

UNIVERSIDADE REGIONAL DE BLUMENAU
CENTRO DE CIÊNCIAS EXATAS E NATURAIS
CURSO DE SISTEMAS DE INFORMAÇÃO – BACHARELADO

APLICATIVO PARA CONTROLE DE EVENTOS
EFETUADOS EM ALTERAÇÕES DE FORMULÁRIOS DE
IMPRESSÃO

ALLAN GUILHERME ZIMMERMANN

BLUMENAU
2011

2011/2-02

ALLAN GUILHERME ZIMMERMANN

**APLICATIVO PARA CONTROLE DE EVENTOS
EFETUADOS EM ALTERAÇÕES DE FORMULÁRIOS DE
IMPRESSÃO**

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 Sistemas
de Informação— Bacharelado.

Prof. Wilson Pedro Carli, Mestre – Orientador

**BLUMENAU
2011**

2011/2-02

**APLICATIVO PARA CONTROLE DE EVENTOS
EFETUADOS EM ALTERAÇÕES DE FORMULÁRIOS DE
IMPRESSÃO**

Por

ALLAN GUILHERME ZIMMERMANN

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

Presidente: _____
Prof. Wilson Pedro Carli, Mestre – Orientador, FURB

Membro: _____
Prof. Everaldo Artur Grahl, Mestre – FURB

Membro: _____
Prof. Mauro Marcelo Mattos, Doutor – FURB

Blumenau, 29 de novembro de 2011.

Dedico este trabalho aos meus pais, meus familiares e amigos que sempre me apoiaram e deram forças em todos os desafios da minha vida.

AGRADECIMENTOS

A Deus, por sempre estar presente em todos os momentos de minha vida.

À minha família, sem a qual nunca conseguiria realizar os meus sonhos.

Aos meus amigos, minha segunda família, sempre presentes nos momentos bons e ruins.

Ao meu orientador, Wilson Pedro Carli, por ter acreditado na conclusão deste trabalho e por me ajudar nos momentos de dúvida.

“Se trabalharmos sobre o mármore, um dia ele acabará. Se trabalharmos sobre o metal, um dia o tempo o consumirá. Se erguemos templos, um dia se tornarão pó. Mas se trabalharmos sobre almas jovens e imortais, se nós a imbuirmos com os princípios de justo temor ao criador e amor a humanidade, nós gravaremos sobre essas almas algo que brilhará eternamente. Daqui a cem anos pouco importará o quanto tenhamos acumulado no banco; que tipo de casa, palacete ou carro possuiremos, mas o mundo poderá ser diferente, talvez porque fomos importantes na vida dos jovens”.

Frank Sherman Land

RESUMO

Este trabalho apresenta um aplicativo para ambiente *desktop* que controla os eventos efetuados em alterações que envolvam formulários de impressão que são utilizados na linha de produção de uma fábrica de automóveis. As alterações usam conceitos de boas práticas da biblioteca ITIL para melhor gerenciamento dos serviços de TI. Esta aplicação tem como objetivo fornecer, além de controle, a possibilidade garantir um histórico confiável de todas as alterações efetuadas nos formulários. Foi desenvolvida na linguagem Java, utilizando banco de dados MySQL para armazenamento das informações. Como resultado destaca-se a possibilidade de mapear quem foi o responsável por determinada alteração garantindo a rastreabilidade.

Palavras-chave: Controle de Eventos. Formulários de Impressão. Java. ITIL.

ABSTRACT

This work presents a desktop application that controls the made changes in events involving print forms that are used in the production line of a plant. These changes use concepts of the best practices provided by ITIL library to better manage of IT services. It is intended to provide control and the ability to ensure a reliable history of all changes in the forms. The application is developed in Java, uses the MySQL database to store the data. As a result there is the possibility to find who was responsible for particular change.

Key-words: Events control. Print form. Java. ITIL.

LISTA DE ILUSTRAÇÕES

Figura 1 - <i>Layout</i> da ferramenta de alteração de formulários.....	18
Figura 2 - Fluxo sistema Atual	20
Figura 3 - Controle de Alterações em Formulários	20
Figura 4 - Tela para cadastro de solicitação de mudança	22
Figura 5 - Tela de cadastro de nova mudança	22
Figura 6 - Tela para cadastro de equipamento.....	23
Figura 7 - Fluxo proposto ao utilizar o aplicativo	25
Quadro 1 - Requisitos funcionais	26
Quadro 2 - Requisitos não funcionais	26
Figura 8 - Caso de Uso - Controle de Acesso.....	27
Figura 9 - Caso de Uso - Controle de Eventos	27
Figura 10 - Caso de Uso - Emissão de Relatórios	28
Figura 11 - Modelo Entidade/Relacionamento.....	28
Figura 12 - Criando relatório no iReport.....	30
Figura 13 - Tela de <i>login</i>	30
Figura 14 - Tela principal do aplicativo	31
Figura 15 - Tela de aviso de chamados desatualizados.....	32
Figura 16 - Tela para cadastro de chamado.....	32
Figura 17 - Tela para inserir eventos ao chamado.....	32
Figura 18 - Tela de aviso para inserir um evento	33
Figura 19 - Tela chamado selecionado	33
Figura 20 - Tela de reabertura de chamado	34
Figura 21 - <i>Menu</i> de configuração do aplicativo.....	34
Figura 22 - Tela cadastro de usuário	35
Figura 23 - Tela cadastro de planta	35
Figura 24 - Tela cadastrar tipo de evento	36
Figura 25 - Tela alterar senha	36
Figura 26 - <i>Menu</i> de pesquisa.....	37
Figura 27 - Tela pesquisar chamado.....	37
Figura 28 - <i>Layout</i> de uma pesquisa por chamado	37
Figura 29 - Tela de relatório gerado em PDF.....	38

Figura 30 - Relatório gerado em PDF	38
Figura 31 - Tela pesquisa por formulário	39
Figura 32 - <i>Layout</i> de uma pesquisa por formulário	39
Figura 33 - Tela pesquisa por usuário	40
Figura 34 - <i>Layout</i> de uma pesquisa por usuário.....	40
Figura 35 - Tela pesquisa por evento.....	41
Figura 36 - <i>Layout</i> de uma pesquisa por evento.....	41
Figura 37 - Método para gerar visualização da pesquisa.....	42
Figura 38 - Método para gerar uma pesquisa em PDF.....	43
Figura 39 - Métodos para filtrar tempo e planta.....	44
Quadro 3 - Caso de uso alterar senha	49
Quadro 4 - Caso de uso cadastrar chamados	50
Quadro 5 - Caso de uso cadastrar eventos.....	50
Quadro 6 - Caso de uso cadastrar usuário	51
Quadro 7 - Caso de uso cadastrar fábrica.....	51
Quadro 8 - Caso de uso cadastro de tipo de evento.....	52
Quadro 9 - Caso de uso pesquisa por chamado.....	52
Quadro 10 - Caso de uso pesquisa por formulário	53
Quadro 11 - Caso de uso pesquisa por usuário.....	53
Quadro 12 - Caso de uso pesquisa por evento.....	54
Quadro 13 - Dicionário de dados da tabela usuário.....	55
Quadro 14 - Dicionário de dados da tabela planta.....	55
Quadro 15 - Dicionário de dados da tabela tipo	55
Quadro 16 - Dicionário de dados da tabela chamado.....	56
Quadro 17 - Dicionário de dados da tabela evento.....	56

LISTA DE SIGLAS

IGP – *Intelligent Graphics Printing*

ITIL – *Information Technology Infrastructure Library*

MER – Modelo entidade/relacionamento

PSC – *Peregrine Service Center*

RNF – Requisito não funcional

RF – Requisito funcional

SLA – *Service Level Agreement*

TI – Tecnologia da Informação

UC – *Use case*

VPN – *Virtual Private Network*

SUMÁRIO

1 INTRODUÇÃO.....	12
1.1 OBJETIVOS DO TRABALHO	13
1.2 ESTRUTURA DO TRABALHO	14
2 FUNDAMENTAÇÃO TEÓRICA	15
2.1 ITIL.....	15
2.2 SISTEMA PARA CONTROLE DE LINHA DE PRODUÇÃO.....	16
2.2.1 Ferramenta para alteração de formulários de impressão.....	17
2.3 ABERTURA DE CHAMADO.....	18
2.4 SISTEMA ATUAL	19
2.5 TRABALHOS CORRELATOS	21
3 DESENVOLVIMENTO DO APLICATIVO	24
3.1 LEVANTAMENTO DE INFORMAÇÕES	24
3.2 ESPECIFICAÇÃO	25
3.2.1 Requisitos Funcionais	25
3.2.2 Requisitos não funcionais	26
3.2.3 Casos de Uso	26
3.2.4 Modelo Entidade Relacionamento	28
3.3 IMPLEMENTAÇÃO	29
3.3.1 Técnicas e ferramentas utilizadas	29
3.3.2 Operacionalidade da implementação.....	30
3.4 RESULTADOS E DISCUSSÃO	44
4 CONCLUSÕES.....	46
4.1 EXTENSÕES	47
REFERÊNCIAS BIBLIOGRÁFICAS	48
APÊNDICE A – Detalhamento dos Casos de Uso	49
APÊNDICE B – Dicionário de Dados.....	55

1 INTRODUÇÃO

Frente às mudanças constantes na área da Tecnologia da Informação (TI) e o aumento na concorrência dessa área, surge à necessidade de que as organizações sejam inteligentes e que se modifiquem o que requer um melhor planejamento de suas informações, de seus conhecimentos e de sua tecnologia da informação. Levando isso em consideração, pode-se chegar à conclusão de que as empresas lidam a cada dia com mais dados e esses dados precisam ser devidamente trabalhados para que possam agregar algum valor à organização a fim de gerar conhecimento quando for necessário (REZENDE, 2003).

A cada ano há um aumento no número de responsabilidades dentro das organizações, e com o aumento das responsabilidades vem também o aumento de informações que precisam ser gerenciadas e armazenadas para que estejam disponíveis quando forem solicitadas. Pensando nesses problemas foram criados conjuntos de boas práticas que visam o melhor gerenciamento dessas informações. Segundo Magalhães e Pinheiro (2007, p. 35), “jargões como melhores práticas, otimização de processos, qualidade de serviço e alinhamento estratégico dos serviços de TI deixam de ser meros jogos de palavras e passam a ser parte do novo estilo de vida de todas as áreas de TI”. Sendo assim as melhores práticas tem como objetivo se basear em erros e atribuições já vivenciadas e superadas para que não seja necessário aprender e crescer através de tentativas.

A empresa T-Systems do Brasil é uma dessas organizações que se atualizam frequentemente para continuar em uma posição elevada perante o mercado nacional e internacional. Dentre os projetos de desenvolvimento e suporte existentes na empresa, um deles tem como objetivo prestar suporte ao sistema modular que gerencia a linha de produção das fábricas de uma montadora de automóveis. Este projeto utiliza os conceitos de boas práticas em gerenciamento de serviços de TI para resolver os chamados de suporte ao sistema de uma maneira eficiente e que garanta a satisfação do cliente.

Nas fábricas desta montadora, a equipe de suporte técnico do sistema de planejamento e controle de produção é constituída por funcionários da T-Systems, que trabalham internamente como terceiros e é denominada de 1º nível. Nas fábricas do exterior, o suporte de 1º nível é realizado por funcionários da própria montadora, sendo que a empresa T-Systems do Brasil fornece apenas suporte de 2º nível neste caso.

O suporte prestado pelo 2º nível é remoto, onde é necessário acessar a rede das fábricas através de uma rede privada virtual (da sigla em inglês *Virtual Private Network (VPN)*) para

se ter acesso aos seus servidores e aplicativos. Atualmente existem equipes de 2º nível nos municípios de Blumenau (SC), Curitiba (PR) e São Bernardo do Campo (SP). A equipe de Blumenau trabalha em escalas de 24 horas por dia nos 7 dias da semana, dividida em três turnos.

Com o crescimento do projeto no município de Blumenau e a necessidade de diminuir o número de responsabilidades do 1º nível, por solicitação da própria montadora, novas responsabilidades são repassadas para a equipe de 2º nível. Uma destas responsabilidades é atender todas as solicitações que envolvam formulários de impressão, porém não existe nenhum aplicativo voltado apenas para o controle dos eventos efetuados nas alterações.

Os chamados que envolvem formulários demandam mais tempo para serem finalizados se comparado a outros tipos de incidente. Um chamado pode ficar pendente por vários dias esperando a confirmação de aprovação da solicitação feita pelo cliente. Sendo assim o chamado ficará com o status de “esperando uma resposta” até que o solicitante (cliente) assine o documento de aceite. Essa situação faz com que o chamado seja esquecido no meio de outros que estão abertos.

A falta de um aplicativo específico para controle deste tipo de incidente acaba trazendo problemas para se manter um histórico confiável das alterações.

1.1 OBJETIVOS DO TRABALHO

O objetivo deste trabalho foi desenvolver um aplicativo para controlar os eventos realizados em alterações de formulários de impressão que são utilizados na linha de produção de uma fábrica de automóveis.

Os objetivos específicos do trabalho proposto são:

- a) permitir o registro das alterações com o intuito de gerar uma base de informações visando à emissão de relatórios operacionais;
- b) permitir a visualização dos eventos que foram realizados em cada solicitação;
- c) permitir a verificação do tempo gasto nas solicitações, garantindo o cumprimento do acordo de nível de serviço que é baseado no *framework* de gerenciamento *Information Technology Infrastructure Library* (ITIL);
- d) permitir a rastreabilidade do usuário que inseriu determinado evento, através do controle de acesso ao aplicativo.

1.2 ESTRUTURA DO TRABALHO

Este trabalho está organizado em quatro capítulos, sendo que, no primeiro, foi apresentada a introdução, os objetivos específicos e como o trabalho está estruturado.

No segundo capítulo é apresentada a fundamentação teórica bem como os assuntos que serviram de base para o desenvolvimento do trabalho e a apresentação de trabalhos correlatos.

No terceiro capítulo está descrito o desenvolvimento do aplicativo, as técnicas e ferramentas utilizadas bem como a elaboração de alguns diagramas para auxiliar na compreensão do aplicativo, a operacionalidade do mesmo e resultados e discussões.

No quarto capítulo apresenta-se a conclusão e extensão do trabalho.

2 FUNDAMENTAÇÃO TEÓRICA

Este capítulo aborda assuntos a serem apresentados nas seções a seguir, tais como, biblioteca ITIL, Sistema para Controle de Linha de Produção, ferramenta para alteração de formulários de impressão, abertura de chamado e os trabalhos correlatos.

2.1 ITIL

O Gerenciamento de Serviços de Tecnologia da Informação é o instrumento pelo qual se pode iniciar uma postura pró-ativa visando atender as necessidades da organização, contribuindo para geração de valor.

Segundo Magalhães e Pinheiro (2007, p. 29), este gerenciamento “visa alocar adequadamente os recursos disponíveis e gerenciá-los de forma integrada, fazendo com que a qualidade do conjunto seja percebida pelos seus clientes e usuários”. A tática utilizada para alcançar estes objetivos é gerenciar os processos internos da área de TI utilizando as práticas da biblioteca ITIL.

A *Information Technology Infrastructure Library* (ITIL) é a biblioteca mais utilizada para o Gerenciamento de Serviços de TI no mundo. Foi criada pelo Governo Britânico em 1980 para resolver as necessidades de padronização dos processos da área de TI e baseia-se na experiência coletiva de inúmeros praticantes de organizações privadas e públicas de todo o mundo.

Ela provê um abrangente e consistente conjunto de melhores práticas para melhorar o alinhamento dos processos de TI com os serviços e necessidades da organização. Prove uma abordagem qualitativa da infra-estrutura de TI com o objetivo de obter vantagem em termos de redução de custos pelo aumento da eficiência na entrega e suporte dos serviços e através do aumento na capacidade de gerar receita permitindo que os esforços sejam direcionados para novos projetos (MAGALHÃES; PINHEIRO, 2007).

As alterações em formulários de impressão são caracterizadas como Requisição de Mudança que faz parte do processo de Gerenciamento de Mudanças da biblioteca ITIL. Este processo é projetado para agir como um processo de planejamento e controle, que asseguram

que a execução da mudança ocorra sem interromper, ou pelo menos interrompendo o mínimo possível a operação dos serviços de TI (MAGALHÃES; PINHEIRO, 2007).

Outra boa prática da biblioteca ITIL bastante utilizada é o escalonamento, onde existem níveis de trabalho que possuem responsabilidades distintas. Sendo assim, “O escalonamento de um incidente durante o seu atendimento é um mecanismo utilizado para se obter a resolução do incidente dentro do menor período de tempo possível, garantindo a disponibilização do conhecimento e os recursos necessários” (MAGALHÃES; PINHEIRO, 2007).

O escalonamento também permite garantir melhor suporte técnico, mais lucratividade e um número maior de pessoas atuando nas tomadas de decisão, a fim de resolver o incidente dentro dos níveis acordados no *Service Level Agreement* (SLA). Sua utilização também serve para manter níveis hierárquicos superiores a par do progresso no atendimento de um incidente ou para, em caso de erro no processo solicitar apoio para resolução da falha (MAGALHÃES; PINHEIRO, 2007).

2.2 SISTEMA PARA CONTROLE DE LINHA DE PRODUÇÃO

Uma linha de produção é “um conjunto de estações de trabalho seqüencialmente dispostas, normalmente interligadas por um sistema contínuo de movimentação de materiais, e projetada para montar componentes e realizar qualquer operação necessária à obtenção de um produto acabado.” (ASKIN; STANDRIDGE, 1993).

O sistema utilizado pela montadora é responsável por gerenciar toda linha de produção, ou seja, gerencia desde a armação da carroceria (onde as chapas de aço são soldadas), passando pela pintura até chegar à montagem final onde acontece a montagem da carroceria e do chassi e são feitos os testes de qualidade.

Ele é dividido em módulos, que são responsáveis por gerenciar diferentes partes do processo produtivo, como por exemplo, depósito, fornecedores, qualidade, disponibilização de veículos para serem produzidos.

A empresa Gedas do Brasil era a responsável por prestação de serviços na área de TI e foi vendida em 2006 para a empresa T-Systems que passou a ser um dos líderes em prestação de serviços para o setor automotivo e fortificou sua expertise nos segmentos de manufatura e

telecomunicações (T-SYSTEMS, 2011). Atualmente a empresa é responsável por prestar suporte e sustentação ao sistema de dez fábricas, entre elas estão às fábricas no Brasil e outras em países como a África do Sul, a Argentina, a Espanha, a Índia e a Inglaterra.

As equipes localizadas dentro das fábricas (1º nível) são responsáveis por resolver problemas físicos e técnicos em relação aos aplicativos utilizados nos módulos do sistema, ou qualquer tipo de incidente que não possa ser resolvido remotamente. Os incidentes que envolvam a parte lógica dos servidores, qualidade ou parte de cadeia de processo são escalonados para a equipe de 2º nível que atenderá remotamente.

2.2.1 Ferramenta para alteração de formulários de impressão

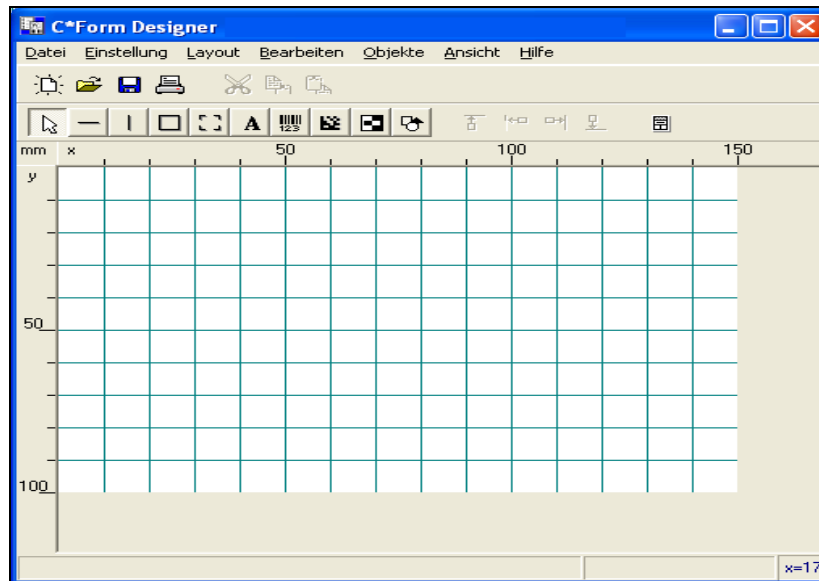
O sistema para planejamento e controle de produção utilizado pela montadora possui diversos módulos que são responsáveis por gerenciar as diferentes partes do processo produtivo. A ferramenta utilizada para alteração de formulários é um desses módulos e ela permite criar, alterar e compilar os formulários utilizados pelas impressoras na linha de produção.

As impressoras utilizam uma linguagem de impressão especial chamada *Intelligent Graphics Printing* (IGP) criada pela empresa Printronix. Segundo DocPath (2011), a linguagem IGP permite “a criação e armazenamento de formulários, geração de logos, códigos de barras, caracteres expandidos, e a criação de outros gráficos.”

Os formulários possuem diversas utilidades, entre elas:

- a) instrução de montagem para os operadores de linha;
- b) controle de qualidade através de Fichas de Exame;
- c) organização e compra do veículo por parte de terceiros;
- d) controle logístico, para noções de armazenamento e controle de fluxo de saída de peças.

Na Figura 1 visualiza-se o *layout* da ferramenta de alteração de formulários.



Fonte: DocPath (2011).

Figura 1 - *Layout* da ferramenta de alteração de formulários

2.3 ABERTURA DE CHAMADO

Quando ocorre uma falha no processo padrão de produção, as fábricas abrem os chamados utilizando um sistema que fica centralizado na matriz da montadora. O nome do sistema é Peregrine ServiceCenter (PSC) que foi desenvolvido pela Peregrine Systems e em 2005 foi vendida para a empresa Hewlett Packard.

Segundo Dolenc (2004), “certificado e premiado pela PinkElephant de acordo com os padrões ITIL, o ServiceCenter é uma aplicação adaptável e robusta que pode ser implementada em módulos para atender às necessidades de cada cliente.”

Desta forma, o 1º nível de cada fábrica possui um grupo próprio que visualiza apenas os chamados relacionados ao seu grupo, o mesmo ocorre com o 2º nível. Em incidentes onde existe a necessidade do 2º nível intervir, o 1º nível abre um chamado e transfere a responsabilidade para o grupo do 2º nível. O tempo de SLA começa a ser contado desde o momento que o chamado é aberto pelo 1º nível.

2.4 SISTEMA ATUAL

Atualmente a equipe de 2º nível da T-Systems do Brasil é responsável por resolver os chamados que envolvam formulários de impressão. A mesma encontra-se no município de Blumenau (SC) e utiliza uma planilha eletrônica do Microsoft Excel para armazenar as informações sobre as alterações.

Foi determinado no contrato com o cliente que todos os chamados que envolvam formulários de impressão serão abertos no Peregrine ServiceCenter como Requisição de Mudança (do inglês *Request For Change* (RFC)) e terão a mesma prioridade obedecendo assim ao SLA e também aos padrões da biblioteca ITIL.

Os chamados de formulários de impressão possuem o seguinte fluxo:

- a) o 1º nível abre um chamado e redireciona para o 2º nível com a solicitação de alteração;
- b) verifica-se se a solicitação é uma alteração, criação ou transferência de um formulário:
 - se for uma alteração ou criação, é feito atendimento;
 - se for uma transferência de um formulário que já foi editado, não há necessidade de alterar o formulário;
- c) o formulário é transferido para o diretório de testes, e o *status* do chamado é alterado para “esperando uma resposta”:
 - se a alteração ou criação não for aprovada, são feitas as devidas correções e o formulário é colocado novamente no diretório de testes;
- d) após o 1º nível realizar os testes de impressão e o cliente aprovar e assinar o documento de aceite, o formulário é transferido para o diretório produtivo;
- e) o chamado é devolvido para o 1º nível (grupo de origem).

Na Figura 2 verifica-se o fluxo atual de uma solicitação de alteração de formulário.

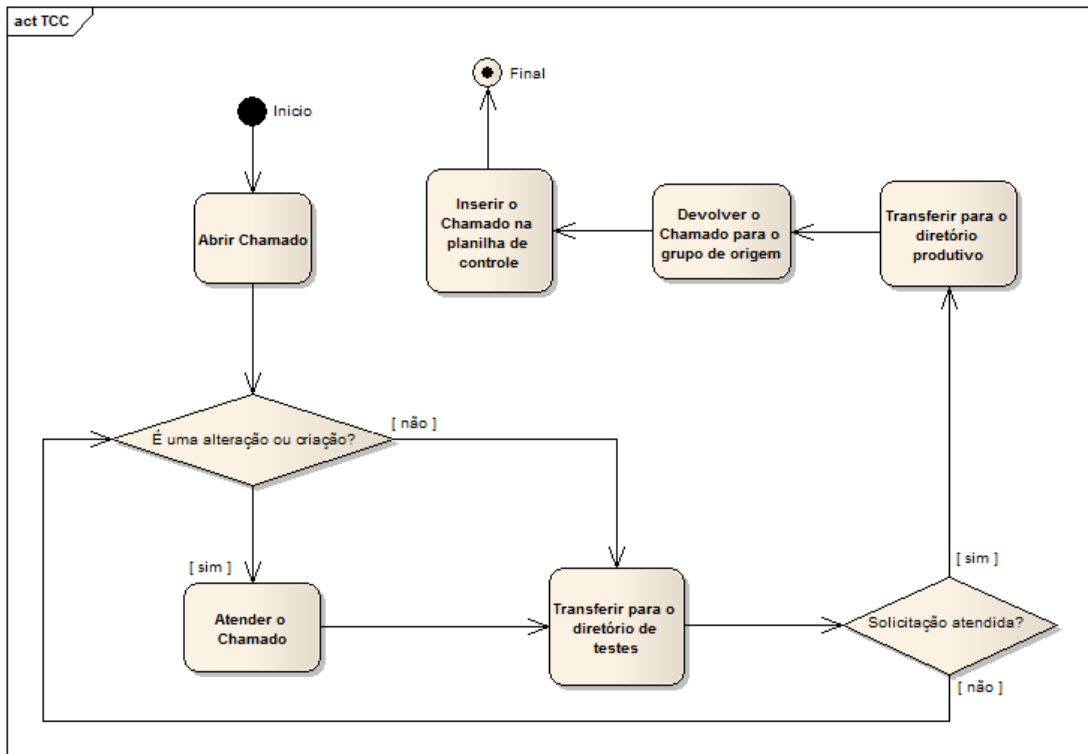


Figura 2 - Fluxo sistema Atual

Depois que o chamado é devolvido para o grupo de origem, as informações são inseridas na planilha de controle. A Figura 3 demonstra o controle atual das alterações utilizando a planilha.

	A	B	C	D	E	F	G	H
1	Formulário	ticket	tipo	Alteração	Tempo	Data		
2	abc123	13576327	Alteração	Criação de um formulário para o verso da Impressão	02:00	31/01/11		
3	abcdelfg	14286634	Transferência	Transferência do anw para o diretório produtivo	00:05	30/03/11		
4	teste	14590175	Transferência	Transferência do anw para o diretório produtivo	00:05	04/05/11		
5	teste	14590175	Transferência	Transferência do anw para o diretório produtivo	00:05	04/05/11		
6	abc123	14996387	Alteração	Inserido clausula no IF_BLB_FOX de que se for fox (5Z) e tiver PR da familia AAU igual a E25 deve imprimir FOX ROCK IN RIO	00:25	10/06/11		
7	abc123	14996387	Alteração	Inserido clausula no IF_EBFOX de que se for fox (5Z) e tiver PR da familia AAU igual a E25 deve imprimir FOX ROCK IN RIO	00:25	10/06/11		
8	abc123	14996387	Transferência	Transferência do anw para o diretório teste .	00:10	10/06/11		
9	abc123	14996387	Transferência	Transferência do anw para o diretório teste .	00:10	10/06/11		
10	abc123	14996387	Alteração	Foi alterado o local do IF dentro do arquivo	00:20	17/06/11		
11	abc123	14996387	Transferência	Transferência do anw para o diretório teste	00:10	17/06/11		
12	abc123	14996387	Alteração	A condição foi retirada do IF_BLB_FOX e adicionada no IF_TMA	00:20	18/06/11		
13	abc123	14996387	Transferência	Transferência do anw para o diretório teste	00:10	18/06/11		
14	lqps	15674589	Alteração	Inclusão de código de barras com informações diferentes caso o TMA seja 5Z7	00:30	18/08/11		
15	qssss	15664595	Transferência	Transferência do anw para o diretório produtivo	00:10	18/08/11		
16	pfecde	15770290	Alteração	Foi incluido no if a condição para SpaceFox chamando o formulário lqpsf.anw	00:20	05/09/11		
17	pfecde	15770290	Criação	Foi criado formulário com os mesmo campos do formulário lqpsf.anw	00:20	05/09/11		
18								
19								
20								
21								
22								
23								
24								
25								
26								
27								
28								
29								
30								
31								
32								
33								
34								
35								
36								
37								
38								
39								
40								
41								
42								
43								

Figura 3 - Controle de Alterações em Formulários

A planilha possui abas, uma para cada fábrica e as informações inseridas para cada chamado são:

- a) nome do formulário;
- b) número do chamado aberto pelo 1º nível;
- c) tipo de alteração;
- d) descrição da ação que foi realizada;
- e) tempo estimado gasto para realizar o evento;
- f) data que a ação foi realizada.

São várias as pessoas responsáveis por resolver os chamados e muitas vezes, em trocas de turno, não se sabe quais ações já foram tomadas, pois não existe um lugar onde essas informações estão armazenadas. O fato de existir vários responsáveis traz outro problema, pois as alterações na planilha podem ser feitas por qualquer pessoa sem que a mesma precise se identificar.

2.5 TRABALHOS CORRELATOS

Pode-se citar como trabalhos correlatos as monografias realizadas por Karin Schoenfelder e Clóvis Diego Schuldt para o Trabalho de Conclusão de Curso (TCC) do curso de Sistemas de Informação na Universidade Regional de Blumenau (FURB).

O TCC de Schoenfelder (2010) descreve um sistema de gerência de mudanças aplicado a empresa Centrium Informática, baseado nas melhores práticas sugeridas pela biblioteca ITIL, utilizando como ferramenta de desenvolvimento o aplicativo Genexus e o banco de dados Microsoft SQL Server 2005. Embora o sistema desenvolvido não tenha ficado totalmente aderente ao processo de Gerenciamento de Mudanças da ITIL por não estar integrado com outros processos, ele atende aos objetivos de uma organização que queira iniciar um processo controlado das mudanças efetuadas. Na Figura 4 pode-se verificar a tela para cadastro de solicitação de mudança do sistema.

Solicitação de mudança									
Número	25								
Título	Atualização do sistema								
Data solicitação	20/06/10								
Descrição	Atualização do sistema para a última versão								
Motivo	Constantes travamentos na emissão de NF-e, comunicação com a SEFAZ								
Efeito da não implementação	Sistema lento para emitir nota, reclamações por parte do faturamento								
Data da proposta	21/06/10								
Prioridade	Urgente Impacto no ambiente Alta								
Observações	O problema está ocorrendo desde a última atualização do módulo								
Tipo item de configuração	Software								
	<table border="1"> <thead> <tr> <th>Cód. Item</th> <th>Descrição</th> </tr> </thead> <tbody> <tr> <td>2</td> <td>NF-e</td> </tr> <tr> <td>10</td> <td></td> </tr> <tr> <td>10</td> <td></td> </tr> </tbody> </table>	Cód. Item	Descrição	2	NF-e	10		10	
Cód. Item	Descrição								
2	NF-e								
10									
10									
<input type="button" value="Salvar"/> <input type="button" value="Fechar"/>									

Fonte: Schoenfelder (2010).

Figura 4 - Tela para cadastro de solicitação de mudança

O TCC de Schuldt (2010) descreve um sistema para facilitar a requisição, armazenamento e consulta das mudanças aplicadas no ambiente de diversos sistemas de informação, bancos de dados e infra-estrutura em geral para diminuir o impacto das alterações nos serviços. As tecnologias utilizadas foram JSP e *Javascript* utilizando banco de dados Oracle. O resultado do trabalho foi uma melhora no âmbito gerencial do ambiente de TI, proporcionando uma maior confiabilidade, integridade dos dados e segurança ao mesmo pelo uso do banco de dados Oracle. Paralelamente obteve-se uma maior produtividade do técnico devido à menor alocação na documentação das mudanças, facilitando a recuperação dos dados já cadastrados. Na Figura 5 apresenta-se a tela de cadastro de nova mudança.

Gerência de mudanças Tecnológica	
Bem vindo, Clóvis Diego Schuldt. Sair	
<input type="button" value="Cadastrar nova mudança"/>	<input type="button" value="Pesquisar mudança"/>
<input type="button" value="Pendentes de avaliação"/>	<input type="button" value="Pendentes de liberação"/>
<input type="button" value="Controle de usuários"/>	<input type="button" value="Cadastros"/>
Tipo Mudança:	Normal
Decorrencia da Mudança:	Melhoria
Tipo de configuração:	Servidor
Item de configuração:	secs001
Impacto:	Alto
Prioridade:	Baixa
Motivo da Mudança:	Atualizar OpenLDAP devido a melhorias de controle de acessos na nova versão.
Ambiente da Mudança:	Produção
Fluxo da Liberação:	1 - Realizar backup da base do LDAP; 2 - Baixar serviço do LDAP; 3 - Atualizar LDAP para nova versão; 4 - Iniciar serviço do LDAP.
Restrições da Liberação:	Necessário realizar fora do horário comercial.
Plano retorno:	1 - Parar serviço do LDAP; 2 - Instalar versão anterior do LDAP; 3 - Iniciar serviço do LDAP; 4 - Validar base de dados.
<input type="button" value="Inserir"/> <input type="button" value="Desfazer"/>	

Fonte: Schuldt (2010).

Figura 5 - Tela de cadastro de nova mudança

Existem *softwares* no mercado voltados para o controle de eventos e chamados, entre eles pode-se citar o 0800net, desenvolvido inicialmente pela HBsis Informática Ltda em 1988. A partir do ano de 2006, foi criada a empresa Ellevo para dar continuidade ao sistema. O software gerencia e controla, de forma centralizada, todos os tipos de problemas e pedidos de suporte de usuários e possibilita a abertura de chamados pelo cliente ou por e-mail, controle de SLA, Inventário, Controle de Projetos, Gestão de Prospect, extração de dados para gerenciamento de métricas/índices de controle (ELLEVO, 2011). Na Figura 6, visualiza-se a tela de cadastro de equipamento.

The screenshot shows a web-based interface for equipment registration. The top navigation bar includes 'Logoff' and a '0800NET' logo. The main content area is titled 'Equipamentos' and features a tabbed interface with 'Equipamentos', 'Softwares', and 'Relatórios'. The 'Equipamentos' tab is active, displaying a form for registering a computer. The form includes the following fields and values:

- Etiqueta:** AB2020
- Número de série:** 908908978
- Cadastro:** 10/05/2006
- Auditoria:** (empty)
- Nome:** COMPUTADOR PENTIUM IV
- Compra:** 10/5/2006
- Instalação:** 10/5/2006
- Descrição:** (empty)
- Garantia:** 10/5/2008
- Centro Custo:** (empty)
- Categoria:** Computadores
- Custo Unit.:** 0,00
- Custo Total:** 0,00
- Localização:** Predio 1 - SALA 1
- Fabricante:** DELL
- Marca:** CENTRINO
- Modelo:** A4000
- Fornecedor:** DELL
- Departamento:** RH
- Projeto:** ==Projetos==
- Status:** Ativo

Additional features include a 'RepresFim' field, a 'Usuários relacionados' list with 'Amélia Flores' selected, and 'Confirmar' and 'Excluir' buttons at the bottom.

Fonte: ELLEVO (2011).

Figura 6 - Tela para cadastro de equipamento

3 DESENVOLVIMENTO DO APLICATIVO

Neste capítulo são apresentadas as características do aplicativo desenvolvido, através de fluxogramas, especificação de requisitos funcionais e não funcionais, o diagrama de casos de uso e do Modelo de Entidade Relacional (MER). São descritas também as técnicas e ferramentas utilizadas no processo de implementação, a operacionalidade do aplicativo e os resultados obtidos.

3.1 LEVANTAMENTO DE INFORMAÇÕES

A solução desenvolvida surgiu através da necessidade de melhoria no processo de controle e histórico de alterações que envolvam formulários de impressão. Ela consiste num aplicativo *desktop* que possibilite controlar as alterações que foram e estão sendo realizadas. O controle acontece através da inserção de eventos no aplicativo, junto à execução das atividades necessárias na alteração de um formulário de impressão, garantindo que os envolvidos na atividade sempre tenham informações atualizadas, ajudando a diminuir os problemas de comunicação entre membros da equipe.

Optou-se por priorizar a simplicidade para que o usuário não perca muito tempo ao inserir ou visualizar as informações no aplicativo, fazendo com que seja possível antecipar ações e agir pró ativamente através do acompanhamento dos chamados pendentes, sem a necessidade de despender muito tempo nestas atividades.

Como os chamados já são abertos obedecendo a critérios da biblioteca ITIL, é possível verificar o tempo de cada evento e também o tempo total de um chamado cadastrado no aplicativo, facilitando o cumprimento do contrato (SLA).

Na Figura 7 é possível visualizar o novo fluxo utilizado durante a alteração de um formulário.

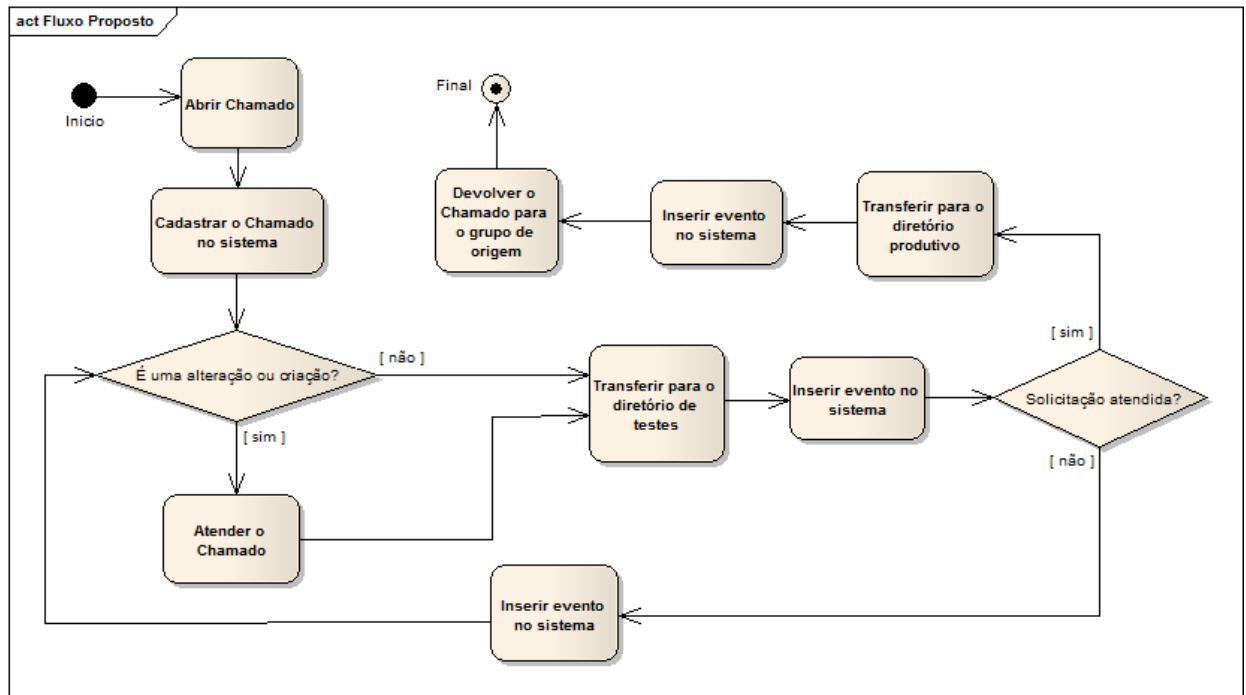


Figura 7 - Fluxo proposto ao utilizar o aplicativo

3.2 ESPECIFICAÇÃO

Nesta seção serão apresentados os principais requisitos funcionais (RF), requisitos não funcionais (RNF), sua rastreabilidade com casos de uso e o modelo entidade relacionamento (MER) do aplicativo. A descrição dos casos de uso pode ser visualizada no Apêndice A.

3.2.1 Requisitos Funcionais

No Quadro 1 são apresentados os requisitos funcionais do aplicativo.

Requisitos Funcionais	Caso de Uso
RF01: O aplicativo deverá permitir efetuar <i>login</i> por perfil de usuário.	UC01
RF02: O aplicativo deverá permitir alterar a senha do <i>login</i> .	UC02
RF03: O aplicativo deverá permitir cadastrar chamados.	UC03
RF04: O aplicativo deverá permitir cadastrar eventos.	UC04

RF05: O aplicativo deverá permitir o administrador cadastrar novos usuários.	UC05
RF06: O aplicativo deverá permitir o administrador cadastrar novas fábricas.	UC06
RF07: O aplicativo deverá permitir o administrador cadastrar novos tipos de evento.	UC07
RF08: O aplicativo deverá permitir gerar histórico de chamado.	UC08
RF09: O aplicativo deverá permitir gerar pesquisa por formulário.	UC09
RF10: O aplicativo deverá permitir gerar pesquisa por usuário.	UC10
RF11: O aplicativo deverá permitir gerar pesquisa por evento.	UC11

Quadro 1 - Requisitos funcionais

3.2.2 Requisitos não funcionais

O Quadro 2 lista os requisitos não funcionais previstos para o aplicativo.

Requisitos Não Funcionais
RNF01: O aplicativo deverá utilizar banco de dados MySQL versão 5.1.
RNF02: O aplicativo será implementado na linguagem Java versão 6.
RNF03: O aplicativo deverá ser executado no sistema operacional Windows XP e 7.
RNF04: O aplicativo deve ser desenvolvido para ambiente <i>desktop</i> .

Quadro 2 - Requisitos não funcionais

3.2.3 Casos de Uso

Na Figura 8 pode-se verificar o caso de uso para controle de acesso. Os atores envolvidos neste caso de uso são o usuário e o administrador do aplicativo, onde o administrador pode cadastrar novos usuários além de utilizar as funções do usuário normal.

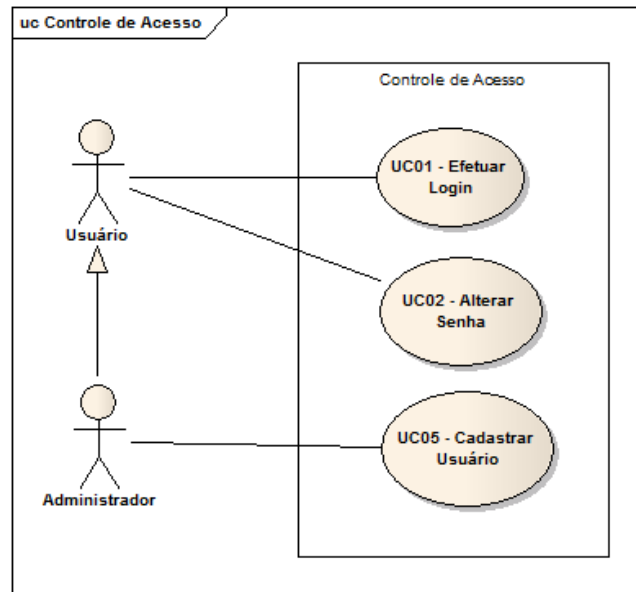


Figura 8 - Caso de Uso - Controle de Acesso

Na Figura 9 verifica-se o caso de uso para controle de eventos. Neste caso de uso o ator usuário pode manter os chamados e os eventos, o administrador pode realizar as mesmas atividades do usuário e ainda cadastrar novas fábricas e novos tipos de evento.

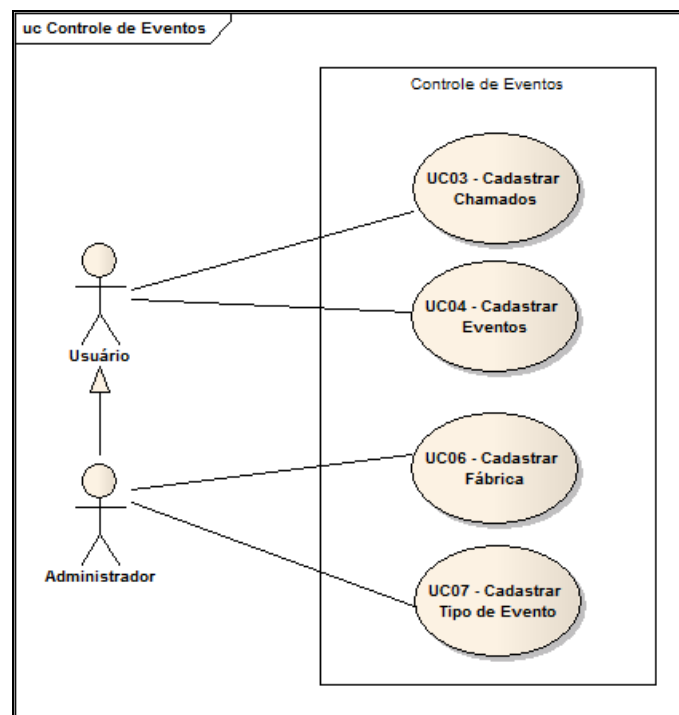


Figura 9 - Caso de Uso - Controle de Eventos

A Figura 10 mostra o caso de uso para emissão de relatórios, neste caso de uso ambos os atores (usuário e administrador) podem realizar todas as atividades.

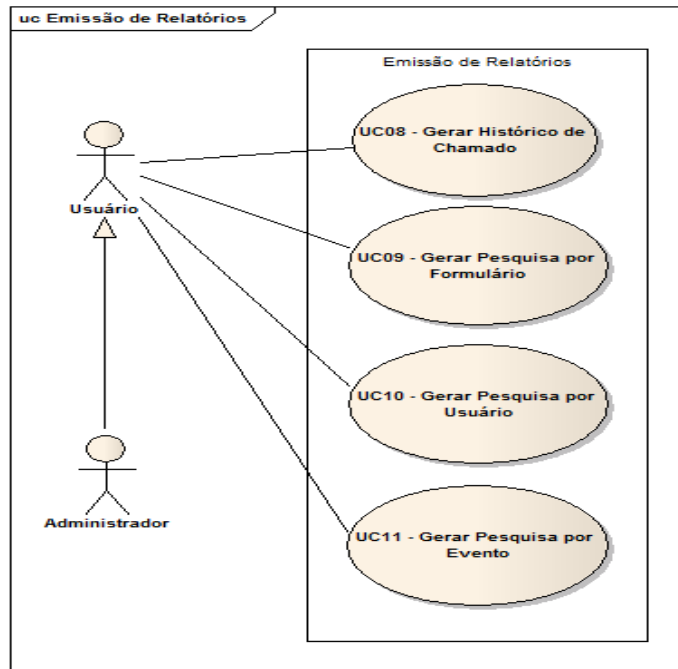


Figura 10 - Caso de Uso - Emissão de Relatórios

3.2.4 Modelo Entidade Relacionamento

Na Figura 11 verifica-se o Modelo Entidade Relacionamento (MER) do aplicativo desenvolvido com as entidades criadas e seus relacionamentos. Não foram utilizadas *foreign-keys*, pois o controle é feito pelo aplicativo através do campo *ticket* que representa o número do chamado aberto no sistema PSC. No Apêndice B pode-se visualizar o dicionário de dados.

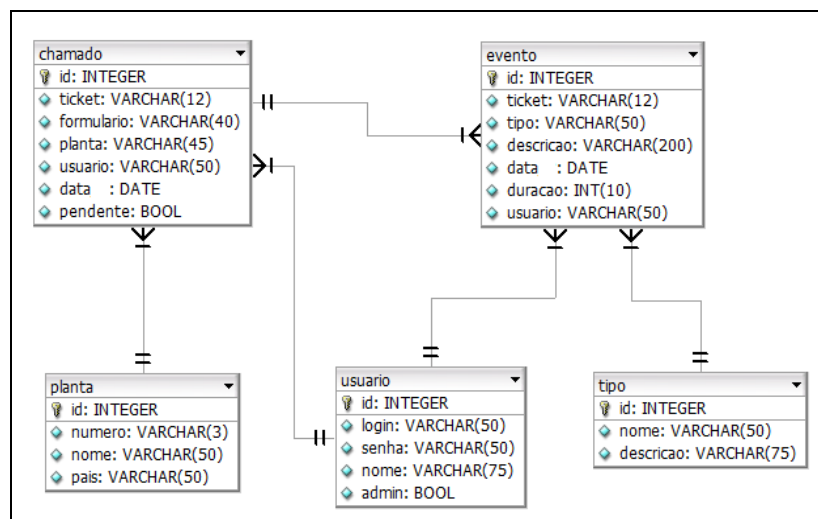


Figura 11 - Modelo Entidade/Relacionamento

3.3 IMPLEMENTAÇÃO

A seguir são mostradas as técnicas e ferramentas utilizadas e a operacionalidade da implementação.

3.3.1 Técnicas e ferramentas utilizadas

O aplicativo foi desenvolvido utilizando Java que é uma linguagem de programação de alto nível, orientada a objetos que foi desenvolvida em 1990, pela SUN Microsystems. (CASTELA, 2010)

Para implementação do aplicativo foi utilizada a ferramenta NetBeans IDE, que permite a criação de códigos Java e facilita a criação de interfaces gráficas de um modo visual, pois apresenta todos os componentes *Swing* em uma paleta, bastando arrastá-los para interface e configurar suas propriedades e eventos da forma desejada.

Para o armazenamento de dados foi utilizado o gerenciador de banco de dados MySQL. Os testes de conformidade do código no aplicativo desenvolvido com as operações no banco foram realizados com auxílio da ferramenta MySQL Query Browser.

Os relatórios do aplicativo são gerados utilizando uma biblioteca Java chamada JasperReports. Nesta biblioteca a definição é feita em XML e pode ser editada manualmente. Contudo é normalmente usada a ferramenta iReport, que é um editor gráfico para o arquivo XML. O arquivo XML é depois compilado, num arquivo com extensão “.jasper” que tem a mesma função de um arquivo “.class”. Este arquivo é depois executado contra uma fonte de dados, que na maioria dos casos é um banco de dados. Existem várias fontes de dados e a biblioteca oferece mecanismos para criação de fontes compatíveis (JAVAFREE, 2011).

Na Figura 12 visualiza-se a tela do iReport onde um relatório está sendo elaborado. A ferramenta possui uma palheta com os campos que podem ser utilizados, como também menus para inserção de *queries* de pesquisa na base de dados.

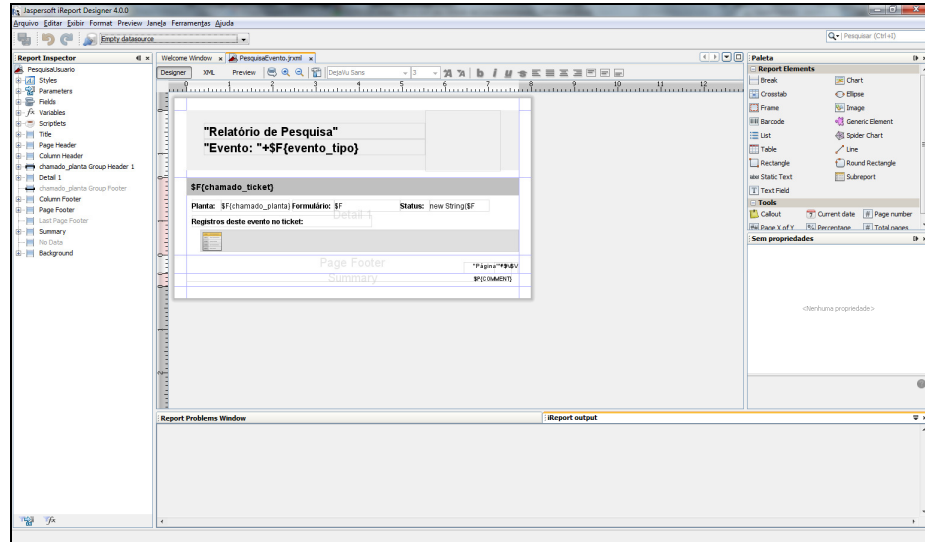


Figura 12 - Criando relatório no iReport

3.3.2 Operacionalidade da implementação

Nesta sub-seção são apresentadas as telas do aplicativo e trechos de código relevantes. O aplicativo possui dois perfis de usuário, sendo que a diferença básica entre suas permissões é o fato de um poder realizar cadastros e o outro não. Será utilizado o perfil do administrador para demonstrar todas as funcionalidades do aplicativo.

A operacionalidade do aplicativo é inicialmente apresentada pela tela de *login*, onde o usuário deve preencher o campo de usuário e senha, como é apresentado na Figura 13.

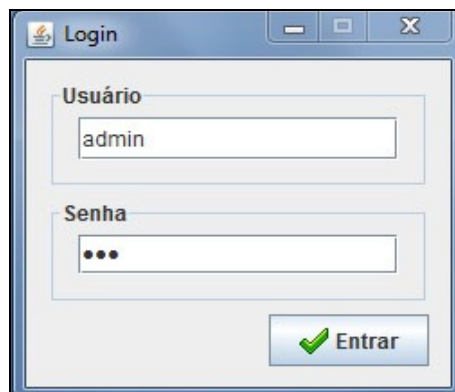


Figura 13 - Tela de *login*

Na primeira utilização do aplicativo é sugerido ao usuário que altere a senha, já que todos os usuários ao serem cadastrados recebem uma senha padrão. Se os campos possuírem

informações incorretas ou estiverem em branco, uma mensagem será mostrada informando que o usuário deve preencher corretamente os campos. Após realizar o *login*, o usuário é redirecionado para tela principal, sendo que esta tela pode ser visualizada na Figura 14.

Planta	Número Chamado	Formulário	Usuário	Data Abertura	Última Atualização
001 - Blumenau	12345	abc123	admin	20.11.2011	20.11.2011
015 - São Paulo	54321	teste	admin	18.11.2011	18.11.2011
001 - Blumenau	98765	pl1555	admin	15.11.2011	15.11.2011
018 - Porto Alegre	77766	bnu765	admin	20.11.2011	20.11.2011

Total: 4

Figura 14 - Tela principal do aplicativo

A Figura 13 possui na parte superior dois menus, um para cadastro de novos dados (quando for um usuário administrador), alteração de senha e *logout* do aplicativo e o outro serve para realizar pesquisas por chamado, formulário, usuário ou evento. O conteúdo destes menus será explicado de uma forma mais detalhada posteriormente. Ainda na parte superior existe o botão para cadastrar novos chamados e uma área informando qual o usuário está conectado. O resto da tela é ocupado por uma tabela com os chamados pendentes. Nesta tabela as linhas podem possuir três cores:

- branca: o chamado está devidamente atualizado;
- amarela: o chamado está a dois dias sem ser atualizado;
- vermelha: o chamado está a três dias ou mais sem ser atualizado.

A última atualização refere-se à data do último evento que foi inserido para este chamado, se existir algum chamado com mais de três dias sem atualização, ao *logar* no aplicativo uma mensagem aparece antes da tela inicial ser exibida, conforme verifica-se na Figura 15.

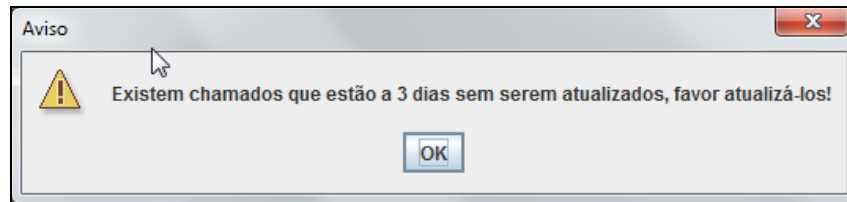


Figura 15 - Tela de aviso de chamados desatualizados

Para cadastrar um novo chamado, deve-se clicar no botão “cadastrar chamado” que aparece na Figura 16. Ao clicar neste botão a tela da Figura 15 será exibida.

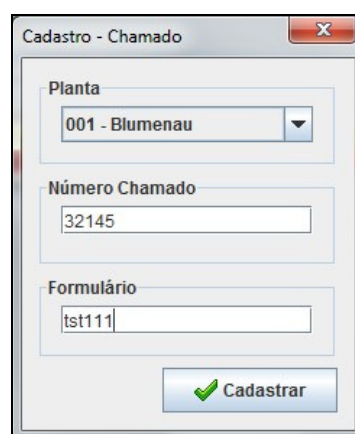


Figura 16 - Tela para cadastro de chamado

Se todos os campos estiverem preenchidos e o número do chamado não existir na base de dados, o usuário será redirecionado para tela de inserção de eventos onde deve inserir pelo menos um evento ao chamado. Não é possível sair desta tela antes de inserir pelo menos um evento, conforme pode-se verificar nas Figura 17 e Figura 18.

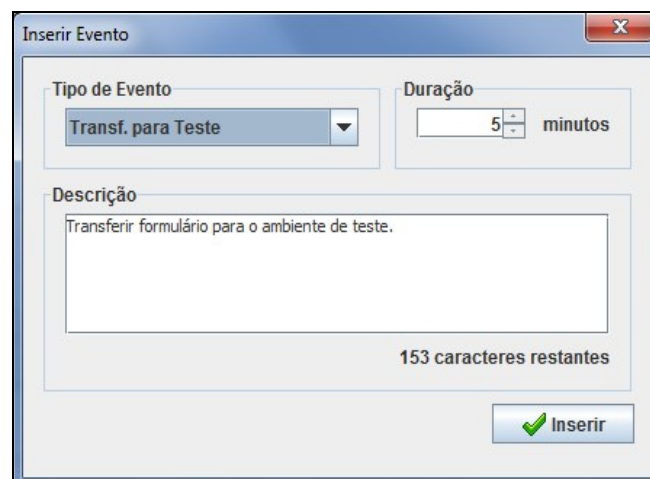


Figura 17 - Tela para inserir eventos ao chamado

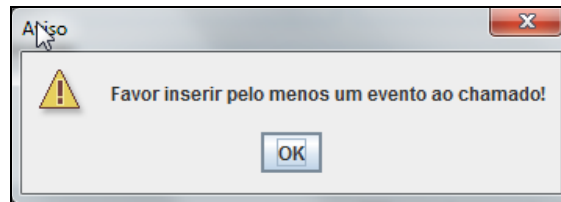


Figura 18 - Tela de aviso para inserir um evento

Após inserir um evento ao chamado, a tela de chamado selecionado será exibida, conforme Figura 19.

Data	Tipo	Descrição	Usuário	Dur.
20.11.2011	Alterar Campo	Foi alterado o campo descrição do veículo.	admin	15
20.11.2011	Transf. para Teste	Transferir formulário para o ambiente de teste.	admin	5

Tempo total: 20 minutos

Figura 19 - Tela chamado selecionado

A Figura 19 possui uma tabela com os eventos que já foram inseridos neste chamado. Na parte inferior da tela existem campos que são preenchidos quando uma linha da tabela for selecionada e servem para proporcionar uma melhor visualização dos dados. Ao clicar no botão “inserir evento” a tela da Figura 17 é exibida e ao clicar no botão “voltar o usuário”, volta para a tela principal (Figura 14).

Existem duas maneiras de finalizar um chamado:

- a) inserir um evento do tipo “Transf. para Produtivo”: ao inserir este evento pode-se concluir que a alteração já foi executada e finalizada, pois houve a aprovação do cliente para colocar o formulário em ambiente produtivo;
- b) inserir um evento do tipo “Devolver Ticket”: este evento será utilizado quando o chamado estiver a muito tempo (1 mês ou mais) sem uma resposta do cliente. Neste caso o chamado é devolvido para o grupo de origem e quando houver uma resposta ele será reaberto.

Como visto anteriormente, ao cadastrar um chamado é feita a verificação do número do chamado para não ocorrer duplicidade. No caso do *ticket* ter sido finalizado com o evento “Devolver Ticket” ao tentar inseri-lo novamente, o aplicativo avisará o usuário que este chamado foi finalizado sem o formulário ter sido colocado em ambiente produtivo e irá perguntar se deseja abri-lo novamente, como mostra a Figura 20.

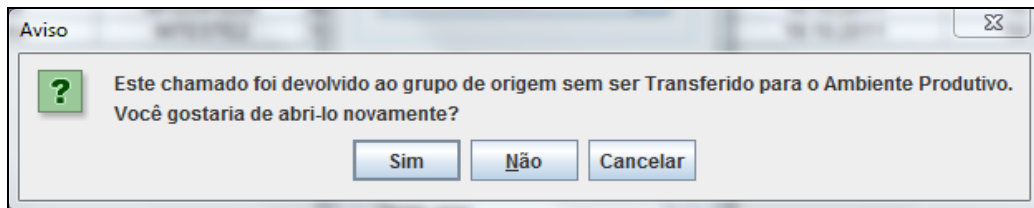


Figura 20 - Tela de reabertura de chamado

Se o usuário optar por reabrir o chamado, ele volta a ter o status de chamado pendente e pode ser visualizado na tabela da tela principal.

Na Figura 21 é possível visualizar o conteúdo do *menu* de configurações do aplicativo.

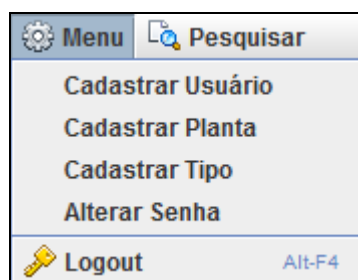


Figura 21 - Menu de configuração do aplicativo

As três primeiras opções, que são relativas a cadastro estão disponíveis apenas para os usuários que tem permissão de administrador. A Figura 22 mostra a tela de cadastro de usuário. Para cadastrar um usuário é necessário informar o nome, o *login* e o nível de permissão. Ao clicar em cadastrar uma mensagem de cadastro efetuado com sucesso é exibida junto com a senha padrão gerada para o usuário.

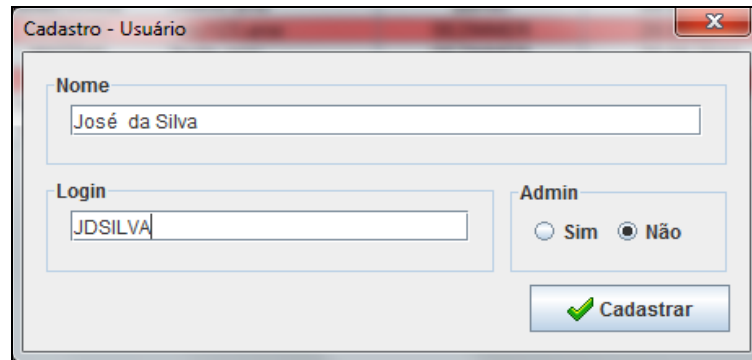


Figura 22 - Tela cadastro de usuário

A Figura 23 apresenta a tela de cadastro de planta de fábricas, que também é acessada através deste *menu*. Neste cadastro o administrador precisa preencher os campos de código da planta, nome e país onde ela se localiza. O código da planta é baseado nos códigos que a própria montadora utiliza para distinguir as fábricas.

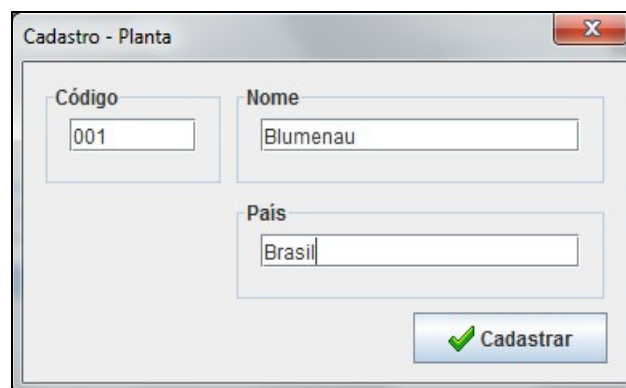
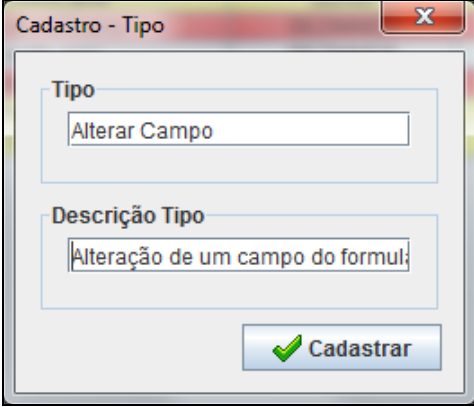


Figura 23 - Tela cadastro de planta

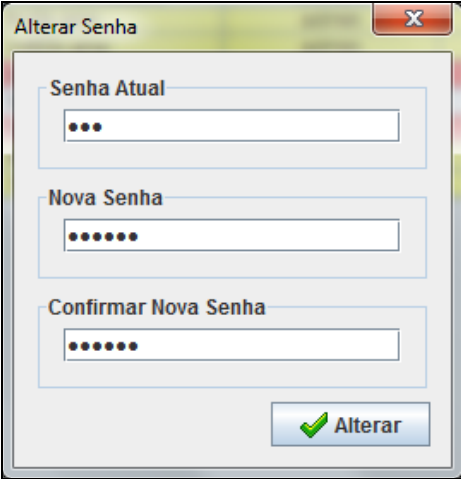
A Figura 24 mostra a tela de cadastro de tipo de evento. Para se cadastrar um tipo de evento é necessário informar qual será o evento, sendo que este nome irá aparecer no *combo box* da tela inserir evento e da tela pesquisar evento. O mesmo deve ser um nome curto, mas de fácil entendimento. O outro campo que precisa ser preenchido é a descrição, neste caso pode-se explicar de uma maneira mais completa a finalidade daquele tipo de evento.



The image shows a dialog box titled "Cadastro - Tipo". It has a close button (X) in the top right corner. Inside the dialog, there are two text input fields. The first field is labeled "Tipo" and contains the text "Alterar Campo". The second field is labeled "Descrição Tipo" and contains the text "Alteração de um campo do formul". At the bottom right of the dialog, there is a button labeled "Cadastrar" with a green checkmark icon.

Figura 24 - Tela cadastrar tipo de evento

A Figura 25 mostra a tela de alteração de senha. Esta tela é comum para os dois tipos de usuário. Para alterar a senha é necessário informar a senha atual do usuário e a nova senha que deseja ser colocada. Para garantir que não houve erro de digitação a nova senha deverá ser informada no campo de confirmação de senha. Caso a senha atual ou os campos de nova senha e confirmação de nova senha não estiverem com as informações corretas, uma mensagem de erro será exibida.



The image shows a dialog box titled "Alterar Senha". It has a close button (X) in the top right corner. Inside the dialog, there are three text input fields. The first field is labeled "Senha Atual" and contains three dots. The second field is labeled "Nova Senha" and contains six dots. The third field is labeled "Confirmar Nova Senha" and contains six dots. At the bottom right of the dialog, there is a button labeled "Alterar" with a green checkmark icon.

Figura 25 - Tela alterar senha

A última opção deste *menu* é o *logout*, sendo que o usuário ao clicar nesta opção ou utilizar o atalho "Alt + F4", o mesmo será desconectado do aplicativo e a tela de *login* será exibida. Já na Figura 26 pode-se visualizar o *menu* de pesquisa.

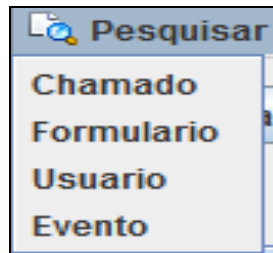


Figura 26 - Menu de pesquisa

A Figura 27 mostra a tela de pesquisa por chamado. Nesta tela o usuário informa o número do chamado e escolhe a opção de visualizar a pesquisa ou de gerar a pesquisa no formato de PDF.

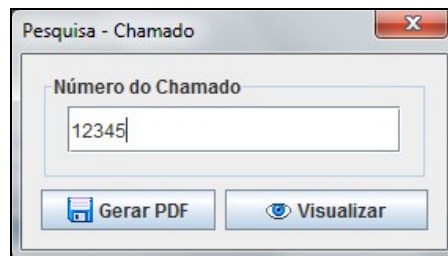


Figura 27 - Tela pesquisar chamado

A Figura 28 demonstra o *layout* de uma pesquisa por chamado quando a opção visualizar é escolhida.

 A screenshot of a report window titled 'Relatório de Pesquisa Ticket - 12345'. The report displays the following information:

- Planta: 001 - Blumenau
- Data de abertura: 20/11/2011
- Status: Pendente
- Usuário: admin
- Formulário: abc123
- Tempo total: 20 minutos

 Below this information is a table titled 'Eventos' with the following data:

Data	Tipo	Descrição	Usuário	Duração
20/11/2011	Alterar Campo	Foi alterado o campo descrição do veículo.	admin	15 min
20/11/2011	Transf. para Teste	Transferir formulário para o ambiente de teste.	admin	5 min

Figura 28 - Layout de uma pesquisa por chamado

A Figura 29 mostra a mensagem exibida ao escolher a opção gerar PDF.

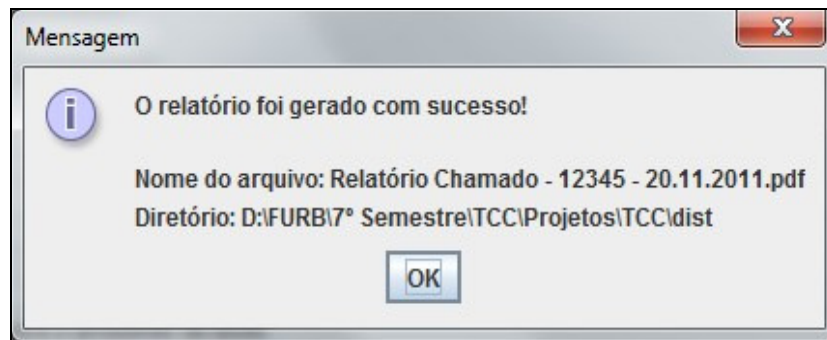


Figura 29 - Tela de relatório gerado em PDF

Na Figura 30 visualiza-se o relatório gerado em PDF.

Relatório de Pesquisa				
Ticket - 12345				
Planta: 001 - Blumenau	Data de abertura: 20/11/2011	Status: Pendente		
Usuário: admin	Formulário: abc123	Tempo total: 35 minutos		
Eventos				
Data	Tipo	Descrição	Usuário	Duração
20/11/2011	Alterar Campo	Foi alterado o campo descrição do veículo.	admin	15 min
20/11/2011	Transf. para Teste	Transferir formulário para o ambiente de teste.	admin	5 min
27/11/2011	Alterar Campo	Foi necessário corrigir o tamanho do texto na descrição.	admin	10 min
27/11/2011	Transf. para Teste	Transferir formulário para o ambiente de teste.	admin	5 min

Página 1 de 1

Figura 30 - Relatório gerado em PDF

Na Figura 31 pode-se visualizar a tela de pesquisa por formulário. Nesta pesquisa é necessário informar o nome do formulário e a quantidade de dias que deve ser utilizada para realizar a pesquisa. Existem cinco opções:

- a) 7 dias;
- b) 15 dias;

- c) 30 dias;
- d) 60 dias;
- e) Todos os registros relacionados aquele formulário.

Figura 31 - Tela pesquisa por formulário

Na Figura 32 está o *layout* de uma pesquisa por formulário ao clicar no botão visualizar. Se o usuário optar por gerar a pesquisa em PDF, uma tela com o nome e o diretório será exibida.

**Relatório de Pesquisa
Formulário - teste**

Planta: 015 - São Paulo Data: 18/11/2011 Status: Pendente
Ticket: 54321 Usuário: admin

Eventos			
Data	Tipo	Usuario	Duração
18/11/2011	Transf. para Teste	admin	5 min

Planta: 015 - São Paulo Data: 20/11/2011 Status: Finalizado
Ticket: 55522 Usuário: admin

Eventos			
Data	Tipo	Usuario	Duração
20/11/2011	Remover Campo	admin	10 min
20/11/2011	Transf. para Teste	admin	5 min
20/11/2011	Contato	admin	10 min
20/11/2011	Transf. para Produtivo	admin	5 min

* Registro(s) dos últimos 7 dias

Figura 32 - *Layout* de uma pesquisa por formulário

No final da pesquisa existe uma observação onde são mostrados quantos dias foram pesquisados na base de dados.

A Figura 33 mostra a tela de pesquisa por usuário. Para realizar esta pesquisa o usuário deve escolher um usuário, a fábrica e a quantidade de dias que a pesquisa deve buscar.

Figura 33 - Tela pesquisa por usuário

Na Figura 34 visualiza-se o *layout* da pesquisa ao clicar no botão visualizar. Ao clicar no botão gerar PDF uma tela com o nome e o diretório será exibida.

Relatório de Pesquisa				
Usuário - admin				
001 - Blumenau				
Ticket:	Data:	Formulário:	Status:	Tempo Total:
1) 98765	15/11/2011	pll555	Pendente	20 min
2) 12345	20/11/2011	abc123	Pendente	20 min
015 - São Paulo				
Ticket:	Data:	Formulário:	Status:	Tempo Total:
1) 54321	18/11/2011	teste	Pendente	5 min
2) 55522	20/11/2011	teste	Finalizado	30 min
018 - Porto Alegre				
Ticket:	Data:	Formulário:	Status:	Tempo Total:
1) 99912	20/11/2011	1231st	Finalizado	25 min
2) 77766	20/11/2011	bnu765	Pendente	5 min

* Esta pesquisa não utilizou nenhum filtro de tempo

Figura 34 - *Layout* de uma pesquisa por usuário

Na Figura 35 pode-se visualizar a tela de pesquisa por evento. Nesta pesquisa o usuário deve escolher um tipo de evento, a fábrica e a quantidade de dias que a pesquisa deve buscar.

Figura 35 - Tela pesquisa por evento

Na Figura 36 visualiza-se o *layout* da pesquisa ao clicar no botão visualizar. Ao clicar no botão gerar PDF uma tela com o nome e o diretório será exibida.

Relatório de Pesquisa						
Evento: Transf. para Teste						
98765						
Planta:	001 - Blumenau	Formulário:	pll555	Status:	Pendente	
Registros deste evento no ticket:						
Data:	15/11/2011	Usuário:	admin	Duração:	5 min	
Descrição: Transferir formulário para o ambiente de teste.						
54321						
Planta:	015 - São Paulo	Formulário:	teste	Status:	Pendente	
Registros deste evento no ticket:						
Data:	18/11/2011	Usuário:	admin	Duração:	5 min	
Descrição: Transferir formulário para o ambiente de teste.						
77766						
Planta:	018 - Porto Alegre	Formulário:	bnu765	Status:	Pendente	
Registros deste evento no ticket:						

Figura 36 - *Layout* de uma pesquisa por evento

Na Figura 37 pode-se visualizar o método utilizado para visualizar uma pesquisa utilizando a biblioteca JasperReports.

```

public JFrame visualiza(String usuario, String planta, String data, String comment) {
    InputStream input = getClass().getResourceAsStream("/PesquisaUsuario.jasper");
    Map<String, Object> parametros = new HashMap<String, Object>();
    Connection con = Conexao.getConexao();

    String condicao = planta+" "+data;

    parametros.put("USUARIO", usuario);
    parametros.put("CONDICAO", condicao);
    parametros.put("COMMENT", comment);

    try {
        JasperPrint print = JasperFillManager.fillReport(input, parametros, con);
        JRViewer viewer = new JRViewer(print);
        JFrame frame = new JFrame("Pesquisa Usuário");
        frame.add(viewer, BorderLayout.CENTER);
        frame.setSize(800, 600);
        frame.setExtendedState(JFrame.MAXIMIZED_BOTH);
        frame.setDefaultCloseOperation(JFrame.DISPOSE_ON_CLOSE);
        frame.setVisible(true);
        return frame;
    } catch (JRException ex) {
        System.out.println(ex.getMessage());
    }

    return null;
}

```

Figura 37 - Método para gerar visualização da pesquisa

O método mostrado na Figura 37 representa a visualização de uma pesquisa por usuário. Todos os métodos para visualizar pesquisas possuem as mesmas características, mudando apenas os parâmetros. Neste método os parâmetros recebidos são:

- a) usuário que será pesquisado;
- b) fábrica escolhida no *combo box* de plantas;
- c) data escolhida no *combo box* de tempo;
- d) comentário que irá no final da pesquisa mostrando quantos dias foram utilizados nesta pesquisa.

Ao iniciar o método é feito um input do formulário compilado (.jasper) que é armazenado em uma variável do tipo *InputStream*, em seguida cria-se um *HashMap* para informar os parâmetros utilizados no relatório. É necessário existir uma conexão com a base de dados para que a pesquisa possa ser realizada.

A variável condição recebe as informações de planta e data escolhidas pelo usuário para servir de filtro na pesquisa.

É adicionado ao *HashMap* todos parâmetros e então utiliza-se uma classe da biblioteca JasperReports para preencher o formulário de pesquisa com as informações de *input*, parâmetros e conexão. O resultado é visualizado em outra classe da biblioteca chamada de *JRViewer*.

Na Figura 38 visualiza-se o método para gerar o relatório em PDF utilizando a biblioteca JasperReports.

```
public String geraPdf(String usuario, String planta, String data, String comment) {
    InputStream input = getClass().getResourceAsStream("/PesquisaUsuario.jasper");
    Map<String, Object> parametros = new HashMap<String, Object>();
    Connection con = Conexao.getConexao();

    String condicao = planta+" "+data;

    parametros.put("USUARIO", usuario);
    parametros.put("CONDICAO", condicao);
    parametros.put("COMMENT", comment);

    String arquivo = "Relatório Usuário - " + usuario.toUpperCase() + " - "
        + Controle.formatarData(new Date()) + ".pdf";

    try {
        JasperPrint print = JasperFillManager.fillReport(input, parametros, con);
        JasperExportManager.exportReportToPdfFile(print, arquivo);
        return arquivo;
    } catch (JRException ex) {
    }
    return "";
}
```

Figura 38 - Método para gerar uma pesquisa em PDF

Este método utiliza as mesmas informações para realizar o preenchimento do formulário de pesquisa. A diferença é que ao invés de mostrar o resultado utilizando a classe *JRViewer*, o formulário preenchido é exportado para um arquivo PDF utilizando a classe *JasperExportManager*, onde é necessário informar o nome que o arquivo deve possuir. Este nome é definido na variável *arquivo*.

Na Figura 39 estão representados os métodos utilizados para filtrar data e planta nas pesquisas.

```

//Métodos para verificar os dias selecionados nas Pesquisas
public String verificaComboBoxTempo(JCheckBox chb, JComboBox cb, String campo) {
    String data = "";
    if (chb.isSelected()) {
        data = "";
    } else if (cb.getSelectedIndex() == 0) {
        data = "and TO_DAYS(NOW()) - TO_DAYS("+campo+") <= 7";
    } else if (cb.getSelectedIndex() == 1) {
        data = "and TO_DAYS(NOW()) - TO_DAYS("+campo+") <= 15";
    } else if (cb.getSelectedIndex() == 2) {
        data = "and TO_DAYS(NOW()) - TO_DAYS("+campo+") <= 30";
    } else if (cb.getSelectedIndex() == 3) {
        data = "and TO_DAYS(NOW()) - TO_DAYS("+campo+") <= 60";
    }
    return data;
}

//Método para verificar qual planta está selecionada na Pesquisa
public String verificaComboBoxPlanta(JCheckBox chb, JComboBox cb) {
    String planta = "";
    if (chb.isSelected()) {
        planta = "";
    } else {
        planta = "and chamado.planta='"+cb.getSelectedItem().toString()+"'";
    }
    return planta;
}

```

Figura 39 - Métodos para filtrar tempo e planta

Estes métodos definem se as consultas na base de dados devem ser filtradas por tempo e planta e quais dados devem ser buscados. No caso do método “verificaComboBoxTempo” é verificado qual dos índices está selecionado e utiliza-se uma função do banco de dados para procurar apenas pelo dias onde a data atual subtraída da data recebida por parâmetro for menor ou igual a um determinado número de dias.

O método “verificaComboBoxPlanta” serve para verificar se existe alguma planta selecionada ou se o usuário escolheu a opção todas as plantas.

3.4 RESULTADOS E DISCUSSÃO

O objetivo deste trabalho, de desenvolver um aplicativo para controlar os eventos efetuados em alterações de formulários de impressão, foi alcançado. O aplicativo

implementado facilita o controle e a rastreabilidade das alterações efetuadas, possibilitando dar continuidade em uma alteração que outro colaborador havia iniciado graças ao fato de as informações estarem sempre atualizadas. Com um melhor gerenciamento das informações, o aplicativo possibilita também garantir que nenhum chamado será esquecido por falta de resposta do cliente. O fato de as informações ficarem armazenadas em uma base de dados e ser possível acessá-las através das pesquisas tornaram mais fáceis as tarefas de se fazer levantamentos de informações.

O aplicativo foi apresentado ao líder da área e para alguns colaboradores que serão os principais usuários. Após a sua utilização por um período de aproximadamente uma semana, foi comentado que apesar de ser uma atividade a mais que deverá ser realizada em paralelo a alteração, o usuário não precisa perder muito tempo para atualizar os dados e nem para verificar em que ponto está uma alteração. Outro ponto comentado é que o fato de possuir um controle de acesso, melhora a comunicação entre os membros da equipe, pois agora é possível saber para quem se deve perguntar caso exista alguma dúvida sobre determinada alteração. Sendo assim houve a aprovação do aplicativo desenvolvido e a planilha será totalmente substituída em breve.

Este aplicativo utiliza linguagem de desenvolvimento Java, biblioteca JasperReports e ferramenta iReport para criação dos relatórios e banco de dados MySQL. Todas essas tecnologias e ferramentas são livres, o que garante que não existe a necessidade de compra de licenças ou realizar qualquer tipo de pagamento. O aplicativo também não necessita de investimentos na parte de infra-estrutura, tornando-se viável para a equipe, pois não traz nenhum custo para o projeto e nem para a empresa.

Quanto aos trabalhos correlatos, verificam-se semelhanças com o sistema desenvolvido pela empresa Ellevo, porém em uma escala menor e voltado apenas para determinado tipo de serviço. Em relação aos trabalhos de conclusão de curso, verifica-se que o foco dos trabalhos são diferentes. Um deles busca gerenciar as mudanças de uma empresa de informática baseando-se na biblioteca ITIL e o outro serve para facilitar a requisição, armazenamento e consulta das mudanças aplicadas no ambiente de diversos sistemas de informação, bancos de dados e infra-estrutura em geral para diminuir o impacto das alterações nos serviços. Desta forma este trabalho foi feito buscando melhorar o controle e a rastreabilidade das alterações que são realizadas em formulário de impressão.

Apesar destas diferenças, destaca-se como semelhança a melhoria no controle das atividades nas soluções que foram apresentadas.

4 CONCLUSÕES

A eficaz gestão da informação é essencial para qualquer empresa atualmente. Sem o apoio de sistemas de informação e de processos automatizados, a maioria das empresas não teria capacidade de competir com as demais, pois novas tecnologias são criadas a cada dia e as empresas necessitam dessas tecnologias para conseguir entregar seus produtos, prestar seus serviços com mais a qualidade, produtividade e competitividade que as demais.

Este aplicativo foi desenvolvido pensando em automatizar um processo que antes era feito por meio de planilhas e sem condições de ser gerenciado. Pode-se dizer que os objetivos foram alcançados, pois graças ao desenvolvimento deste aplicativo existe a possibilidade de gerenciar as alterações de forma simples e objetiva sem a necessidade de despender muito tempo. Com isso gera-se uma base de dados que pode ser consultada quando necessário.

O controle de acesso ao aplicativo garante a rastreabilidade de qualquer atividade que foi realizada proporcionando uma maior interação entre os membros, que podem verificar quem realizou determinada ação.

O controle do tempo de cada evento, chamado e da última atualização que um chamado recebeu, garante o cumprimento do acordo de nível de serviço e orienta o usuário a manter contato com o cliente para que o chamado se mantenha atualizado sempre.

A possibilidade de gerar relatórios através de pesquisas também ajuda nos levantamentos de informações e acabam economizando tempo graças ao fato de não existir a necessidade de procurar as informações.

A utilização das boas práticas da biblioteca ITIL para o gerenciamento dos serviços de TI traz um grande benefício para a organização, possibilitando alinhar os serviços com as necessidades da organização garantindo que o cliente receba o melhor serviço sempre.

Como maior dificuldade destaca-se a integração dos relatórios no iReport com a base de dados, pois foi necessário criar filtros nas consultas para que as *queries* executassem corretamente. Em relação ao desenvolvimento, vários conceitos aprendidos nos semestres anteriores, como por exemplo, programação e banco de dados tiveram que ser lembrados e outros assuntos necessitaram de pesquisa com o intuito de encontrar soluções para os problemas que apareciam durante o desenvolvimento.

Conclui-se que a realização deste trabalho trouxe um aumento significativo de conhecimento na parte técnica garantindo assim um desenvolvimento pessoal valioso para trabalhos futuros.

4.1 EXTENSÕES

Dando continuidade ao projeto, seria interessante que na próxima versão o usuário pudesse realizar a alteração da descrição de um determinado evento. Esta alteração geraria um *log* para que existisse um controle das alterações que foram realizadas.

A área de pesquisa poderia ser ampliada e permitir verificar estatísticas através de gráficos. Os gráficos podem conter informações de usuários com mais chamados abertos, plantas com maior número de alterações, tipo de alteração mais freqüente, entre outros.

Para facilitar o cadastro inicial do chamado, seria interessante uma integração entre este aplicativo e o sistema utilizado para abertura de *tickets*. Esta integração iria garantir que todos os chamados abertos seriam cadastrados automaticamente no aplicativo.

REFERÊNCIAS BIBLIOGRÁFICAS

ASKIN, R.G.; STANDRIDGE, C.R. - **Modeling and Analysis of Manufacturing Systems**. New York: John Wiley & Sons, 1993.

CASTELA, Rodrigo Tenedini. **Introdução a Linguagem Java**. São Paulo, 2010. Disponível em: <<http://www.dotsharp.com.br/programacao/java/introducao-a-linguagem-java.html>>. Acesso em: 05 out. 2011.

DOCPATH. **IGP/PGL; Intelligent Graphics Printing software for xthe Printronix Graphics Language**. São Caetano do Sul, 2011. Disponível em: <<http://www.docpath.com/pt/print-output-formats.aspx>>. Acesso em: 10 mar. 2011.

DOLENC, Luciano. **Peregrine Systems lança ServiceCenter 6**. São Paulo, 2004. Disponível em: <http://www.s2publicom.com.br/imprensa/ReleaseTextoS2Publicom.aspx?press_release_id=17143>. Acesso em: 09 mar. 2011.

ELLEVO. **ELLEVO - Soluções em TI**. Blumenau, 2011. Disponível em: <<http://www.ellevoo.com.br/empresa.asp>>. Acesso em: 22 mar. 2011.

JAVAFREE, **JasperReports**. Blumenau, 2011. Disponível em: <<http://javafree.uol.com.br/wiki/JasperReports>>. Acesso em: 08 out. 2011.

MAGALHÃES, Ivan Luizio; PINHEIRO, Walfrido Brito. **Gerenciamento de serviços de TI na prática: uma abordagem com base na ITIL, inclui ISO/IEC 20.000 e IT Flex**. São Paulo: Novatec, 2007.

REZENDE, Denis Alcides. **Planejamento de sistemas de informação e informática: guia prático para planejar a tecnologia da informação integrada ao planejamento estratégico das organizações**. São Paulo: Atlas, 2003.

SCHOENFELDER, Karin. **Sistema de Gerenciamento de Mudanças Baseado nas Melhores Práticas da ITIL**. 2010. 74 páginas. Trabalho de Conclusão de Curso de Sistemas de Informação - Bacharelado - FURB, Blumenau.

SCHULDT, Clóvis Diego. **Sistema de Apoio às Mudanças de Ambientes Coporativos Baseado na Biblioteca ITIL**. 2010. 71 páginas. Trabalho de Conclusão de Curso de Sistemas de Informação - Bacharelado - FURB, Blumenau.

T-SYSTEMS. **T-Systems – um parceiro forte**. São Paulo, 2011. Disponível em: <<http://www.t-systems.com.br/tsi/pt/93300/Homepage/Sobrea-a-T-Systems/Empresa>>. Acesso em: 09 mar. 2011.

APÊNDICE A – Detalhamento dos Casos de Uso

No Quadro 3 verifica-se o caso de uso para “Alterar senha”.

<p>UC02 – Alterar Senha Ator: Usuário. Objetivo: Alterar a senha de acesso ao aplicativo. Pré-condições: Usuário estar conectado. Pós-condições: Senha alterada.</p> <p>Cenário Principal:</p> <ol style="list-style-type: none"> 1. Usuário acessa a tela de alteração de senha 2. Usuário preenche os campos: senha atual, nova senha e confirmar nova senha 3. Sistema valida os dados digitados 4. Sistema altera a senha na base de dados <p>Cenário Alternativo: No passo 3, se um dos campos estiver incorreto:</p> <ol style="list-style-type: none"> 3.1 Sistema apresenta mensagem de erro para o usuário 3.2 Usuário corrige o campo incorreto 3.3 Volta ao cenário principal

Quadro 3 - Caso de uso alterar senha

No Quadro 4 verifica-se o caso de uso “Cadastrar Chamados”.

<p>UC03 – Cadastrar Chamados Ator: Usuário. Objetivo: Cadastrar novos chamados. Pré-condições: Usuário estar conectado. Pós-condições: Chamado foi cadastrado.</p> <p>Cenário Principal:</p> <ol style="list-style-type: none"> 1. Usuário acessa a tela de cadastro de chamado 2. Usuário preenche os campos: planta, número do chamado e nome do formulário 3. Sistema valida os dados digitados 4. Sistema redireciona o usuário para tela de chamado selecionado 5. Usuário inseri um evento ao sistema preenchendo os campos: tipo de evento, duração e descrição 6. Sistema valida os campos preenchidos 7. Sistema cadastra o chamado na base de dados <p>Cenário Alternativo: No passo 3, se um dos campos estiver incorreto ou não estiver preenchido:</p> <ol style="list-style-type: none"> 3.1 Sistema apresenta mensagem de erro para o usuário 3.2 Usuário corrige ou preenche o campo 3.3 Volta ao cenário principal

Cenário Alternativo:

No passo 5, se um dos campos estiver incorreto ou não estiver preenchido:

- 5.1 Sistema apresenta mensagem de erro para o usuário
- 5.2 Usuário corrige ou preenche o campo
- 5.3 Volta ao cenário principal

Cenário Alternativo:

No passo 5, se o usuário tentar sair da tela sem inserir nenhum evento ao chamado:

- 5.1 Sistema apresenta mensagem de aviso para o usuário
- 5.2 Volta ao passo 5.

Quadro 4 - Caso de uso cadastrar chamados

No Quadro 5 verifica-se o caso de uso “Cadastrar Eventos”.

UC04 – Cadastrar Eventos

Ator: Usuário.

Objetivo: Inserir novos eventos ou finalizar chamados.

Pré-condições: Usuário estar conectado e existir um chamado pendente.

Pós-condições: Chamado atualizado ou finalizado.

Cenário Principal:

1. Usuário acessa a tela principal
2. Usuário seleciona um chamado da tabela de chamados pendentes
3. Sistema redireciona o usuário para tela de chamado selecionado
4. Usuário seleciona a opção inserir evento
5. Usuário preenche os campos: tipo de evento, duração e descrição
6. Sistema valida os campos preenchidos
7. Sistema inseri o evento na base de dados

Cenário Alternativo:

No passo 6, se o evento escolhido for um evento de finalização:

- 6.1 Sistema apresenta mensagem de chamado finalizado com sucesso
- 6.2 Volta para tela principal

Cenário Alternativo:

No passo 6, se um dos campos estiver incorreto ou não estiver preenchido:

- 6.1 Sistema apresenta mensagem de erro para o usuário
- 6.2 Usuário corrige ou preenche o campo
- 6.3 Volta ao cenário principal

Quadro 5 - Caso de uso cadastrar eventos

No Quadro 6 verifica-se o caso de uso “Cadastrar Usuário”.

UC05 – Cadastrar Usuário

Ator: Administrador.

Objetivo: Cadastrar um novo usuário na base de dados.

Pré-condições: Administrador estar conectado.

Pós-condições: Novo usuário cadastrado.

Cenário Principal:

1. Administrador acessa a tela de cadastro de usuário
2. Administrador preenche os campos: nome, *login* e escolhe se o usuário será administrador
3. Sistema valida os dados digitados
4. Sistema cadastra o novo usuário na base de dados com uma senha padrão

Cenário Alternativo:

No passo 3, se um dos campos estiver incorreto ou não estiver preenchido:

- 3.1 Sistema apresenta mensagem de erro para o administrador
- 3.2 Administrador corrige ou preenche o campo
- 3.3 Volta ao cenário principal

Quadro 6 - Caso de uso cadastrar usuário

No Quadro 7 verifica-se o caso de uso “Cadastrar Fábrica”.

UC06 – Cadastrar Fábrica

Ator: Administrador.

Objetivo: Cadastrar uma nova fábrica na base de dados.

Pré-condições: Administrador estar conectado.

Pós-condições: Nova fábrica cadastrada.

Cenário Principal:

1. Administrador acessa a tela de cadastro de fábrica
2. Administrador preenche os campos: código, nome e país
3. Sistema valida os dados digitados
4. Sistema cadastra a nova fábrica na base de dados

Cenário Alternativo:

No passo 3, se um dos campos estiver incorreto ou não estiver preenchido:

- 3.1 Sistema apresenta mensagem de erro para o administrador
- 3.2 Administrador corrige ou preenche o campo
- 3.3 Volta ao cenário principal

Quadro 7 - Caso de uso cadastrar fábrica

No Quadro 8 verifica-se o caso de uso “Cadastrar Tipo de Evento”.

UC07 – Cadastrar Tipo de Evento

Ator: Administrador.

Objetivo: Cadastrar um novo tipo de evento na base de dados.

Pré-condições: Administrador estar conectado.

Pós-condições: Novo tipo de evento cadastrado.

Cenário Principal:

1. Administrador acessa a tela de cadastro de tipo de evento
2. Administrador preenche os campos: tipo e descrição
3. Sistema valida os dados digitados
4. Sistema cadastra o novo tipo de evento na base de dados

Cenário Alternativo:

No passo 3, se um dos campos estiver incorreto ou não estiver preenchido:

- 3.1 Sistema apresenta mensagem de erro para o administrador
- 3.2 Administrador corrige ou preenche o campo
- 3.3 Volta ao cenário principal

Quadro 8 - Caso de uso cadastro de tipo de evento

No Quadro 9 verifica-se o caso de uso “Gerar Histórico de Chamado”.

UC08 – Gerar Histórico de Chamado

Ator: Usuário.

Objetivo: Pesquisar o histórico de um determinado chamado

Pré-condições: Usuário estar conectado.

Pós-condições: Pesquisa pode ser visualizada ou gerada em PDF.

Cenário Principal:

1. Usuário acessa a tela de pesquisa por chamado
2. Usuário preenche o campo do número do chamado
3. Usuário escolhe se deseja visualizar ou gerar em PDF
4. Sistema valida os dados digitados
5. Sistema mostra o resultado da pesquisa

Cenário Alternativo:

No passo 3, se o usuário selecionar a opção gerar em PDF:

- 3.1 Sistema valida os dados digitados.
- 3.2 Sistema apresenta mensagem de relatório gerado com sucesso, com o nome do arquivo e o diretório onde foi gerado.

Cenário Alternativo:

No passo 4, se o campo estiver em branco ou o chamado não existir:

- 3.1 Sistema apresenta mensagem de erro para o usuário
- 3.2 Usuário corrige ou preenche o campo
- 3.3 Volta ao cenário principal

Quadro 9 - Caso de uso pesquisa por chamado

No Quadro 10 verifica-se o caso de uso “Gerar Pesquisa por Formulário”.

UC09 – Gerar Pesquisa por Formulário

Ator: Usuário.

Objetivo: Pesquisar o histórico de alterações em determinado formulário

Pré-condições: Usuário estar conectado.

Pós-condições: Pesquisa pode ser visualizada ou gerada em PDF.

Cenário Principal:

1. Usuário acessa a tela de pesquisa por formulário
2. Usuário preenche o campo do nome do formulário e escolhe a quantidade de dias que deve ser pesquisada
3. Usuário escolhe se deseja visualizar ou gerar em PDF
4. Sistema valida os dados digitados

5. Sistema mostra o resultado da pesquisa

Cenário Alternativo:

No passo 3, se o usuário selecionar a opção gerar em PDF:

3.1 Sistema valida os dados digitados.

3.2 Sistema apresenta mensagem de relatório gerado com sucesso, com o nome do arquivo e o diretório onde foi gerado.

Cenário Alternativo:

No passo 4, se o campo estiver em branco, formulário não existir ou não existir nenhum registro naquela quantidade de dias.

3.1 Sistema apresenta mensagem de erro para o usuário

3.2 Usuário corrige ou preenche o campo

3.3 Volta ao cenário principal

Quadro 10 - Caso de uso pesquisa por formulário

No Quadro 11 verifica-se o caso de uso “Gerar Pesquisa por Usuário”.

UC10 – Gerar Pesquisa por Usuário

Ator: Usuário.

Objetivo: Pesquisar o histórico de chamados de um determinado usuário

Pré-condições: Usuário estar conectado.

Pós-condições: Pesquisa pode ser visualizada ou gerada em PDF.

Cenário Principal:

1. Usuário acessa a tela de pesquisa por usuário

2. Usuário escolhe o usuário, a planta e a quantidade de dias que deve ser pesquisada

3. Usuário escolhe se deseja visualizar ou gerar em PDF

4. Sistema valida os dados

5. Sistema mostra o resultado da pesquisa

Cenário Alternativo:

No passo 3, se o usuário selecionar a opção gerar em PDF:

3.1 Sistema valida os dados

3.2 Sistema apresenta mensagem de relatório gerado com sucesso, com o nome do arquivo e o diretório onde foi gerado

Cenário Alternativo:

No passo 4, se não existir nenhum registro para as opções escolhida:

3.1 Sistema apresenta mensagem de erro para o usuário

3.2 Usuário corrige as opções escolhidas

3.3 Volta ao cenário principal

Quadro 11 - Caso de uso pesquisa por usuário

No Quadro 12 verifica-se o caso de uso “Gerar Pesquisa por Evento”.

UC11 – Gerar Pesquisa por Evento

Ator: Usuário.

Objetivo: Pesquisar o histórico de chamados com um determinado evento

Pré-condições: Usuário estar conectado.

Pós-condições: Pesquisa pode ser visualizada ou gerada em PDF.

Cenário Principal:

1. Usuário acessa a tela de pesquisa por evento
2. Usuário escolhe evento, planta e a quantidade de dias que deve ser pesquisada
3. Usuário escolhe se deseja visualizar ou gerar em PDF
4. Sistema valida os dados
5. Sistema mostra o resultado da pesquisa

Cenário Alternativo:

No passo 3, se o usuário selecionar a opção gerar em PDF:

- 3.1 Sistema valida os dados
- 3.2 Sistema apresenta mensagem de relatório gerado com sucesso, com o nome do arquivo e o diretório onde foi gerado

Cenário Alternativo:

No passo 4, se não existir nenhum registro para as opções escolhida:

- 3.1 Sistema apresenta mensagem de erro para o usuário
- 3.2 Usuário corrige as opções escolhidas
- 3.3 Volta ao cenário principal

Quadro 12 - Caso de uso pesquisa por evento

APÊNDICE B – Dicionário de Dados

Este apêndice apresenta o dicionário de dados de tabelas do sistema, e visa fornecer uma breve descrição das tabelas e seus respectivos campos. Os campos do tipo “int” e “tinyint” representam valores numéricos. O tipo “date” serve para armazenar datas e o tipo “varchar” representa uma seqüência de letras ou palavras.

O Quadro 13 apresenta o dicionário de dados da tabela “usuario”.

Tabela: USUARIO				
Tabela responsável por armazenar as informações relativas aos usuários do aplicativo.				
Colunas:				
Nome	Tipo	Tamanho	Obrigatório	Descrição
id	<i>int</i>	10	Sim	Chave primária representa o id do usuário
<i>login</i>	<i>varchar</i>	50	Sim	Nome utilizado pelo usuário para acessar o aplicativo
senha	<i>varchar</i>	50	Sim	Senha do usuário
nome	<i>varchar</i>	75	Sim	Nome do usuário
admin	<i>tinyint</i>	1	Sim	Indica se o usuário é ou não administrador

Quadro 13 - Dicionário de dados da tabela usuário

O Quadro 14 apresenta o dicionário de dados da tabela “planta”.

Tabela: PLANTA				
Tabela responsável por armazenar as informações relativas às fábricas				
Colunas:				
Nome	Tipo	Tamanho	Obrigatório	Descrição
id	<i>int</i>	10	Sim	Chave primária representa o id da fábrica
numero	<i>varchar</i>	3	Sim	Código utilizado para representar determinada fábrica
nome	<i>varchar</i>	50	Sim	Nome da fábrica
pais	<i>varchar</i>	50	Sim	País onde a fábrica está localizada

Quadro 14 - Dicionário de dados da tabela planta

O Quadro 15 apresenta o dicionário de dados da tabela “tipo”

Tabela: TIPO				
Tabela responsável por armazenar as informações dos tipos de evento utilizados no aplicativo				
Colunas:				
Nome	Tipo	Tamanho	Obrigatório	Descrição
id	<i>int</i>	10	Sim	Chave primária representa o id do tipo
tipo	<i>varchar</i>	50	Sim	Nome do tipo de evento
descricao	<i>varchar</i>	75	Sim	Descrição mais detalhada do tipo de evento

Quadro 15 - Dicionário de dados da tabela tipo

O Quadro 16 apresenta o dicionário de dados da tabela “chamado”

Tabela: CHAMADO				
Tabela responsável por armazenar as informações dos chamados cadastrados.				
Colunas:				
Nome	Tipo	Tamanho	Obrigatório	Descrição
id	<i>int</i>	10	Sim	Chave primária representa o id do chamado
ticket	<i>varchar</i>	12	Sim	Número do chamado aberto no sistema PSC
formulario	<i>varchar</i>	40	Sim	Nome do formulário que está sendo alterado
planta	<i>varchar</i>	45	Sim	Nome da fábrica
usuario	<i>varchar</i>	50	Sim	Nome do usuário que cadastrou o chamado
data	<i>date</i>		Sim	Data da abertura do chamado
pendente	<i>tinyint</i>	1	Sim	Indica se o chamado está pendente ou finalizado

Quadro 16 - Dicionário de dados da tabela chamado

O Quadro 17 apresenta o dicionário de dados da tabela “evento”

Tabela: EVENTO				
Tabela responsável por armazenar as informações dos eventos efetuados em um chamado.				
Colunas:				
Nome	Tipo	Tamanho	Obrigatório	Descrição
id	<i>int</i>	10	Sim	Chave primária representa o id do evento
ticket	<i>varchar</i>	12	Sim	Número do chamado aberto no sistema PSC
tipo	<i>varchar</i>	50	Sim	Tipo do evento
descricao	<i>varchar</i>	200	Sim	Descrição do que foi realizado neste evento
data	<i>date</i>		Sim	Data do evento
duracao	<i>int</i>	10	Sim	Duração da ação do evento
usuario	<i>varchar</i>	50	Sim	Usuário que inseriu o evento

Quadro 17 - Dicionário de dados da tabela evento