não editáveis, ou seja, campos que são alterados apenas no BD segundo especificação;
- Verificar se os dados foram gravados corretamente em todas as tabelas envolvidas no banco de Dados;
- Verificar cada regra de integridade (não importando dependências com outros UC);
- Verificar Encaminhamentos (Forwards), etc.

3.3 Avaliação da qualidade do código produzido

Esta atividade é responsável pela garantia da qualidade técnica do código através da verificação do uso correto dos *Design Patterns* determinados pela empresa. A mesma foi dividida em tarefas que correspondem aos padrões básicos a serem utilizados:

1ª tarefa: Verificação do TO (Transfer Object), que é a classe que manipulam os dados

que são transferidos para o Banco de Dados. Além disso, são configuradas as regras de negócio quando existem;

2ª tarefa: Verificação do DAO (Data Access Object), composto por uma ou mais classes para a persistência dos diferentes tipos de dados do sistema. O acesso ao banco será feito através do Hibernate;

3ª tarefa: Verificação dos métodos da lógica de negócio, que consistem nos métodos: salvar, carregar, editar, excluir e listar;

4ª tarefa: Verificação dos Action Methods, que consistem nos métodos: listar, criar, editar, gravar e excluir. Alguns destes métodos possuem uma mensagem de retorno, seguindo o padrão de internacionalização (I18N), como: "Objeto cadastrado com sucesso", "Objeto alterado com sucesso" etc;

5ª tarefa: Verificação das páginas JSP, geralmente são criadas uma, duas ou três páginas, sendo elas de listagem, cadastro e consulta, que correspondem a cada fluxo do sistema. Cada página está associada a um forward. Nas páginas são criados os fields (campos), os botões, estes são associados a cada Action Method. Em seguida, é feita a configuração da validação de cada campo. Todos os componentes da página possuem seus nomes seguindo o padrão de internacionalização (I18N).

3.4 Re-teste dos testes dos demais testadores para detecção de falhas

Nesta atividade são realizados novamente os testes desempenhados por outro Testador para verificação de erros pendentes, são executados novamente os testes de aceitação através do EasyAccept. Caso haja alguma inconsistência no código, são enviadas mensagens para o desenvolvedor para que este faça as devidas correções no código.

3.5 Fazer reparos no código

Através dos testes realizados anteriormente, para cada teste de aceitação que não passa, é feito um reparo no código criado até que passe em 100% dos testes, para garantir que o código gerado é de qualidade e é robusto.

Por fim, a Revisão final é realizada pelo líder de célula, para garantir que o código gerado está totalmente correto. Só depois que for aprovado pelo líder de célula, é que o caso de uso implementado fica concluído.

SEÇÃO IV
CONSIDERAÇÕES FINAIS



4. CONSIDERAÇÕES FINAIS

A proposta deste estágio foi baseada no teste dos “Casos de Uso”, ou seja, na criação e execução de scripts de teste de aceitação, execução de testes segundo o Controle de Qualidade, entre outras tarefas especificadas anteriormente, ao longo de seu desenvolvimento tais tarefas foram desempenhadas com êxito.

A oportunidade de por em prática teorias compreendidas em ambiente acadêmico é realmente muito gratificante e fundamental na formação profissional de qualquer estudante. Tive o prazer de aprofundar meus conhecimentos em Programação, Banco de Dados, Sistemas de Informação e Engenharia de Software. Além de ter noção de como é trabalhar em uma empresa de grande porte com processos de desenvolvimento consolidados e em andamento.

Desta forma, este estágio possibilitou a execução dos seguintes pontos: utilizar técnicas de Teste de Software para programação Orientada a Objetos, com intuito de prover à aplicação uma maior robustez, eficiência, escalabilidade e corretude; praticar a utilização de “frameworks” da área de desenvolvimento de componentes; praticar a habilidade de adquirir novos conhecimentos necessários para a solução do problema, assimilar novas tecnologias e o desenvolvimento de aplicações com arquitetura J2EE;

Desta forma, este estágio realizado foi de grande valia e possibilitou a solução de problemas como validação de software, verificação da qualidade de software, utilizando as mais diversas tecnologias e metodologias de desenvolvimento de software do mundo computacional moderno.

REFERÊNCIAS BIBLIOGRÁFICAS



REFERÊNCIAS BIBLIOGRÁFICAS

- [1] Politec, Brasília: Empresa **Politec Inteligência em TI**, disponível em <http://www.politec.com.br>. Acesso em 15 de Abril de 2007.
- [2] M2M, Brasília: Empresa **M2M Digital**, disponível em <http://www.m2m.com.br>. Acesso em 18 de Abril de 2007.
- [3] Sun Developer Network Site: **API para a plataforma J2EE**, disponível em http://java.sun.com/j2ee/sdk_1.3/techdocs/api/index.html. Acesso em 18 de Abril de 2007.
- [4] The Apache Software Foundation: **Struts – Framework para aplicações web**, disponível em <http://struts.apache.org>. Acesso em 15 de Abril de 2007
- [5] Relational Persistence for Java and .NET: **Hibernate – Framework para persistência dos dados**, disponível em <http://www.hibernate.org>. Acesso em 15 de Abril de 2002.
- [6] Sauv e, Jacques P. **Manual para a ferramenta de Testes de Aceita o EasyAccept**, dispon vel em <http://jacques.dsc.ufcg.edu.br/projetos/common/easyaccept/usermanual.html>. Acesso em 16 de maio de 2006.
- [7] **Xplanner - Ferramenta de Gerenciamento de Projetos**, dispon vel em <http://www.xplanner.org> Acesso em 18 de Abril de 2007.
- [8] **Test Driven Development**, dispon vel em: http://en.wikipedia.org/wiki/Test_driven_development. Acesso em 18 de Dezembrol de 2006.
- [9] Rocha, Anne. **Relat rio de Est gio Integrado**.

APÊNDICES



APÊNDICE A

Neste apêndice, serão apresentadas as atividades realizadas no estágio em ordem de realização, incluindo todas as tarefas especificadas na seção III para cada Caso de Uso.

Caso de Uso	Data
Atributos Técnicos	11/01/07
Quesitos Técnicos	18/01/07
Banco	23/01/07
País	31/01/07
Característica	08/02/07
Estado	12/02/07
Moeda	17/02/07
Idioma	23/02/07
Empresa	26/02/07
Manutenção de Atas	01/03/07
Fabricante	05/03/07
Mensagem	08/03/07
Publicidade	13/03/07
Endereço	16/03/07
Tipo de Arquivo	20/03/07
Transportadora	23/03/07
Planta	28/03/07
Taxa	02/04/07
Transporte (Tipo de Transporte)	06/04/07
Unidade Orçamentária	10/04/07
Equipe de Trabalho	12/04/07
Requisitos de Licitação	16/04/07
Unidade de Comercialização	20/04/07

ANEXOS



27

POLITEC

```
expect 691 getAtributoStringForm atributo="nuBanco" formName="Banco"
```

#Editar

```
setPropriedadeRequest nome="nmBanco" valor="Novo Teste Nome Banco 1"
```

```
setPropriedadeRequest nome="nmSiglaBanco" valor="Sigla Banco 1"
```

```
setPropriedadeRequest nome="nuBanco" valor=457
```

```
executaAcao entidade="banco" acao="gravar"
```

```
verificaNaoHaErros
```

```
verificaEncaminhamento nome="principal"
```

```
verificaMensagens mensagens="msg.003.02"
```

```
limpaRequest
```

```
# Verificar se a alteração ocorreu
```

```
# Listar
```

```
executaAcao entidade="banco" acao="listar"
```

```
verificaExisteListagem nomeLista="bancoLista"
```

```
breakPoint linha=56
```

```
i=consultarId entidade="banco" propriedade="nmBanco" valor="Novo Teste Nome Banco 1"
```

```
setPropriedadeRequest nome="idBanco" valor=${i}
```

```
executaAcao entidade="banco" acao="editar"
```

```
expect "Novo Teste Nome Banco 1" getAtributoStringForm atributo="nmBanco"
```

```
formName="Banco"
```

```
expect "Sigla Banco 1" getAtributoStringForm atributo="nmSiglaBanco" formName="Banco"
```

```
expect 457 getAtributoStringForm atributo="nuBanco" formName="Banco"
```

```
limpaRequest
```

```
# Excluir
```

```
# Listar
```

```
executaAcao entidade="banco" acao="listar"
```

```
verificaExisteListagem nomeLista="bancoLista"
```

```
i=consultarId entidade="banco" propriedade="nmBanco" valor="Novo Teste Nome Banco 1"
```

```
setPropriedadeRequest nome="idBanco" valor=${i}
```

```
executaAcao entidade="banco" acao="excluir"
```

```
breakPoint linha=74
```

```
verificaNaoHaErros
```

```
verificaEncaminhamento nome="principal"
```

```
verificaMensagens mensagens="msg.003.03"
```

```
tearDown
```

Criar

```
# M2M Digital (www.m2m.com.br)
```

```
# Projeto de Conversao do E-Compras para Java
```

```
# Caso de Teste Criar de Caso de Uso "Banco"
```

```
# Autor: Eduardo Moura
```

```
# Testa se os campos iniciam vazios, após a ação criar
```

```
setUp
```

```
#Login
```

```
setPropriedadeRequest nome="login" valor=admin
setPropriedadeRequest nome="senha" valor=admin
executaAcao entidade="login" acao="login"
setModulo modulo="cadastro"
setPropriedadeRequest nome="idCli" valor="1"
executaAcao entidade="empresasDisponiveis" acao="escolherEmpresa"
```

```
setNomeIdentificador nomeIdentificador="idBanco"
```

```
executaAcao entidade="banco" acao="criar"
expect null getAtributoStringForm atributo="idBanco" formName="Banco"
expect null getAtributoStringForm atributo="nmBanco" formName="Banco"
expect null getAtributoStringForm atributo="nmSiglaBanco" formName="Banco"
expect null getAtributoStringForm atributo="nuBanco" formName="Banco"
verificaEncaminhamento nome="cadastro"
verificaNaoHaErros
verificaNaoHaMensagens
limpaRequest
```

```
# Gravar com o campo nmBanco vazio
setPropriedadeRequest nome="nmBanco" valor=""
setPropriedadeRequest nome="nmSiglaBanco" valor="Sigla CRIAR 01"
setPropriedadeRequest nome="nuBanco" valor="5001"
executaAcao entidade="banco" acao="gravar"
verificaEncaminhamento nome="principal"
verificaErros erros="reg.003.01"
verificaNaoHaMensagens
limpaRequest
```

```
# Gravar com o campo nmBanco null
setPropriedadeRequest nome="nmBanco" valor=null
setPropriedadeRequest nome="nmSiglaBanco" valor="Sigla CRIAR 02"
setPropriedadeRequest nome="nuBanco" valor="5002"
executaAcao entidade="banco" acao="gravar"
verificaEncaminhamento nome="principal"
verificaErros erros="reg.003.01"
verificaNaoHaMensagens
limpaRequest
```

```
# Gravar com o campo nmSiglaBanco vazio
setPropriedadeRequest nome="nmBanco" valor="Nome CRIAR 03"
setPropriedadeRequest nome="nmSiglaBanco" valor=""
setPropriedadeRequest nome="nuBanco" valor="5003"
executaAcao entidade="banco" acao="gravar"
verificaEncaminhamento nome="principal"
verificaErros erros="reg.003.02"
verificaNaoHaMensagens
limpaRequest
```

```
# Gravar com o campo nmSiglaBanco null
setPropriedadeRequest nome="nmBanco" valor="Nome CRIAR 04"
setPropriedadeRequest nome="nmSiglaBanco" valor=null
setPropriedadeRequest nome="nuBanco" valor="5004"
executaAcao entidade="banco" acao="gravar"
```

```
verificaEncaminhamento nome="principal"  
verificaErros erros="reg.003.02"  
verificaNaoHaMensagens  
limpaRequest
```

```
# Gravar com o campo nuBanco vazio  
setPropriedadeRequest nome="nmBanco" valor="Nome CRIAR 05"  
setPropriedadeRequest nome="nmSiglaBanco" valor="Sigla CRIAR 05"  
setPropriedadeRequest nome="nuBanco" valor=""  
executaAcao entidade="banco" acao="gravar"  
verificaEncaminhamento nome="principal"  
verificaErros erros="reg.003.03"  
verificaNaoHaMensagens  
limpaRequest
```

```
# Gravar com o campo nuBanco null  
setPropriedadeRequest nome="nmBanco" valor="Nome CRIAR 06"  
setPropriedadeRequest nome="nmSiglaBanco" valor="Sigla CRIAR 06"  
setPropriedadeRequest nome="nuBanco" valor=null  
executaAcao entidade="banco" acao="gravar"  
verificaEncaminhamento nome="principal"  
verificaErros erros="reg.003.03"  
verificaNaoHaMensagens  
limpaRequest
```

```
# Gravar com o campos vazios  
setPropriedadeRequest nome="nmBanco" valor=""  
setPropriedadeRequest nome="nmSiglaBanco" valor=""  
setPropriedadeRequest nome="nuBanco" valor=""  
executaAcao entidade="banco" acao="gravar"  
verificaEncaminhamento nome="principal"  
verificaErros erros="reg.003.01 reg.003.02 reg.003.03"  
verificaNaoHaMensagens  
limpaRequest
```

```
# Gravar com o campos nullos  
setPropriedadeRequest nome="nmBanco" valor=null  
setPropriedadeRequest nome="nmSiglaBanco" valor=null  
setPropriedadeRequest nome="nuBanco" valor=null  
executaAcao entidade="banco" acao="gravar"  
verificaEncaminhamento nome="principal"  
verificaErros erros="reg.003.01 reg.003.02 reg.003.03"  
verificaNaoHaMensagens  
limpaRequest
```

```
# Cria um banco para testes de duplicações  
setPropriedadeRequest nome="nmBanco" valor="Nome CRIAR 07"  
setPropriedadeRequest nome="nmSiglaBanco" valor="Sigla CRIAR 07"  
setPropriedadeRequest nome="nuBanco" valor="5007"  
executaAcao entidade="banco" acao="gravar"  
verificaEncaminhamento nome="principal"  
verificaNaoHaErros  
verificaMensagens mensagens="msg.003.01"  
limpaRequest
```



Cria um banco obedecendo os limites de tamanho dos campos
setPropriedadeRequest nome="nmBanco" valor="Nome CRIAR 08"
setPropriedadeRequest nome="nmSiglaBanco" valor="Sigla CRIAR 08"
setPropriedadeRequest nome="nuBanco" valor="5008"
executaAcao entidade="banco" acao="gravar"
verificaMensagens mensagens="msg.003.01"
verificaNaoHaErros
limpaRequest

Cria um banco obedecendo o limite de tamanho do campo nmBanco - 60
setPropriedadeRequest nome="nmBanco" valor="Nome CRIAR
09ijklmnopqrstuvwxyzabcdefghijklmnopqrstuvxyzabc"
setPropriedadeRequest nome="nmSiglaBanco" valor="Sigla CRIAR 09"
setPropriedadeRequest nome="nuBanco" valor="5009"
executaAcao entidade="banco" acao="gravar"
verificaMensagens mensagens="msg.003.01"
verificaNaoHaErros
limpaRequest

Cria um banco obedecendo o limite de tamanho do campo nmSiglaBanco - 20
setPropriedadeRequest nome="nmBanco" valor="Nome CRIAR 10"
setPropriedadeRequest nome="nmSiglaBanco" valor="sigla abcdefghijkl12"
setPropriedadeRequest nome="nuBanco" valor="5010"
executaAcao entidade="banco" acao="gravar"
verificaMensagens mensagens="msg.003.01"
verificaNaoHaErros
limpaRequest

Cria um banco obedecendo o limite de tamanho do campo nuBanco - 20
setPropriedadeRequest nome="nmBanco" valor="Nome CRIAR 11"
setPropriedadeRequest nome="nmSiglaBanco" valor="Sigla CRIAR 11"
setPropriedadeRequest nome="nuBanco" valor="12345678901234567890"
executaAcao entidade="banco" acao="gravar"
verificaMensagens mensagens="msg.003.01"
verificaNaoHaErros
limpaRequest

Cria um banco desobedecendo os limites de tamanho do campo nmBanco
setPropriedadeRequest nome="nmBanco" valor="Nome CRIAR
09ijklmnopqrstuvwxyzabcdefghijklmnopqrstuvxyzabc1"
setPropriedadeRequest nome="nmSiglaBanco" valor="Sigla CRIAR 09"
setPropriedadeRequest nome="nuBanco" valor="5009"
executaAcao entidade="banco" acao="gravar"
verificaErros erros="global.err.02"
verificaNaoHaMensagens
limpaRequest

Cria um banco desobedecendo os limites de tamanho do campo nmSiglaBanco
setPropriedadeRequest nome="nmBanco" valor="Nome CRIAR 10"
setPropriedadeRequest nome="nmSiglaBanco" valor="sigla abcdefghijkl123"
setPropriedadeRequest nome="nuBanco" valor="5010"
executaAcao entidade="banco" acao="gravar"



```
verificaErros erros="global.err.02"  
verificaNaoHaMensagens  
limpaRequest
```

```
# Cria um banco desobedecendo os limites de tamanho do campo nuBanco  
setPropriedadeRequest nome="nmBanco" valor="Nome CRIAR 11"  
setPropriedadeRequest nome="nmSiglaBanco" valor="Sigla CRIAR 11"  
setPropriedadeRequest nome="nuBanco" valor="sigla abcdefghijkl123"  
executaAcao entidade="banco" acao="gravar"  
verificaEncaminhamento nome="principal"  
verificaErros erros="global.err.02"  
verificaNaoHaMensagens  
limpaRequest
```

```
# Grava com o num. do banco (nuBanco) recebendo valor diferente de inteiro positivo  
setPropriedadeRequest nome="nmBanco" valor="Nome CRIAR 12"  
setPropriedadeRequest nome="nmSiglaBanco" valor="Sigla CRIAR 12"  
setPropriedadeRequest nome="nuBanco" valor="-0012"  
executaAcao entidade="banco" acao="gravar"  
verificaEncaminhamento nome="principal"  
verificaErros erros="reg.003.04"  
verificaNaoHaMensagens  
limpaRequest
```

```
setPropriedadeRequest nome="nmBanco" valor="Nome CRIAR 13"  
setPropriedadeRequest nome="nmSiglaBanco" valor="Sigla CRIAR 13"  
setPropriedadeRequest nome="nuBanco" valor="302as"  
executaAcao entidade="banco" acao="gravar"  
verificaEncaminhamento nome="principal"  
verificaErros erros="reg.003.04"  
verificaNaoHaMensagens  
limpaRequest
```

```
# Criar um banco com um nmBanco já existente  
setPropriedadeRequest nome="nmBanco" valor="Nome CRIAR 07"  
setPropriedadeRequest nome="nmSiglaBanco" valor="Sigla CRIAR 14"  
setPropriedadeRequest nome="nuBanco" valor="5014"  
executaAcao entidade="banco" acao="gravar"  
verificaEncaminhamento nome="cadastro"  
verificaErros erros="reg.003.06"  
verificaNaoHaMensagens  
limpaRequest
```

```
# grava com caracteres especiais  
setPropriedadeRequest nome="nmBanco" valor="Nome CRIAR%"  
setPropriedadeRequest nome="nmSiglaBanco" valor="Sigla%"  
setPropriedadeRequest nome="nuBanco" valor="41"  
executaAcao entidade="banco" acao="gravar"  
verificaNaoHaErros  
verificaEncaminhamento nome="principal"  
verificaMensagens mensagens="msg.003.01"  
limpaRequest
```



```
setPropriedadeRequest nome="nmBanco" valor="%CRIAR%"
setPropriedadeRequest nome="nmSiglaBanco" valor="%Sigla%"
setPropriedadeRequest nome="nuBanco" valor="40"
executaAcao entidade="banco" acao="gravar"
verificaNaoHaErros
verificaEncaminhamento nome="principal"
verificaMensagens mensagens="msg.003.01"
limpaRequest
```

```
setPropriedadeRequest nome="nmBanco" valor="?ome CRIAR 07"
setPropriedadeRequest nome="nmSiglaBanco" valor="Sigla CRIA? 14"
setPropriedadeRequest nome="nuBanco" valor="50"
executaAcao entidade="banco" acao="gravar"
verificaNaoHaErros
verificaEncaminhamento nome="principal"
verificaMensagens mensagens="msg.003.01"
limpaRequest
```

Grava com registros duplicados com espaços em branco e case sensitive

```
setPropriedadeRequest nome="nmBanco" valor=" Nome CRIAR 07"
setPropriedadeRequest nome="nmSiglaBanco" valor="Sigla CRIAR 15"
setPropriedadeRequest nome="nuBanco" valor="5015"
executaAcao entidade="banco" acao="gravar"
verificaEncaminhamento nome="cadastro"
verificaErros erros="reg.003.06"
verificaNaoHaMensagens
limpaRequest
```

```
setPropriedadeRequest nome="nmBanco" valor="Nome CRIAR 07"
setPropriedadeRequest nome="nmSiglaBanco" valor="Sigla CRIAR 16"
setPropriedadeRequest nome="nuBanco" valor="5016"
executaAcao entidade="banco" acao="gravar"
verificaEncaminhamento nome="cadastro"
verificaErros erros="reg.003.06"
verificaNaoHaMensagens
limpaRequest
```

```
setPropriedadeRequest nome="nmBanco" valor="NOME CRIAR 07"
setPropriedadeRequest nome="nmSiglaBanco" valor="Sigla CRIAR 17"
setPropriedadeRequest nome="nuBanco" valor="5017"
executaAcao entidade="banco" acao="gravar"
verificaEncaminhamento nome="cadastro"
verificaErros erros="reg.003.06"
verificaNaoHaMensagens
limpaRequest
```

Criar um banco com um nuBanco já existente

```
setPropriedadeRequest nome="nmBanco" valor="Nome CRIAR 18"
setPropriedadeRequest nome="nmSiglaBanco" valor="Sigla CRIAR 18"
setPropriedadeRequest nome="nuBanco" valor="5007"
executaAcao entidade="banco" acao="gravar"
verificaEncaminhamento "cadastro"
verificaErros erros="reg.003.07"
verificaNaoHaMensagens
```



limpaRequest

```
# Remove os bancos criados para os testes de duplicação  
executaAcao entidade="banco" acao="listar"  
executaAcao entidade="banco" acao="preencherBancoLookup"  
verificaExisteListagem nomeLista="bancoLista"
```

```
i=consultarId entidade="banco" propriedade="nmBanco" valor="Nome CRIAR 07"  
setPropriedadeRequest nome="idBanco" valor=${i}  
executaAcao entidade="banco" acao="excluir"  
verificaMensagens mensagens="msg.003.03"  
verificaNaoHaErros  
limpaRequest
```

```
executaAcao entidade="banco" acao="listar"  
executaAcao entidade="banco" acao="preencherBancoLookup"
```

```
i=consultarId entidade="banco" propriedade="nmBanco" valor="Nome CRIAR 08"  
setPropriedadeRequest nome="idBanco" valor=${i}  
executaAcao entidade="banco" acao="excluir"  
verificaMensagens mensagens="msg.003.03"  
verificaNaoHaErros  
limpaRequest
```

```
i=consultarId entidade="banco" propriedade="nmBanco" valor="Nome CRIAR%"  
setPropriedadeRequest nome="idBanco" valor=${i}  
executaAcao entidade="banco" acao="excluir"  
limpaRequest
```

```
i=consultarId entidade="banco" propriedade="nmBanco" valor="%CRIAR%"  
setPropriedadeRequest nome="idBanco" valor=${i}  
executaAcao entidade="banco" acao="excluir"  
limpaRequest
```

```
i=consultarId entidade="banco" propriedade="nmBanco" valor="Nome CRIAR  
09ijklmnopqrstuvwxyzabcdefghijklmnopqrstuvwxyzabc"  
setPropriedadeRequest nome="idBanco" valor=${i}  
executaAcao entidade="banco" acao="excluir"  
limpaRequest
```

```
i=consultarId entidade="banco" propriedade="nmBanco" valor="Nome CRIAR 10"  
setPropriedadeRequest nome="idBanco" valor=${i}  
executaAcao entidade="banco" acao="excluir"  
limpaRequest
```

```
i=consultarId entidade="banco" propriedade="nmBanco" valor="Nome CRIAR 11"  
setPropriedadeRequest nome="idBanco" valor=${i}  
executaAcao entidade="banco" acao="excluir"  
limpaRequest
```

```
i=consultarId entidade="banco" propriedade="nmBanco" valor="Nome CRIAR 07"  
setPropriedadeRequest nome="idBanco" valor=${i}  
executaAcao entidade="banco" acao="excluir"  
tearDown
```



Editar

```
# M2M Digital (www.m2m.com.br)
# Projeto de Conversao do E-Compras para Java
# Caso de Teste do Fluxo Alternativo Editar do Caso de Uso Banco
# Autor: Eduardo Moura
```

```
setUp
```

```
#Login
```

```
setPropriedadeRequest nome="login" valor=admin
setPropriedadeRequest nome="senha" valor=admin
executaAcao entidade="login" acao="login"
setModulo modulo="cadastro"
setPropriedadeRequest nome="idCli" valor="1"
executaAcao entidade="empresasDisponiveis" acao="escolherEmpresa"
```

```
setNomeIdentificador nomeIdentificador="idBanco"
```

```
# Editar sem selecionar um banco
```

```
executaAcao entidade="banco" acao="editar"
verificaEncaminhamento nome="principal"
verificaErros erros="err.003.01"
verificaNaoHaMensagens
limpaRequest
```

```
# Editar passando idBanco zero
```

```
setPropriedadeRequest nome="idBanco" valor=0
executaAcao entidade="banco" acao="editar"
verificaEncaminhamento nome="principal"
verificaErros erros="global.err.03"
verificaNaoHaMensagens
limpaRequest
```

```
# Editar passando idBanco inválido
```

```
setPropriedadeRequest nome="idBanco" valor="abc"
executaAcao entidade="banco" acao="editar"
verificaEncaminhamento nome="principal"
verificaErros erros="global.err.01"
verificaNaoHaMensagens
limpaRequest
```

```
# Criar os Bancos que serão utilizados nos testes abaixo
```

```
setPropriedadeRequest nome="nmBanco" valor="Nome EDITAR 01"
setPropriedadeRequest nome="nmSiglaBanco" valor="Sigla EDITAR 01"
setPropriedadeRequest nome="nuBanco" valor="5001"
executaAcao entidade="banco" acao="gravar"
verificaMensagens mensagens="msg.003.01"
limpaRequest
```

```
setPropriedadeRequest nome="nmBanco" valor="Nome EDITAR 02"
setPropriedadeRequest nome="nmSiglaBanco" valor="Sigla EDITAR 02"
setPropriedadeRequest nome="nuBanco" valor="5002"
executaAcao entidade="banco" acao="gravar"
```



*verificaMensagens mensagens="msg.003.01"
limpaRequest*

Alterar o nmBanco desse banco para outro já existente

Listar

*executaAcao entidade="banco" acao="listar"
verificaExisteListagem nomeLista="bancoLista"*

*i=consultarId entidade="banco" propriedade="nmBanco" valor="Nome EDITAR 02"
setPropriedadeRequest nome="idBanco" valor=\${i}*

executaAcao entidade="banco" acao="editar"

setPropriedadeRequest nome="nmBanco" valor="Nome EDITAR 01"

executaAcao entidade="banco" acao="gravar"

verificaEncaminhamento nome="cadastro"

verificaErros erros="reg.003.06"

verificaNaoHaMensagens

limpaRequest

Alterar o nuBanco desse banco para outro já existente

Listar

*executaAcao entidade="banco" acao="listar"
verificaExisteListagem nomeLista="bancoLista"*

i=consultarId entidade="banco" propriedade="nmBanco" valor="Nome EDITAR 02"

setPropriedadeRequest nome="idBanco" valor=\${i}

executaAcao entidade="banco" acao="editar"

setPropriedadeRequest nome="nuBanco" valor="5001"

executaAcao entidade="banco" acao="gravar"

verificaEncaminhamento nome="cadastro"

verificaErros erros="reg.003.07"

verificaNaoHaMensagens

limpaRequest

modificar o nome do banco usando caracteres especiais

Listar

*executaAcao entidade="banco" acao="listar"
verificaExisteListagem nomeLista="bancoLista"*

i=consultarId entidade="banco" propriedade="nmBanco" valor="Nome EDITAR 02"

setPropriedadeRequest nome="idBanco" valor=\${i}

executaAcao entidade="banco" acao="editar"

setPropriedadeRequest nome="nmBanco" valor="Nome EDITAR%"

executaAcao entidade="banco" acao="gravar"

verificaEncaminhamento nome="principal"

verificaNaoHaErros

verificaMensagens mensagens="msg.003.02"

limpaRequest

i=consultarId entidade="banco" propriedade="nmBanco" valor="Nome EDITAR 01"

setPropriedadeRequest nome="idBanco" valor=\${i}

executaAcao entidade="banco" acao="editar"

setPropriedadeRequest nome="nmBanco" valor="%Nome EDITAR"

executaAcao entidade="banco" acao="gravar"

verificaEncaminhamento nome="principal"



verificaNaoHaErros

verificaMensagens mensagens="msg.003.02"

limpaRequest

i=consultarId entidade="banco" propriedade="nmBanco" valor="%Nome EDITAR"

setPropriedadeRequest nome="idBanco" valor=\${i}

executaAcao entidade="banco" acao="editar"

setPropriedadeRequest nome="nmBanco" valor="Nome EDITAR 01"

executaAcao entidade="banco" acao="gravar"

verificaEncaminhamento nome="principal"

verificaNaoHaErros

verificaMensagens mensagens="msg.003.02"

limpaRequest

i=consultarId entidade="banco" propriedade="nmBanco" valor="Nome EDITAR%"

setPropriedadeRequest nome="idBanco" valor=\${i}

executaAcao entidade="banco" acao="editar"

setPropriedadeRequest nome="nmBanco" valor="Nome EDITAR 02"

executaAcao entidade="banco" acao="gravar"

verificaEncaminhamento nome="principal"

verificaNaoHaErros

verificaMensagens mensagens="msg.003.02"

limpaRequest

Modificando o campo nuBanco para um valor negativo

Listar

executaAcao entidade="banco" acao="listar"

verificaExisteListagem nomeLista="bancoLista"

i=consultarId entidade="banco" propriedade="nmBanco" valor="Nome EDITAR 02"

setPropriedadeRequest nome="idBanco" valor=\${i}

executaAcao entidade="banco" acao="editar"

setPropriedadeRequest nome="nuBanco" valor="-5003"

executaAcao entidade="banco" acao="gravar"

verificaEncaminhamento nome="principal"

verificaErros "reg.003.04"

verificaNaoHaMensagens

limpaRequest

Modificando o campo nuBanco um valor inválido

Listar

executaAcao entidade="banco" acao="listar"

verificaExisteListagem nomeLista="bancoLista"

i=consultarId entidade="banco" propriedade="nmBanco" valor="Nome EDITAR 02"

setPropriedadeRequest nome="idBanco" valor=\${i}

executaAcao entidade="banco" acao="editar"

setPropriedadeRequest nome="nuBanco" valor="abc"

executaAcao entidade="banco" acao="gravar"

verificaEncaminhamento nome="principal"

verificaErros "reg.003.04"

verificaNaoHaMensagens

limpaRequest



```
#Alterando um Banco e verificando se ainda existe um registro com o nome do banco antigo
executaAcao entidade="banco" acao="listar"
verificaExisteListagem nomeLista="bancoLista"

i=consultarId entidade="banco" propriedade="nmBanco" valor="Nome EDITAR 01"
setPropriedadeRequest nome="idBanco" valor=${i}
executaAcao entidade="banco" acao="editar"
setPropriedadeRequest nome="nmBanco" valor="Nome EDITAR 04"
setPropriedadeRequest nome="nmSiglaBanco" valor="Sigla EDITAR 04"
setPropriedadeRequest nome="nuBanco" valor="5004"
executaAcao entidade="banco" acao="gravar"
verificaMensagens mensagens="msg.003.02"
verificaNaoHaErros
limpaRequest

executaAcao entidade="banco" acao="listar"
verificaExisteListagem nomeLista="bancoLista"

expect -1 consultarId entidade="banco" propriedade="nmBanco" valor="Nome EDITAR 01"
limpaRequest

#Gravar com id Invalido
executaAcao entidade="banco" acao="listar"
verificaExisteListagem nomeLista="bancoLista"

i=consultarId entidade="banco" propriedade="nmBanco" valor="Nome EDITAR 04"
setPropriedadeRequest nome="idBanco" valor=${i}
executaAcao entidade="banco" acao="editar"

setPropriedadeRequest nome="idBanco" valor="abc"
executaAcao entidade="banco" acao="gravar"
verificaErros erros="global.err.01"

#Gravar com id negativo
executaAcao entidade="banco" acao="listar"
verificaExisteListagem nomeLista="bancoLista"

i=consultarId entidade="banco" propriedade="nmBanco" valor="Nome EDITAR 04"
setPropriedadeRequest nome="idBanco" valor=${i}
executaAcao entidade="banco" acao="editar"

setPropriedadeRequest nome="idBanco" valor=-11
executaAcao entidade="banco" acao="gravar"
verificaErros erros="global.err.05"

#Gravar com id diferente do inicial
executaAcao entidade="banco" acao="listar"
verificaExisteListagem nomeLista="bancoLista"

i=consultarId entidade="banco" propriedade="nmBanco" valor="Nome EDITAR 04"
setPropriedadeRequest nome="idBanco" valor=${i}
executaAcao entidade="banco" acao="editar"

setPropriedadeRequest nome="idBanco" valor=999999999
```

```
executaAcao entidade="banco" acao="gravar"  
verificaErros erros="global.err.05"
```

```
# Apagar Bancos criados para teste  
## Listar  
executaAcao entidade="banco" acao="listar"  
verificaExisteListagem nomeLista="bancoLista"
```

```
i=consultarId entidade="banco" propriedade="nmBanco" valor="Nome EDITAR 01"  
setPropriedadeRequest nome="idBanco" valor=${i}  
executaAcao entidade="banco" acao="excluir"  
limpaRequest
```

```
i=consultarId entidade="banco" propriedade="nmBanco" valor="Nome EDITAR 02"  
setPropriedadeRequest nome="idBanco" valor=${i}  
executaAcao entidade="banco" acao="excluir"  
verificaMensagens mensagens="msg.003.03"  
verificaNaoHaErros  
limpaRequest
```

```
i=consultarId entidade="banco" propriedade="nmBanco" valor="Nome EDITAR 04"  
setPropriedadeRequest nome="idBanco" valor=${i}  
executaAcao entidade="banco" acao="excluir"  
verificaMensagens mensagens="msg.003.03"  
verificaNaoHaErros  
tearDown
```

Excluir

```
# M2M Digital (www.m2m.com.br)  
# Projeto de Conversao do E-Compras para Java  
# Caso de Teste do Fluxo Alternativo Excluir Banco  
# Autor: Eduardo Moura
```

```
# Excluir sem selecionar o banco  
setUp
```

```
#Login  
setPropriedadeRequest nome="login" valor=admin  
setPropriedadeRequest nome="senha" valor=admin  
executaAcao entidade="login" acao="login"  
setModulo modulo="cadastro"  
setPropriedadeRequest nome="idCli" valor="1"  
executaAcao entidade="empresasDisponiveis" acao="escolherEmpresa"
```

```
setNomeIdentificador nomeIdentificador="idBanco"  
executaAcao entidade="banco" acao="excluir"  
verificaEncaminhamento nome="principal"  
verificaErros erros="err.003.01"  
verificaNaoHaMensagens
```

limpaRequest

```
# Excluir com idBanco = 0  
setPropriedadeRequest nome="idBanco" valor=0  
executaAcao entidade="banco" acao="excluir"  
verificaEncaminhamento nome="principal"  
verificaErros erros="global.err.03"  
limpaRequest
```

```
# Excluir com idBanco vazio  
setPropriedadeRequest nome="idBanco" valor=""  
executaAcao entidade="banco" acao="excluir"  
verificaEncaminhamento nome="principal"  
verificaErros erros="err.003.01"  
limpaRequest
```

```
# Excluir com idBanco inválido  
setPropriedadeRequest nome="idBanco" valor="abc"  
executaAcao entidade="banco" acao="excluir"  
verificaEncaminhamento nome="principal"  
verificaErros erros="global.err.01"  
limpaRequest
```

```
#Excluindo um Banco e verificando se realmente foi excluído  
executaAcao entidade="banco" acao="criar"  
setPropriedadeRequest nome="nmBanco" valor="Nome EXCLUIR 01"  
setPropriedadeRequest nome="nmSiglaBanco" valor="Sigla EXCLUIR 01"  
setPropriedadeRequest nome="nuBanco" valor="5001"  
executaAcao entidade="banco" acao="gravar"  
verificaMensagens mensagens="msg.003.01"  
verificaNaoHaErros  
limpaRequest
```

```
#Listar  
executaAcao entidade="banco" acao="listar"  
verificaExisteListagem nomeLista="bancoLista"
```

```
i=consultarId entidade="banco" propriedade="nmBanco" valor="Nome EXCLUIR 01"  
setPropriedadeRequest nome="idBanco" valor=${i}  
executaAcao entidade="banco" acao="excluir"  
verificaMensagens mensagens="msg.003.03"  
limpaRequest
```

```
#Listar  
executaAcao entidade="banco" acao="listar"  
verificaExisteListagem nomeLista="bancoLista"
```

```
expect -1 consultarId entidade="banco" propriedade="nmBanco" valor="Nome EXCLUIR 01"  
tearDown
```

Integridade

M2M Digital (www.m2m.com.br)
Projeto de Conversao do E-Compras para Java
Caso de Teste de Integridade do Caso de Uso Banco
Autor: Eduardo Moura

setUp

#Login

setPropriedadeRequest nome="login" valor=admin
setPropriedadeRequest nome="senha" valor=admin
executaAcao entidade="login" acao="login"
setPropriedadeRequest nome="idCli" valor="1"
setModulo modulo="cadastro"
executaAcao entidade="empresasDisponiveis" acao="escolherEmpresa"

#Criação de entidades para testes de integridade (regra reg.003.05)

setPropriedadeRequest nome="idPais" valor=38
setPropriedadeRequest nome="nmPais" valor="Teste Pais 38"
setPropriedadeRequest nome="nuCadErpPais" valor="Teste ERP Pais 38"
setPropriedadeRequest nome="sgPais" valor="P38"
executaAcao entidade="pais" acao="gravar"
verificaNaoHaErros
verificaMensagens mensagens="msg.004.01"
limpaRequest

Criar uma regio para teste

setPropriedadeRequest nome="regiaoPk__idPaisReg" valor=38
setPropriedadeRequest nome="regiaoPk__idReg" valor="RT"
setPropriedadeRequest nome="nmReg" valor="Regiao Teste"
executaAcao entidade="regiao" acao="gravar"
verificaNaoHaErros
verificaEncaminhamento nome="principal"
verificaMensagens mensagens="msg.006.01"
limpaRequest

Criar estado para teste

setPropriedadeRequest nome="estadoPk__idPaisUf" valor=38
executaAcao entidade="estado" acao="criar"
verificaNaoHaErros
verificaEncaminhamento nome="cadastro"
setPropriedadeRequest nome="estadoPk__idUf" valor="TJ"
setPropriedadeRequest nome="idRegUf" valor="RT"
setPropriedadeRequest nome="nmUf" valor="Rio Grande de Janeiro"
setPropriedadeRequest nome="nuCadErpUf" valor="TJ01"
executaAcao entidade="estado" acao="gravar"
verificaEncaminhamento nome="principal"
verificaNaoHaErros
verificaMensagens mensagens="msg.007.01"
limpaRequest

Criar uma moeda para teste

executaAcao entidade="moeda" acao="criar"



```
setPropriedadeRequest nome="idSimboloMoeda" valor="TR"  
setPropriedadeRequest nome="nmMoeda" valor="Teste Real"  
setPropriedadeRequest nome="dtAtuMoeda" valor="15/02/2006"  
setPropriedadeRequest nome="vIDiaMoeda" valor=2,80  
setPropriedadeRequest nome="vIMesMoeda" valor=2,0  
setPropriedadeRequest nome="nuCadErpMoeda" valor="Teste"  
setPropriedadeRequest nome="nmParidadeMoeda" valor="Real"  
executaAcao entidade="moeda" acao="gravar"  
verificaMensagens mensagens="msg.008.01"  
verificaNaoHaErros  
verificaEncaminhamento nome="principal"  
limpaRequest
```

Cria um Banco para testes

```
executaAcao entidade="banco" acao="criar"  
verificaEncaminhamento nome="cadastro"  
setPropriedadeRequest nome="nmBanco" valor="Teste Nome Banco 1"  
setPropriedadeRequest nome="nmSiglaBanco" valor="Teste Sigla Banco 1"  
setPropriedadeRequest nome="nuBanco" valor="4114"  
executaAcao entidade="banco" acao="gravar"  
verificaNaoHaErros  
verificaEncaminhamento nome="principal"  
verificaMensagens mensagens="msg.003.01"  
limpaRequest
```

#Criação de uma empresa relacionada com o banco criado.

```
setPropriedadeRequest nome="nmCli" valor="Razao Social Teste"  
setPropriedadeRequest nome="nmFantasiaCli" valor="Nome Fantasia Teste"  
setPropriedadeRequest nome="idTpcontCli" valor=1  
setPropriedadeRequest nome="nuFone1Cli" valor="(83)0000-0000"  
setPropriedadeRequest nome="nuFone2Cli" valor="(83)0000-0001"  
setPropriedadeRequest nome="nuFax1Cli" valor="(83)0000-0002"  
setPropriedadeRequest nome="nuFax2Cli" valor="(83)0000-0003"  
setPropriedadeRequest nome="enHomepageCli" valor="www.m2m.com.br"  
setPropriedadeRequest nome="nmEmail1Cli" valor="teste@teste.com.br"  
setPropriedadeRequest nome="nmEmail2Cli" valor="teste2@teste2.com.br"  
setPropriedadeRequest nome="sgCli" valor="ABCDE"  
setPropriedadeRequest nome="tpPessoaCli" valor=0  
setPropriedadeRequest nome="nuCicCli" valor="033.185.704-94"  
setPropriedadeRequest nome="nuInscrEstCli" valor="00001"  
setPropriedadeRequest nome="nuInscrMunCli" valor="00002"  
setPropriedadeRequest nome="nuInscrIndCli" valor="00003"  
setPropriedadeRequest nome="nuCadErpCli" valor=15  
setPropriedadeRequest nome="idPaisCli" valor=38  
setPropriedadeRequest nome="idUfCli" valor="TJ"  
setNomIdentificador nomIdentificador="idBanco"  
idBanco=consultarId entidade="banco" propriedade="nmBanco" valor="Teste Nome Banco 1"  
setPropriedadeRequest nome="idBancoCli" valor=${idBanco}  
setPropriedadeRequest nome="inRankCli" valor="N"  
  
setPropriedadeRequest nome="nuAgenciaCli" valor="00004"  
setPropriedadeRequest nome="nuContacorrenteCli" valor="00005"  
  
setNomIdentificador nomIdentificador="idMoeda"
```

```
idMoeda=consultarId entidade="moeda" propriedade="nmMoeda" valor="Teste Real"
setPropriedadeRequest nome="idMoedaCli" valor=${idMoeda}
setPropriedadeRequest nome="dtVencCadCli" valor="20/01/2007"
setPropriedadeRequest nome="inFornAtivoCli" valor="S"
setPropriedadeRequest nome="inAdmCli" valor="S"
setPropriedadeRequest nome="inCentralCli" valor="S"
executaAcao entidade="empresa" acao="gravar"
verificaNaoHaErros
verificaEncaminhamento nome="principal"
verificaMensagens mensagens="msg.010.01"
limpaRequest
```

#Exclusão do banco e verificação da regra de integridade (reg.003.05)

```
executaAcao entidade="banco" acao="listar"
verificaExisteListagem nomeLista="bancoLista"
```

```
setNomeIdentificador nomeIdentificador="idBanco"
i=consultarId entidade="banco" propriedade="nmBanco" valor="Teste Nome Banco 1"
setPropriedadeRequest nome="idBanco" valor=${i}
executaAcao entidade="banco" acao="excluir"
verificaErros erros="reg.003.05"
verificaNaoHaMensagens
limpaRequest
```

Cria uma empresa Pré-Fornecedora

```
setModulo modulo="precadastro"
executaAcao entidade="preCadastroEmpresa" acao="iniciarCadastroForward"
setPropriedadeRequest nome="idPessoaFornPrefor" valor="1"
executaAcao entidade="preCadastroEmpresa" acao="primeiraEtapa"
setPropriedadeRequest nome="razao social" valor="Empresa Duque de Caxias LTDA"
setPropriedadeRequest nome="fantasia" valor="Duque de Caxias"
setPropriedadeRequest nome="cnpj" valor="23.970.662/0001-90"
setPropriedadeRequest nome="insc. estadual" valor="25.5457"
setPropriedadeRequest nome="endereco" valor="Rua Albuquerque Medeiros, 23a"
setPropriedadeRequest nome="cidade" valor="Campina Grande"
setPropriedadeRequest nome="cep" valor="58602-1"
setPropriedadeRequest nome="pais" valor=""Brasil"
setPropriedadeRequest nome="uf" valor="PB"
setPropriedadeRequest nome="ddd fone" valor="83"
setPropriedadeRequest nome="fone" valor="3421-2030"
```

```
consultarId entidade="" propriedade="" valor=""
setPropriedadeRequest nome="banco" nome=idBanco
setPropriedadeRequest nome="agencia" valor=""
setPropriedadeRequest nome="conta corrente" valor=""
setPropriedadeRequest nome="login" valor=""
setPropriedadeRequest nome="senha" valor=""
setPropriedadeRequest nome="confirma senha" valor=""
setPropriedadeRequest nome="pergunta" valor=""
setPropriedadeRequest nome="resposta" valor=""
executaAcao entidade="preCadastroEmpresa" acao="terceiraEtapa"
# exclui banco com dependencias de pré-fornecedor
#Exclusão das entidades utilizadas no teste
setNomeIdentificador nomeIdentificador="idCli"
```



```
i=consultarId entidade="empresa" propriedade="nmCli" valor="Razao Social Teste"  
excluiEmpresa id=${i}
```

```
setNomeIdentificador nomeIdentificador="idBanco"  
idBanco=consultarId entidade="banco" propriedade="nmBanco" valor="Teste Nome Banco 1"  
setPropriedadeRequest nome="idBanco" valor=${idBanco}  
executaAcao entidade="banco" acao="excluir"  
verificaNaoHaErros  
verificaMensagens mensagens="msg.003.03"  
limpaRequest
```

```
setNomeIdentificador nomeIdentificador="idMoeda"  
idMoeda=consultarId entidade="moeda" propriedade="nmMoeda" valor="Teste Real"  
setPropriedadeRequest nome="idMoeda" valor=${idMoeda}  
executaAcao entidade="moeda" valor="excluir"  
verificaNaoHaErros  
verificaMensagens mensagens="msg.008.03"  
limpaRequest
```

```
setPropriedadeRequest nome="estadoPk__idPaisUf" valor=38  
setPropriedadeRequest nome="estadoPk__idUf" valor="TJ"  
executaAcao entidade="estado" valor="excluir"  
verificaNaoHaErros  
verificaMensagens mensagens="msg.007.03"  
limpaRequest
```

```
setPropriedadeRequest nome="regiaoPk__idPaisReg" valor=38  
setPropriedadeRequest nome="regiaoPk__idReg" valor="RT"  
executaAcao entidade="regiao" valor="excluir"  
verificaNaoHaErros  
verificaMensagens mensagens="msg.006.03"  
limpaRequest
```

```
setPropriedadeRequest nome="idPais" valor=38  
executaAcao entidade="pais" valor="excluir"  
verificaNaoHaErros  
verificaMensagens mensagens="msg.004.03"  
tearDown
```

Permissões

```
# M2M Digital (www.m2m.com.br)  
# Projeto de Conversao do E-Compras para Java  
# Caso de Teste de Permissões do Caso de Uso Banco  
# Autor: Eduardo Moura
```

```
setUp  
setModulo modulo=""
```

```
#Login de Empresa Compradora  
setPropriedadeRequest nome="login" valor=admin  
setPropriedadeRequest nome="senha" valor=admin
```

```
executaAcao entidade="login" acao="login"  
verificaNaoHaErros  
setModulo modulo="cadastro"  
setPropriedadeRequest nome="idCli" valor="2"  
executaAcao entidade="empresasDisponiveis" acao="escolherEmpresa"
```

```
executaAcao entidade="banco" acao="criar"  
verificaErros erros="reg.003.08"  
limpaRequest  
tearDown
```

#Login de Empresa Fornecedora

```
setUp  
setModulo modulo=""  
setPropriedadeRequest nome="login" valor=admin  
setPropriedadeRequest nome="senha" valor=admin  
executaAcao entidade="login" acao="login"  
verificaNaoHaErros  
setPropriedadeRequest nome="idCli" valor="3"  
setModulo modulo="cadastro"  
executaAcao entidade="empresasDisponiveis" acao="escolherEmpresa"  
executaAcao entidade="banco" acao="criar"  
verificaErros erros="reg.003.08"  
limpaRequest  
tearDown
```

#Login de Empresa Administradora

```
setUp  
setModulo modulo=""  
setPropriedadeRequest nome="login" valor=admin  
setPropriedadeRequest nome="senha" valor=admin  
executaAcao entidade="login" acao="login"  
verificaNaoHaErros  
setPropriedadeRequest nome="idCli" valor="1"  
setModulo modulo="cadastro"  
executaAcao entidade="empresasDisponiveis" acao="escolherEmpresa"  
executaAcao entidade="banco" acao="criar"  
verificaNaoHaErros  
verificaEncaminhamento nome="cadastro"  
tearDown
```



ANEXO B

Exemplos do Controle de Qualidade

Nº: 030	
UC: Manutenção de Atas	
Fluxos: 01, 02, 03, 04, 05, 06, 07	
Testador: Eduardo Santiago Moura	
Data: 01/03/2007 Hora: 14:45	
Implementador: Kresller Silva	
Prioridade: 1	
DESCRICAO DO TESTE	STATUS
1. Verificar a prioridade, se for alta, mudar o status do UC/FUC para TESTING;	OK
2. abrir tarefa no XPlanner e Anotar a data/hora de início da revisão do UC/FUC;	OK
3. Analisar a especificação do UC/FUC;	OK
4. Para todo o Caso de uso(Não importa o fluxo testado):	ERRO
4.1. Verificar regra de acesso à funcionalidade:	ERRO
4.1.1. Via menu, fazendo habilitação da função para empresa ou usuário sem permissão;	NA
4.1.2. Via url, caso não tenha acesso via menu;	ERRO
5. Para cada fluxo a ser testado, verificando os seguintes itens:	ERRO
5.1. Rodando os testes no Internet Explorer e no FireFox:	ERRO
5.1.1. As páginas não devem apresentar erros de JavaScript;	ERRO
5.1.2. As páginas não devem apresentar erros com relação ao foco;	NA
5.2. Todos os texto da tela devem ser Internacionalizados, incluindo erros;	OK
5.3. Nenhum elemento da página deve ultrapassar os limites do cabeçalho ou rodapé;	OK
5.4. Verificar se o rastro, na parte superior da página, reflete a ação executada;	ERRO
5.5. Verificar para cada campo do formulário:	ERRO
5.5.1. Se o tamanho do campo está igual ao tamanho do campo no protótipo ou é adequado;	ERRO
5.5.2. Se a quantidade máxima de caracteres aceita pelo campo é a mesma descrita na especificação (propriedade maxlength da tag input);	ERRO
5.5.3. Se, quando houver necessidade, a máscara de entrada de dados está correta e não permite a entrada de dados incorretos;	ERRO
5.6. Verificar se, no local em que as mensagens de erro, sucesso e advertência são exibidas, localiza-se na parte superior do formulário;	OK

5.7. Verificar se as mensagens de erro, sucesso e advertência estão exatamente iguais às descritas no documento de especificação;	ERRO
5.8. Verificar se após uma inclusão ou exclusão os campos do formulário retornam para seus valores default;	ERRO
5.9. Verificar se quando o botão excluir é clicado é exibida uma janela de confirmação de exclusão de registro;	NA
5.10. Verificar se quando o botão cancelar da janela de confirmação de exclusão de registro é clicado, nenhuma operação é realizada;	NA
5.11. Verificar se toda validação é feita via JavaScript no Browser (exceção foreign key e unique);	WAIT
5.12. Verificar se a Interface apresenta dados corretamente na resolução 800x600;	OK
5.13. Todos os dados exibidos estão acessíveis (uso de barra de rolagem);	OK
5.14. Se paginado, apresenta controles e dados da paginação;	ERRO
5.15. Todos os fluxos de erro relacionado ao fluxo testado;	NA
5.16. Caso seja uma consulta, todos os dados especificados na documentação do caso de uso são apresentados;	OK
5.17. ComboBox apresenta opções Todos ou Selecione além das opções válidas;	NA
5.18. Todas as páginas e seus componentes estão no padrão(StyleSheet) de tela definido pela m2m;	ERRO
5.19. Para toda ação que necessita selecionar um objeto, verificar execução sem a seleção do objeto:	OK
5.19.1. Deve exibir um mensagem exigindo a seleção do objeto;	OK
5.20. Não devem apresentar erros de ortografia;	ERRO
5.21. Todos os dados devem estar formatados segundo locale do usuario. (Ex: Moeda, Data)	OK
5.22. Inclusão ou Edição:	ERRO
5.22.1. Verificar campos não editáveis, ou seja, campos que são alterados apenas no BD segundo especificação;	WAIT
5.22.2. Verificar sequencias dos campos na tela;	WAIT
5.22.3. Verificar campos requeridos e seus criticas;	WAIT
5.22.4. Verificar inclusão de caracteres especiais (' , " , * , & , " , \$, # , @ , etc);	WAIT
5.22.5. Verificar inclusão de HTML;	WAIT
5.22.6. Verificar inclusão de espaços vazios;	WAIT
5.22.7. Verificar inclusão de dados incorretos ou incoerentes com o tipo do campo; EX: texto em campo numérico;	WAIT
5.22.8. Tentar incluir grande volume de dados, superior ao suportado pelo campo;	WAIT
5.22.9. Testar os campos colocando somente o limite máximo de dados permitido;	WAIT
5.22.10. Testar regras de unicidade, se for exigido;	WAIT
5.22.11. Caso a funcionalidade trabalhe com arquivo:	OK
5.22.11.1. O arquivo deve ser salvo no PATH definido na especificação;	OK
5.22.11.2. Se no UC tem somente um arquivo, apos incluir ou editar, no PATH deve ter somente um arquivo;	OK
5.22.11.3. Verificar se os dados foram gravados corretamente em todas as tabelas envolvidas no banco de Dados;	OK

Nº: 030		
5.23. Exclusão:		NA
5.23.1.	Verificar cada regra de integridade (não importando dependências com outros UC);	NA
5.23.2.	verificar se todos os registros e campos envolvidos foram limpos ou modificados;	NA
5.23.3.	verificar exclusão dos arquivos caso trabalhe com anexos;	NA
5.24. Padrão de Visualização:		OK
5.24.1.	Textos e Tabelas deve estar alinhados à esquerda da tela;	OK
5.24.2.	Botões devem estar no canto direito inferior da tela;	OK
5.24.3.	Botão retornar deve ser sempre o mais à esquerda e na primeira linha (se ocorrer no UC);	OK
5.25. Verificar Encaminhamentos (Forwards)		ERRO
6. Ao terminar o Check List acima, verificar se há erros:		OK
6.1.	Se houver erros, reportar o erro para o responsável e mudar o status do UC/FUC para FIXING;	OK
6.2.	Senão, fazer uma revisão geral e, caso esteja tudo OK, mudar o status do UC/FUC para OK;	OK
7. Fechar tarefa no Xplanner e Anotar data/hora do término da revisão do UC/FUC;		OK
TERMINO EM: 05/03/2007 HORA: 15:50		OK
Erro Nº:	Descrição:	
4.1.2.	Em qualquer situação o usuário logado está tendo acesso o que não deveria acontecer.	
5.1.1.	O link "Retornar" do fluxo principal está erro de Javascript.	
5.4.	No Fluxo 2.4 – Consultar e no Fluxo 2.7 – Listar está faltando o rastro.	
5.5.1.	Fluxo 2.1 – Criar, o campo "Data Criação" está com tamanho ilimitado.	
5.5.2.	Aplicar maxLength ao campo "Data Criação" do fluxo 2.1.	
5.5.3.	Aplicar a máscara de Datas ao campo "Data Criação".	
5.7.	Fluxo 2.5 – Finalizar: - A mensagem correspondente está sendo exibida na página "Criar", quando o Finalizar é feito da pagina "Visualização de Ata".	
5.8.	Fluxo 2.1 - Criar: - Na hora de Criar uma ata o campo "Título" deveria estar em branco. - Na hora de criar uma ata o campo "Data de Criação" deveria estar preenchido com a data e hora corrente.	
5.14.	Fluxo 2.7 – Listar: - O componente de paginação está dando erro de Javascript ao clicar em ">", ou "<", ou "<<". - As atas listadas sempre são as mesmas, mesmo navegando pelo componente de paginação.	
5.18.	Aplicar o novo CSS a todas as paginas.	

Nº: 030	
5.20.	<p>Fluxo 2.7 - Listar: - Colocar acento em "Circunstanciada" (Circunstânciada).</p>
5.22.	<p>Fluxo 2.1 – Criar: - Quando selecionei um arquivo no componente "DEFINIÇÃO DO DOCUMENTO DA ATA", em seguida cliquei no link "Gravar" de a seguinte exceção:</p> <p>???pt-br.error.primarykey???</p> <p>(ManutencaoAtasFacade.java:87) - error.primarykey br.com.m2m.ecompras.dados.util.ConstraintException: error.primarykey at br.com.m2m.ecompras.dados.util.HibernateUtil.trataException(HibernateUtil.java:178) at br.com.m2m.ecompras.dados.util.PersistenceControl.save(PersistenceControl.java:141) at br.com.m2m.ecompras.dados.util.TransactionDao.saveObject(TransactionDao.java:232) at br.com.m2m.ecompras.dados.util.TransactionDao.save(TransactionDao.java:206) at br.com.m2m.ecompras.licitacao.manutencaoatas.ManutencaoAtasDAO.gravar(ManutencaoAtasDAO.java:180) at br.com.m2m.ecompras.licitacao.manutencaoatas.ManutencaoAtasFacade.gravar(ManutencaoAtasFacade.java:79) at br.com.m2m.ecompras.licitacao.manutencaoatas.ManutencaoAtasDelegate.gravar(ManutencaoAtasDelegate.java:59) at br.com.m2m.ecompras.licitacao.manutencaoatas.ManutencaoAtasAction.gravar(ManutencaoAtasAction.java:276) ... - Servlet.service() for servlet action threw exception javax.servlet.jsp.JspException: ServletException in '/paginas/licitacao/manutencaoatas/cadastrar.jsp': No getter method for property dsAta of bean org.apache.struts.taglib.html.BEAN at org.apache.struts.taglib.tiles.InsertTag\$InsertHandler.doEndTag(InsertTag.java:923) at org.apache.struts.taglib.tiles.InsertTag.doEndTag(InsertTag.java:462) at org.apache.jsp.paginas.layout_005fhoriz_jsp._jspx_meth_tiles_insert_1(org.apache.jsp.paginas.layout_005fhoriz_jsp:254) at org.apache.jsp.paginas.layout_005fhoriz_jsp._jspService(org.apache.jsp.paginas.layout_005fhoriz_jsp:123)</p>
5.25.	<p>Fluxo 2.1 – Criar: - Quando é feito upload de um arquivo o sistema deve permanecer na pagina corrente.</p> <p>Fluxo 2.5 – Finalizar: - Não está permanecendo na mesma página.</p> <p>Fluxo 2.7 – Listar: - Link "Retornar" deveria retornar para a pagina de onde veio a chamada a manutenção de atas, além disso ao clicar em retornar está sendo exibida a seguinte mensagem: - "Selecione o Quesito Técnico."</p>

Observações:**Fluxo 2.1 - Criar:**

- Label "Data e Horário Padrão" não estão exibindo a Data e Hora corrente.
- Ver a questão do Logotipo se é para ter ou não.
- Falta colocar o ":" no label "Nome da Empresa" .
- Tirar o espaço em branco do label "CEP :" é pra ser "CEP:".
- Mesmo quando a Ata está em em Criação (Status igual a zero), não está sendo exibido o link "Criar".
- O componente de Inclusão de Arquivos, só deve aparecer depois da criação com sucesso da ata.
- O a tabela "Definição do Documento da Ata", só deve aparecer depois da criação com sucesso da ata.

Fluxo 2.2 - Aprovar:

** Mesmo logado como administrador de Fornecedora ou Compradora, o link "Aprovar" não aparece então qual usuario logado é um Aprovador de Atas??

Fluxo 2.3 - Atualizar:

- Os dados não estão sendo atualizados.

Fluxo 2.4 - Consultar:

- O label "Criação de Ata" não deve ser concatenado com a descrição do estado da Ata.
- Falta colocar o ":" no label "Código do Participante".
- Label "Data e Horário Padrão" não estão exibindo a Data e Hora corrente.
- Tirar o espaço em branco do label "CEP :" é pra ser "CEP:"
- Falta o separador de linha antes do "CORPO DA DATA".
- No campo "Documentos" o pop up que está sendo exibido não contém o conteúdo do Documento da Ata.
- O campo "Documentos Avulsos" é um componente de Upload quando nao foi feito nenhum upload, ao fazer um upload de um arquivo, ele vira componente de Download.
- No campo "Documentos Avulsos" quando não existe documentos é para exibir a seguinte mensagem: "Não existem Documentos."
- Na listagem das atas não existe uma em que a Fase da Ata esteja em "Aguardando Aprovação" ou em "Aprovação Parcial", por isso não tem como testar sua Aprovação.

** Mesmo logado como administrador de Fornecedora ou Compradora, o link "Aprovar" não aparece então qual usuario logado é um Aprovador de Atas??

Fluxo 2.5 - Finalizar:

- O campo "Situação" da tabela "CORPO DA ATA" não está sendo modificado.
- Quando finalizada uma ata, estão sendo criadas cópias da ata que acabou de ser finalizada, e a ata que foi finalizada nao está sofrendo nenhuma alteração apos a finalização.

Fluxo 2.7 - Listar:

- O campo Documento nao esta como lupa, e nem esta abrindo um pop up que contém o conteúdo do Documento da Ata.

** Alterações em qualquer um desses atributos "ataPk__idLleilaoAta", "ataPk__idFaseAta" e "idEdital", não modificam em nada a listagem das atas????!!!