

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

FERRAMENTA PARA CONTROLE E MONITORAMENTO
DE SERVIÇOS NA ÁREA DE TI DE UMA EMPRESA

RAFAEL ANTONIO PIRES

BLUMENAU
2011

2011/1-18

RAFAEL ANTONIO PIRES

**FERRAMENTA PARA CONTROLE E MONITORAMENTO
DE SERVIÇOS NA ÁREA DE TI**

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. Francisco Adell Péricas, Mestre – Orientador

**BLUMENAU
2011**

2011/1-18

FERRAMENTA PARA CONTROLE E MONITORAMENTO DE SERVIÇOS NA ÁREA DE TI

Por

RAFAEL ANTONIO PIRES

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

Presidente: _____
Prof. Francisco Adell Péricas, Mestre – Orientador, FURB

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

Membro: _____
Prof. Roberto Heinzle, Doutor – FURB

Blumenau, 28 de junho de 2011.

Dedico este trabalho aos meus familiares e amigos, especialmente aqueles que me ajudaram diretamente e incentivaram na realização deste.

AGRADECIMENTOS

A Deus, pelo seu imenso amor e graça.

À minha família, que sempre esteve presente e me apoiando.

Aos meus amigos, pelas ajudas e incentivos.

Ao meu orientador, Francisco Adell Péricas, por ter acreditado na conclusão deste trabalho.

Mesmo as noites totalmente sem estrelas
podem anunciar a aurora de uma grande
realização.

Martin Luther King

RESUMO

Este trabalho apresenta uma ferramenta *web*, que proporcione ao gerente de suporte uma visão sobre o estado do ambiente de TI e dos serviços que são executados nos diferentes cenários de uma organização, estes resultados serão possíveis através das informações geradas com o preenchimento do *checklist* de validação. O desenvolvimento da ferramenta foi feito na linguagem JSP e utilizando banco de dados livre MySQL. Destaca-se como resultado o controle dos principais componentes que definem um ambiente de TI e o aumento da velocidade para correção de problemas, com bases de soluções e alertas de falhas.

Palavras-chave: Gerenciamento. Controle. Melhoria. Serviços de TI.

ABSTRACT

This paper presents a web tool to provide support to the manager an insight into the state of the IT environment and services that run on different scenarios of an organization, these results are possible through the information generated by completing the checklist validation . The development of the tool was done in JSP and using language free database MySQL. Stands out as a result the control of the main components that define an IT environment and increased speed to correct problems with foundations of solutions and warnings of failure.

Keywords: Management. Control. Improvement. IT Services.

LISTA DE ILUSTRAÇÕES

Figura 1 – Nível hierarquico x tipo de sistema.....	17
Figura 2 – Ambiente de apoio a tomada de decisão	18
Figura 3 – Funcionamento de um ambiente <i>WEB</i>	20
Figura 4 – Fluxo de processos da <i>SMF Service Monitoring and Control</i>	23
Figura 5 – <i>Check list</i> atual	24
Figura 6 – Fluxo de preenchimento do <i>checklist</i>	29
Quadro 1 – Requisitos funcionais	30
Quadro 2 – Requisitos não funcionais	30
Figura 7 – Diagrama de caso de uso Administrador	32
Figura 8 – Diagrama de caso de uso Funcionário	33
Figura 9 – Diagrama Entidade Relacionamento (DER)	34
Quadro 3 – Dicionário de dados da tabela "usuário"	34
Quadro 4 – Dicionário de dados da tabela " <i>checklist</i> "	35
Quadro 5 – Dicionário de dados da tabela "solução"	36
Figura 10 – Funcionamento da requisição em JSP	37
Figura 11 – Exemplo código fonte	38
Figura 12 – <i>Logon Checklist</i>	39
Figura 13 – Código de usuário não informado	39
Figura 14 – Itens do menu	40
Figura 15 – Cadastrar novo usuário	41
Figura 16 – Verificação dos campos obrigatórios	41
Figura 17 – Código <i>javascript</i> da verificação dos campos	42
Figura 18 – Cadastrar solução	43
Figura 19 – Consulta de usuário	44
Figura 20 – Consulta de usuário parcial	44
Figura 21 – Itens do <i>checklist</i>	45
Figura 22 – Opções de <i>check</i>	46
Figura 23 – <i>E-mail</i> recebido	46
Figura 24 – Gráfico pizza problema por item	47
Figura 25 – Gráfico coluna problema por hora	48
Figura 26 – Acesso negado	48

LISTA DE SIGLAS

CEP – *Complex Event Processing*

CSS – *Cascading Style Sheets*

EA – *Enterprise Architect*

ERP – *Enterprise Resource Planning*

HTML – *Hyper Text Markup Language*

HTTP – *Hyper Text Transfer Protocol*

IA – *Inteligência Artificial*

JSP – *Java Server Pages*

MER – *Modelo Entidade Relacional*

MOF – *Microsoft Operations Framework*

MOM – *Microsoft Operations Manager*

RF – *Requisitos Funcionais*

RNF – *Requisitos Não Funcionais*

SAD – *Sistema de Apoio à Decisão*

SGBD – *Sistema Gerenciador de Banco de Dados*

SI – *Sistema de Informação*

SIE – *Sistema de Informação Executiva*

SIG – *Sistema de Informação Gerencial*

SPT – *Sistema de Processamento de Transações*

SQL – *Structured Query Language*

TI – *Tecnologia da Informação*

UML – *Unified Modeling Language*

SUMÁRIO

1 INTRODUÇÃO	12
1.1 OBJETIVOS DO TRABALHO	13
1.2 ESTRUTURA DO TRABALHO	14
2 FUNDAMENTAÇÃO TEÓRICA	15
2.1 SISTEMAS DE INFORMAÇÃO.....	15
2.1.1 SISTEMAS DE APOIO À DECISÃO.....	17
2.2 AMBIENTE <i>WEB</i>	19
2.3 MONITORAMENTO DE SERVIÇOS EM TI.....	21
2.4 SISTEMA ATUAL.....	24
2.5 SISTEMAS DE INFORMAÇÃO.....	25
3 DESENVOLVIMENTO.....	27
3.1 DESCRIÇÃO DO SISTEMA.....	27
3.2 REQUISITOS DO SISTEMA	29
3.3 ESPECIFICAÇÃO	31
3.3.1 Técnicas e ferramentas utilizadas	31
3.3.2 Técnicas e ferramentas utilizadas	33
3.3.3 Técnicas e ferramentas utilizadas	34
3.4 IMPLEMENTAÇÃO	36
3.4.1 Técnicas e ferramentas utilizadas	36
3.4.2 Operacionalidade da implementação.....	38
3.5 RESULTADOS E DISCUSSÃO	49
4 CONCLUSÕES.....	50
4.1 EXTENSÕES	50
REFERÊNCIAS BIBLIOGRÁFICAS	52
APÊNDICE A – Detalhamento dos casos de uso	55

1 INTRODUÇÃO

Segundo Pereira (2007), por muito tempo algumas organizações conseguiram manter seu negócio com pouca ajuda da Tecnologia da Informação (TI). Mas hoje a Tecnologia da Informação é algo imprescindível para uma organização obter sucesso, ou mesmo um diferencial competitivo no mercado.

As organizações necessitam de um sistema seguro, livre de falhas, ou que pelo menos estas sejam resolvidas em um curto tempo para evitar paralisações que possam causar algum transtorno para empresa. Como um processo é um conjunto de atividades inter-relacionadas, nunca estará livre de erros ou falhas, mas é possível minimizá-las e até mesmo extinguir alguns problemas. Para isso é preciso controle e monitoramento destas atividades, e assim minimizar os riscos destas falhas.

Uma das funções da área de TI de uma organização é manter todo um sistema, que inclui o hardware e software, funcionando. Para se chegar a um sistema seguro e de qualidade é preciso controlar suas principais atividades e as que mantêm a integridade do sistema. Attie (1986, p. 217) afirma que “a ausência de controles adequado para empresas de estrutura complexa as expõe a riscos inúmeros e infindáveis de todas as espécies”.

Com o aumento da sua importância dentro das organizações, a TI está encontrando diversos desafios. Os gerentes de suporte da área de TI detêm diversas funções e responsabilidades no que diz respeito à funcionalidade do sistema, tais como o controle e preservação da infraestrutura do ambiente, manutenção de serviços e atividades do sistema, e para manter todo este ambiente funcionando precisa-se de algo que os auxiliem nesta tarefa. Com o desenvolvimento deste trabalho pretende-se criar uma ferramenta *web* que dará ao gerente de TI mais controle sobre o ambiente, identificar possíveis problemas antes que estes ocorram, dando ao usuário do sistema um ambiente confiável.

Para garantir a disponibilidade, desempenho e alta confiabilidade de um ambiente, é possível criar uma ferramenta eficaz de manutenção. Com isto os administradores de TI precisam identificar e classificar um conjunto de processos e atividades de manutenção para cada sistema e serviço do ambiente da organização. Devem-se destinar essas tarefas e atividades a grupos ou pessoas específicas e verificar se estão sendo concluídas corretamente.

A idéia surgiu da necessidade de manter e gerenciar o sistema da empresa e deixá-lo sempre funcional, pois devido a falhas de alguns processos e serviços, algumas partes vitais podem paralisar, causando transtornos aos seus usuários. Além disso, também há a

necessidade de uma melhoria contínua do ambiente do sistema, e com o controle do funcionamento é possível localizar estes pontos com falhas e tomar as devidas decisões.

Observando a atual forma de monitorar o ambiente de TI da empresa, identifica-se uma deficiência no tempo de resposta para se tratar um problema. Ao identificá-lo, o operador de computador responsável pela execução do *checklist* tenta resolvê-lo, mas por sua pouca experiência e a falta de uma base de conhecimento do sistema, irá demorar a resolver ou encaminhar para o analista responsável. Outro ponto verificado ocorre com os problemas recorrentes, ou seja, problemas que ocorrem com certa frequência e que são tratados pelo operador, porém não são encaminhados para análise e solução definitiva.

Como cada ambiente de TI é único, então a técnica para monitorá-lo deve atender a esta exclusividade. Por não existir uma solução genérica que atenda inteiramente a este monitoramento, os gastos com softwares especializados em cada função são altos.

Com base nestas informações, este trabalho propõe a criação de uma ferramenta para monitorar um ambiente de TI que garanta uma manutenção proativa, pois permitirá aos administradores serem alertados e tomarem as medidas corretivas antes que os problemas se tornem graves e afetem a confiabilidade do sistema.

Utilizar um programa *web* para gerenciar e controlar a funcionalidade de todo um sistema empresarial trará ao gerente responsável pelo suporte mais rapidez quando necessitar verificar o estado de algum serviço, pois poderá ver de qualquer lugar e tomar as providências de lá mesmo. Além disso, o tempo com as análises também serão menores, pois contará com mais informações, e algumas já tratadas anteriormente, bastando apenas tomar a decisão sobre o que fazer para evitar os pontos de riscos.

O ambiente *web* trará maior liberdade para quem irá utilizá-lo, pois não será necessário instalar nenhum software para o funcionamento além de navegadores de internet.

1.1 OBJETIVOS DO TRABALHO

O objetivo deste trabalho foi criar uma ferramenta *web* que ajude no controle e monitoramento dos diversos serviços e processos da área de TI, com relação ao seu funcionamento.

Os objetivos específicos do trabalho são:

- a) permitir o preenchimento do *checklist* de validação do ambiente;

- b) permitir o cadastramento de roteiros de solução de problema;
- c) aumentar o controle com relação ao funcionamento do ambiente e falhas.

1.2 ESTRUTURA DO TRABALHO

O trabalho está disposto em quatro capítulos. No primeiro capítulo apresenta-se a introdução, os objetivos do trabalho e a estrutura do trabalho.

No segundo capítulo tem-se a fundamentação teórica, destacando-se os conceitos de Sistemas de Informação e Sistemas de Apoio a Decisão, as tecnologias utilizadas para a elaboração, bem como os trabalhos correlatos.

No terceiro capítulo é apresentado o desenvolvimento do aplicativo, incluindo detalhes sobre a especificação, implementação e tecnologia utilizada.

No quarto capítulo tem-se a conclusão, e sugestões para trabalhos futuros.

2 FUNDAMENTAÇÃO TEÓRICA

Este capítulo aborda os assuntos que ajudarão a compreender o trabalho, tais como sistemas de informação, ambientes *web*, monitoramento de serviços de TI, além dos trabalhos correlatos.

2.1 SISTEMAS DE INFORMAÇÃO

Sistema de Informação significa, de acordo com O'Brien (2004, p.6), “um conjunto organizado de pessoas, hardwares, softwares, rede de comunicações e recursos de dados que coleta, transforma e dissemina informações em uma organização”. Há algum tempo as grandes empresas usavam grandes arquivos, que eram manipulados por um arquivista, responsável por organizar, registrar, catalogar e recuperar os dados. Essa forma de manipular as informações e a necessidade de integração entre as áreas começou a se tornar inviável, pois com o aumento das informações a tarefa ficou difícil e pouco confiável. Com a comercialização dos computadores, criou-se a possibilidade de um acesso a informação de forma rápida e confiável para a tomada de decisões.

A evolução da tecnologia permite uma interatividade maior e possibilita um fluxo de informação em tempo real. Para Orlandini (2005), a principal vantagem proporcionada pela tecnologia aos Sistemas de Informação é a capacidade de processar um gigantesco número de dados simultaneamente, tornando a disponibilização das informações praticamente *on-line*.

Para Laudon e Laudon (2004), sistema de informação é um conjunto de elementos que interagem e coletam, processam, guardam e difundem informações que servem como base para tomada de decisão. Hoje os sistemas de informação são importantes ferramentas no auxílio aos dirigentes nas tomadas de decisão, pois lhe dão um acesso confiável e rápido às informações da organização e de seu negócio.

A informação e a capacidade de extrair e aplicar o conhecimento são de extrema importância para as empresas manterem e aumentarem suas atividades e concorrerem no mercado competitivo atual.

O mundo vem mudando em aceleração crescente, significa que a velocidade que aprendíamos antes é insuficiente para mantermos nossos diferenciais competitivos agora. A única vantagem competitiva reside em nossa capacidade de aprender mais rápido que a concorrência. Aprenda com os acertos da concorrência, estudando e utilizando como ponto de partida para inovar os negócios. Aprenda também com os erros da concorrência, afinal, não existe erro melhor para aprender do que os financiados pelos outros. Os erros da concorrência certamente custaram caros pra ela, aprendendo com eles não teremos que arcar com os mesmo custos, isso é Inteligência Competitiva. (HILSDORF, 2009).

Segundo Barbosa Junior (2009), a facilidade de processar e utilizar informações de forma rápida deixou de ser, nos últimos dez anos, um diferencial e virou padrão para empresas de qualquer porte. A computação nas empresas tem uma linha de evolução única, tendo como principal objetivo possibilitar que a informação chegue de forma ágil e confiável conforme as necessidades dos gestores, assim criando novas e melhores ferramentas de apoio à tomada de decisão. Uma prova disso são os *Enterprise Resource Planning* (ERP), que melhoram o tráfego de dados dentro da empresa, tornando os processos automatizados e assim garantindo uma maior confiabilidade para as informações.

Existem diversos tipos de sistemas de informação e pode-se dividir alguns deles conforme os níveis organizacionais das empresas, que geralmente se dividem em três, o operacional, o tático e o estratégico.

No nível operacional há os sistemas de processamento de transações (SPT). No nível tático, há dois tipos de SI, os sistemas de informação gerencial (SIG) e os sistemas de apoio à decisão (SAD). Em se tratando de nível estratégico, que está no topo da estrutura, há os sistemas de informação executiva (SIE).

O SPT é quem processa as transações operacionais de uma empresa, servindo como base para os demais sistemas dentro da empresa. Exemplos destas transações são os fechamentos de pedidos, emissão de notas fiscais e baixas de estoque.

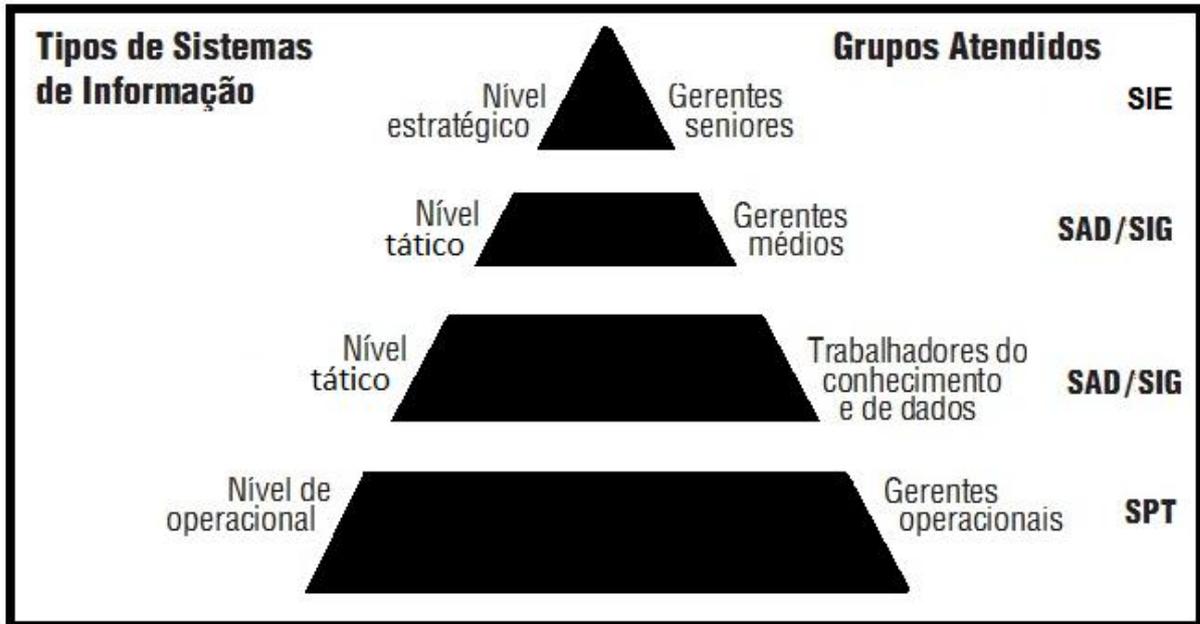
Para os SIG, a prioridade está na saída das informações. Esse sistema extrai as informações das bases de dados e de processos que estão de acordo com o SIG. Segundo Oliveira (1998), após a coleta dos dados e a transformação dos mesmos em informações, ele tem como principal função prover ao gerente informações passadas e presentes sobre as operações internas e sobre o ambiente da empresa, orientando no processo decisório.

O SAD tem como princípio o tratamento de situações onde os problemas são semi-estruturados ou não-estruturados. Sua principal característica é o uso de dados de diferentes fontes, preocupação com o estilo de quem toma as decisões e possibilidade de simulação.

Um SIE é bastante interativo, permitindo ao usuário obter relatórios que indiquem situações fora dos parâmetros estipulados pelos planos de empresa. Além disso, pela análise

de tendências, permite que o executivo antecipe situações que alterem o panorama de negócio em que a organização atua.

A figura 1 mostra a relação entre os níveis hierárquicos de uma empresa e os tipos de sistemas de informação.



Fonte: Adaptado de Universidade Federal do Rio Grande do Norte (2007).

Figura 1 - Nível hierárquico x tipo de sistema

2.1.1 Sistema de Apoio à Decisão

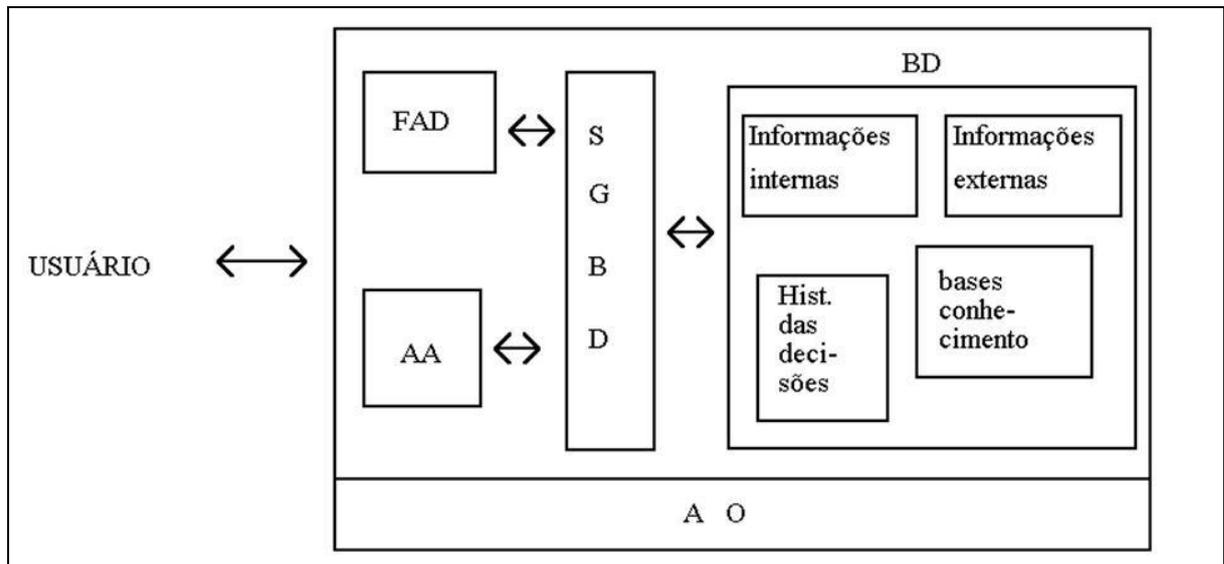
O SAD surgiu do desafio que o mercado competitivo vem exigindo a cada dia da eficiência em relação à localização e gerenciamento de dados para a tomada de decisão. Com a necessidade de ter as informações sempre atuais, de forma rápida, acompanhar as atividades da empresa, aderiu-se a esse sistema, sabendo que este é favorável às atividades sem contar com as concorrências externas que estavam na frente em relação à tecnologia.

Segundo Power (2002), no final dos anos oitenta deu-se início às primeiras tentativas de utilizar técnicas de Inteligência Artificial (IA) para aprimorar a capacidade dos sistemas. A partir dos anos noventa, inúmeras técnicas e métodos de IA foram explorados, os quais, agregadas às novas tecnologias de comunicação disponíveis, deram início a uma nova geração de Sistemas de Apoio à Decisão.

Segundo Watson (1991), SAD são sistemas de informação que apóiam qualquer processo de tomada de decisão em áreas de planejamento estratégico, controle gerencial e controle operacional. O SAD pode ser descrito como um sistema de informação interativo, flexível e adaptável, que foi desenvolvido para apoiar na solução de um problema não estruturado e dar suporte a tomada de decisão. Um SAD utiliza dados e disponibiliza um interface amigável para permitir ao gerente interagir com o mesmo usando a sua percepção para apoiar na tomada de decisão.

Segundo Heinzle (2011), o autor Laudon chama o SAD de "Sistemas de Suporte à Decisão - DSS", e que se encaixa em uma classe de sistemas apóiam os gestores na tomada de decisão em problemas pouco estruturados. E continua afirmando que para Laudon, estes sistemas auxiliam o executivo principalmente nas etapas de desenvolvimento, comparação e classificação de riscos, fornecendo subsídios úteis para a escolha de uma alternativa.

O processo de tomada de decisão se dá através da interação constante do usuário com um ambiente de apoio à decisão, especialmente criado para dar subsídio às decisões a serem tomadas. Esse ambiente é representado na Figura 2.



Fonte: Chaves (2004).

Figura 2 - Ambiente de apoio à tomada de decisão

Os componentes que constam na figura 2 são:

- Bancos de Dados (BD): são formados por informações internas e externas à organização, por conhecimentos e experiências de especialistas e por informações históricas acerca das decisões tomadas;

- b) Sistema Gerenciador de Banco de Dados (SGBD): após os dados estarem instalados no BD, o SGBD deve possibilitar o acesso às informações e a sua atualização, garantindo a segurança e a integridade do BD;
- c) Ferramentas de Apoio à Decisão (FAD): são softwares que auxiliam na simulação de situações, na representação gráfica das informações;
- d) Ambiente Aplicativo (AA): são sistemas aplicativos ou funções acrescentadas aos sistemas existentes que fazem análise de alternativas e fornecem soluções de problemas;
- e) Ambiente Operacional (AO): é composto por hardwares e softwares que permitem que todos os componentes do ambiente sejam integrados.

Segundo Turban (2004), o departamento de sistemas de informação não tem conseguido mais atender à diversidade de necessidades imediatas da empresa e de seus executivos e não havia funções de análise de negócio embutidas nos sistemas existentes. Um SAD eficiente permite fácil interação com o usuário do sistema, para que este possa acessar seu banco de dados e absorver de forma natural as informações e sugestões armazenadas, obtendo vantagem competitiva no mercado em que atua.

2.2 AMBIENTE WEB

Com o desenvolvimento da tecnologia, os hábitos da sociedade estão mudando. Isso tornou a internet muito mais acessível e utilizada. Hoje muitas pessoas compram, vendem, pesquisam e trabalham pela internet. Além disso, é o meio de comunicação e informação com mais abrangência do mundo todo. Segundo Tomael (2008), a internet é considerada uma fonte valorosa de informação que subsidia – com seus recursos – desde pesquisas de alta tecnologia até estudantes de ensino básico.

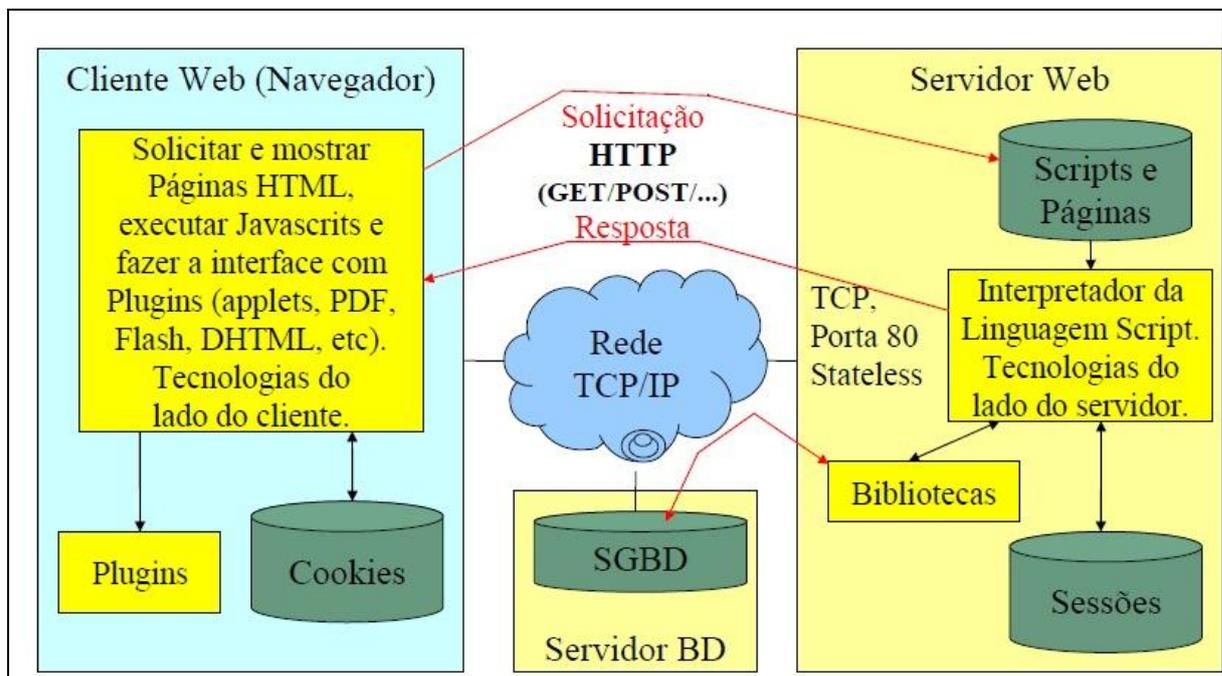
A informática e a internet trazem muitos benefícios às pessoas, principalmente para as empresas, como rapidez, economia e eficácia, e substituem o uso de papel, correio postal, telefone e fax, pelo uso do *e-mail* e *chats*.

Outra tendência, e que já é realidade em muitos negócios, é trabalhar em casa, onde se utiliza o termo *Home Office*. Isto só se tornou possível com o avanço da tecnologia, como os celulares *smartphones*, computadores de alto desempenho, internet banda larga e móvel, etc. Isto está mudando a cultura organizacional tradicional para algo mais interativo.

Segundo a Oracle (2006), para ser possível trabalhar de qualquer lugar é preciso ter um sistema projetado para *web*, ou seja, um sistema que utiliza o navegador como ambiente de execução. Esses sistemas são conjuntos de programas que rodam em um servidor e que para acessá-los utilizam a internet ou a intranet.

As aplicações *web* utilizam uma arquitetura multinível onde as funções executadas pelas aplicações podem estar distribuídas por uma rede de computadores. Elas fazem uso de uma infra estrutura de rede que é o padrão atual adotado pela internet.

Ao nível de serviços, os elementos são organizados da seguinte forma: de um lado está o cliente *web*, ou navegador, que solicita dados ao servidor *web*, e recebe as respostas, formata a informação e apresenta para o usuário. Do outro lado, está o servidor *web* que recebe as requisições, lê os dados do disco e as retorna para o cliente conforme apresentado na Figura 3.



Fonte: Segundo (2008).

Figura 3 - Funcionamento de um ambiente *web*

Segundo Macoratti (2010), as vantagens de se ter um ambiente *Web* são:

- interface HTML reconhecida por uma grande gama de usuários já acostumados com o funcionamento dos navegadores;
- desenvolvimento, manutenção e atualização centralizada da aplicação. Não se precisa instalar aplicativos em diversos equipamentos diferentes, basta colocá-lo no servidor para que os usuários acessem;

- c) a exportação de dados entre usuários remotos usando o protocolo HTTP é mais fácil que outros protocolos;
- d) escalabilidade no processamento. Se houver necessidade de aumentar o poder de processamento, basta fazer isto no servidor.

Já as desvantagens são:

- a) a interface HTML pode ser um problema, pois há pouca padronização entre os diversos navegadores e sua aplicação poderia ser exibida de uma maneira diferente dependendo do navegador;
- b) a entrada de uma grande massa de dados é prejudicada na interface HTML pois não existe uma maneira padrão de criar máscaras de entrada de dados;
- c) a interface HTML não é rica em controles gráficos e peca no quesito posicionamento. O visual da aplicação pode não ficar tão elegante como esperado;
- d) a integração com outros componentes não é tão fácil com HTML.

2.3 MONITORAMENTO DE SERVIÇOS EM TI

Em um ambiente de TI, especificamente na área de suporte, deve-se garantir o funcionamento do sistema de uma empresa. Isto engloba todos os programas que são utilizados na empresa, como o ERP, sistemas de *callcenter*, portaria, controle de produção, serviços básicos com internet e *e-mail*, e toda a infraestrutura, como a funcionalidade dos servidores, temperatura da sala do CPD, *racks* de rede, entre outras funções específicas.

Segundo Marques (2007), se não é possível adivinhar quando haverá algum problema em um ambiente ao menos se pode monitorá-lo com a intenção de prever eventuais incidentes. Ao se implantar um sistema de monitoramento que permita acompanhar de forma precisa e rápida a situação dos serviços de TI, pode-se garantir que os serviços estarão sempre de acordo com o planejado.

Monitorar as obrigações de níveis de serviço e comparar ao que realmente está rendendo não é simples. Isso porque existe um grande número de serviços e nem sempre estão centralizados em um único lugar. Em alguns casos o sucesso do rendimento de um serviço é avaliado pelo número de incidentes gerados.

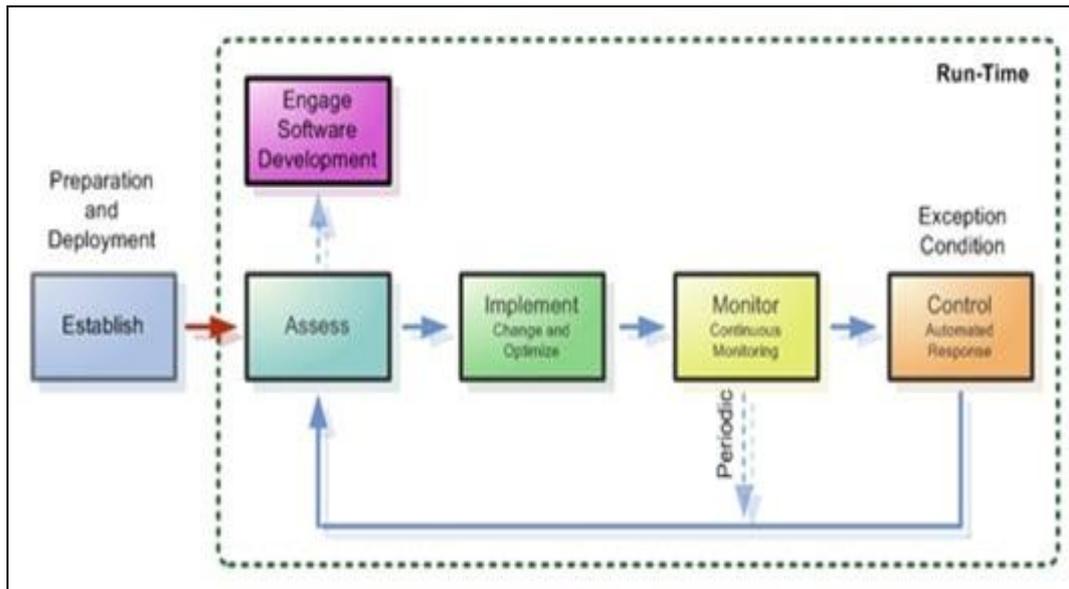
Algumas empresas contam com uma abordagem reativa para garantia de um serviço de TI, onde espera o problema ocorrer para verificar o funcionamento de um serviço. Já outras

empresas estão sendo proativas, com uma intervenção intensiva nos serviços, como conectar-se periodicamente a um serviço para testar seu desempenho manualmente. Outras abordagens proativas podem ser mais automatizadas, e para isto utilizam diversas ferramentas de monitoramento.

Em alguns casos este tipo de monitoramento pode se tornar caro, difícil de implantar e apresentar problemas para manter. Além disto, pode oferecer riscos para segurança do sistema, pois pode haver necessidade de abrir permissões no *firewall* corporativo. Nos piores casos, pode se tornar ineficaz em alcançar os objetivos desejados.

Conforme a Network World (2010), em estudos feitos pela Aberdeen Group, as empresas que conseguem fazer boas escolhas e aplicarem bem as ferramentas de monitoramento alcançam benefícios significativos. Conforme pesquisa da consultoria, elas conseguem se antecipar a 53% de problemas nas aplicações antes de receber uma reclamação e percebem uma melhoria de 48% no tempo de correção de falhas no desempenho. Além disso, as companhias que utilizam esse tipo de ferramenta melhoram a visibilidade sobre transações críticas para os negócios em 42% e reduzem o número total de reclamações dos usuários em 15%.

No *Microsoft Operations Framework 4* (MOF 4, 2010) há o *SMF Service Monitoring and Control* que permite realizar uma monitoração em tempo real dos serviços do ambiente de TI na organização. O principal objetivo da SMF é monitorar o ambiente, possibilitando corrigir as falhas quase que imediatamente, visando retomar o funcionamento normal e melhorar o desempenho de cada serviço. Para implantar um fluxo de controle pode-se seguir as atividades da SMF. Na figura 4 há um diagrama que representa o fluxo a ser seguido.



Fonte: Marques (2007).

Figura 4 - Fluxo de processos da SMF

Conforme Marques (2007), as etapas para se conseguir implantar um sistema e garantir sua qualidade são:

- a) *Establish*: esta etapa é a de preparação e implementação do sistema de monitoramento. É feito o levantamento dos serviços a serem monitorados e sua situação atual;
- b) *Assess*: semelhante com a anterior, mas esta não é mais para a implementação do sistema, e sim uma verificação das atualizações e alterações que ocorrem no sistema, e que devem ser incluídas no processo de monitoramento;
- c) *Implement*: neste momento do processo é feita a implementação da ferramenta, levando em conta as decisões da etapa anterior. Esta etapa deve ser efetuada pelos analistas de suporte;
- d) *Monitor (Continuous Monitoring)*: aqui é feito o monitoramento contínuo do sistema, onde é verificado cada ponto do *checklist*, e registrado, de modo que poderá ajudar a equipe a identificar outros problemas no sistema;
- e) *Control*: nesta etapa é feita a correção do problema encontrado durante o monitoramento e que estão causando eventuais paralisações no sistema. Ao verificar o problema, são apresentadas possíveis soluções e procedimentos que devam ser feitos para corrigir o problema. Caso não seja resolvido, é enviado o *e-mail* alertando o problema para o responsável pelo suporte da empresa.

2.4 SISTEMA ATUAL

A ferramenta proposta neste trabalho trouxe um maior controle sobre o ambiente de TI, pois com a análise freqüente dos serviços e processos críticos, a confiabilidade do sistema aumentará e os erros e problemas por falta de verificação e demora em constatar os erros será reduzido.

Atualmente o processo é feito de forma manual e no papel, não possibilitando um controle de falhas adequado. Ao chegar, o operador do atendimento imprime o *checklist*, que contém os principais serviços que foram levantados pelo departamento de suporte da empresa, e as horas em que deve-se fazer as verificações. A figura 5 mostra sem todas as horas o modelo utilizado.

DATA ___/___/___		
	5h	6h
Operador		
Temperatura		
Ar 1 Ligado?		
Ar 2 Ligado?		
NFePackIntegrator		
NfePackPrintService		
NfePackService		
Serviço de publicação FTP		
Filas Normais		
Fila 06 - NFe		
Fila 16 - NFe		
SisInfoServer		
IniciaRelatório		
Servidor Office		
Ctvoicer		
Tarifador - Calculo Automático		
Tarifador - Coletor de Dados		
Tarifador - SKTFTP		
Suricato		
CVR803		
CVR901		
TSLIN01		
TSLIN04		
TSLIN05		
TSLINFS		
TS-PLA01		
Lincenet		
Singe		
E-mail		
Navegação		

Figura 5 - *Checklist* atual

Neste *checklist* operador deve preencher primeiramente a data e seu nome em cada hora que foi responsável por validar. Caso ocorra uma falha durante o intervalo das verificações, ele deve preencher o formulário com um X para indicar uma falha. No outro dia deverá entregar este *checklist* para o responsável pelo suporte.

O operador por ter menos experiência, não consegue resolver todos os problemas que pode encontrar. Então quando encontra um problema e este não consegue resolver, procura o analista de suporte para ajudá-lo a resolver o problema. Este processo de passar para o analista pode levar algum tempo e com isso casar paralisações indesejadas no sistema.

Outra falha encontrada neste processo são os eventos recorrentes. Muitas vezes o operado identifica um problema e resolve, porém trata-se de um problema que vem ocorrendo seguidamente, e o mesmo não é relatado para o analista resolver definitivamente ou localizar a causa destas falhas.

Por se tratar de uma verificação feita no papel, não dá para o analista de suporte ter uma visão geral do funcionamento do sistema. Pois para verificar as falhas que ocorreram em um determinado serviço terá que buscar em um arquivo e verificar várias folhas com os *checklist*.

2.5 TRABALHOS CORRELATOS

Microsoft Operations Manager (MOM), segundo a Microsoft (2005) é um sistema que ajuda a disponibilizar os conhecimentos necessários para evitar e reduzir a complexidade da gestão de um sistema moderno de uma infraestrutura de TI e baixar os custos operacionais. Uma de suas funções *web* permite verificar o estado de um computador e receber notificações por *e-mail* com ligações para um problema específico que requer solução.

Complex Event Processing (CEP), segundo a Cyberconn (2007), é um sistema de informação capaz de gerar em tempo real um conhecimento imediato de um fato em determinadas condições e responder instantaneamente. Um evento é simplesmente algo que acontece. Um não-evento é o contrário, ou seja, algo que não acontece. Uma cotação enviada por um cliente é um evento. Uma transação financeira é um evento. Uma requisição de compra é um evento. Um não-evento pode ser um pedido de compra que está aguardando a análise do aprovador.

Luiz (2009) apresentou como Trabalho de Conclusão de Curso na Universidade Regional de Blumenau um software de controle e gerenciamento de documentos. O sistema teve como principal objetivo o desenvolvimento de um software de gerenciamento e controle de documentos de projetos, onde é possível cadastrar projetos, vincular cronogramas, cadastrar etapas, atividades e sub-atividades, vincular documentos, controlar prazos, bem como gerenciar as entregas dos documentos necessários aos projetos. Este sistema não é equivalente em termos de gestão de serviços do ambiente, mas sim no que se refere ao objetivo geral deste trabalho, o qual é o controle genérico dos serviços e processos críticos de um ambiente de TI.

Tanto os sistemas citados nos trabalhos correlatos quanto ferramenta proposta têm um aspecto muito importante em comum: identificar falhas e evitar problemas de serviços em um ambiente de TI.

3 DESENVOLVIMENTO

Neste capítulo estão descritas as particularidades técnicas do sistema proposto tais como a descrição do mesmo e a apresentação dos requisitos funcionais e não funcionais. Conterem os diagramas de caso de uso e sua descrição, diagramas de atividades desenvolvidos na metodologia da *Unified Modeling Language* (UML). E por último descreve-se o processo de implementação e os principais softwares utilizados.

3.1 DESCRIÇÃO DO SISTEMA

A ferramenta proposta neste trabalho dará ao gerente de suporte maior controle sobre o funcionamento de seu sistema. Isso porque lhe fornecerá informações de problemas recorrentes, e também receberá informações dos problemas no momento em que eles ocorrem. Esta ferramenta será parte integrante de um SAD, gerando informações que depois de tratadas auxiliaram na tomada de decisão, no que diz respeito ao ambiente tecnológico da empresa.

O SAD implementado por esta ferramenta visa gerar informações do desempenho do sistema, que irá abastecer os bancos de dados de informações internas. Utilizando o conceito de FAD, irá tratar as informações que serão úteis para identificar possíveis problemas e traçar melhorias no ambiente, com a ajuda de gráficos.

Tanto o gerente, que será o administrador do sistema, quanto os operadores de *checklist* serão cadastrados na ferramenta e terão um código de usuário. Esse código será utilizado para fazer o *login* na ferramenta.

Para efetuar o cadastro, serão solicitadas as seguintes informações: nome do usuário, código de usuário, *e-mail* do usuário, nível de acesso, que será representado por 0 e 1, e senha.

O usuário de nível 0 terá acesso a todas as funcionalidades da ferramenta, como cadastrar outros usuários e definir seus níveis, alterar cadastros, consultar usuários cadastrados apresentando número, código, nome e nível. Este usuário deverá cadastrar os procedimentos padrões para solucionar as possíveis falhas para cada serviço do *checklist*.

Estes usuários também terão a opção de gerar relatórios gráficos. O usuário administrador receberá um *e-mail* informando quando algum serviço apresentou problema.

No *checklist* de validação constam os itens compõem o ambiente e que foram levantados pelos analistas de suporte. O monitoramento será feito pelo acompanhamento destes itens, onde o operador deverá validá-los e então preencher selecionar no *checklist* qual o status, “Sim” caso esteja funcionando corretamente e “Não” caso apresente problema.

O usuário de nível 1 será a pessoa responsável pelo processo de monitoramento do ambiente de TI, que realizará o acompanhamento diário e preencherá o *checklist*, com os processos e serviços definidos como principais da empresa. Quando uma das opções for marcada com erro, será apresentado um campo com possíveis soluções e procedimentos a serem seguidos para solução do problema. Caso não tenha sido resolvido, deverá marcar como não resolvido, e com isso será disparado um *e-mail* para o administrador. Também terá acesso à consulta de usuários.

Será possível gerar relatórios com a quantidade de problemas por processo, horários com maior número de problemas.

A figura 6 apresenta o fluxo de preenchimento do *checklist* de validação do ambiente, onde são demonstradas todas as atividades possíveis para efetuar este procedimento.

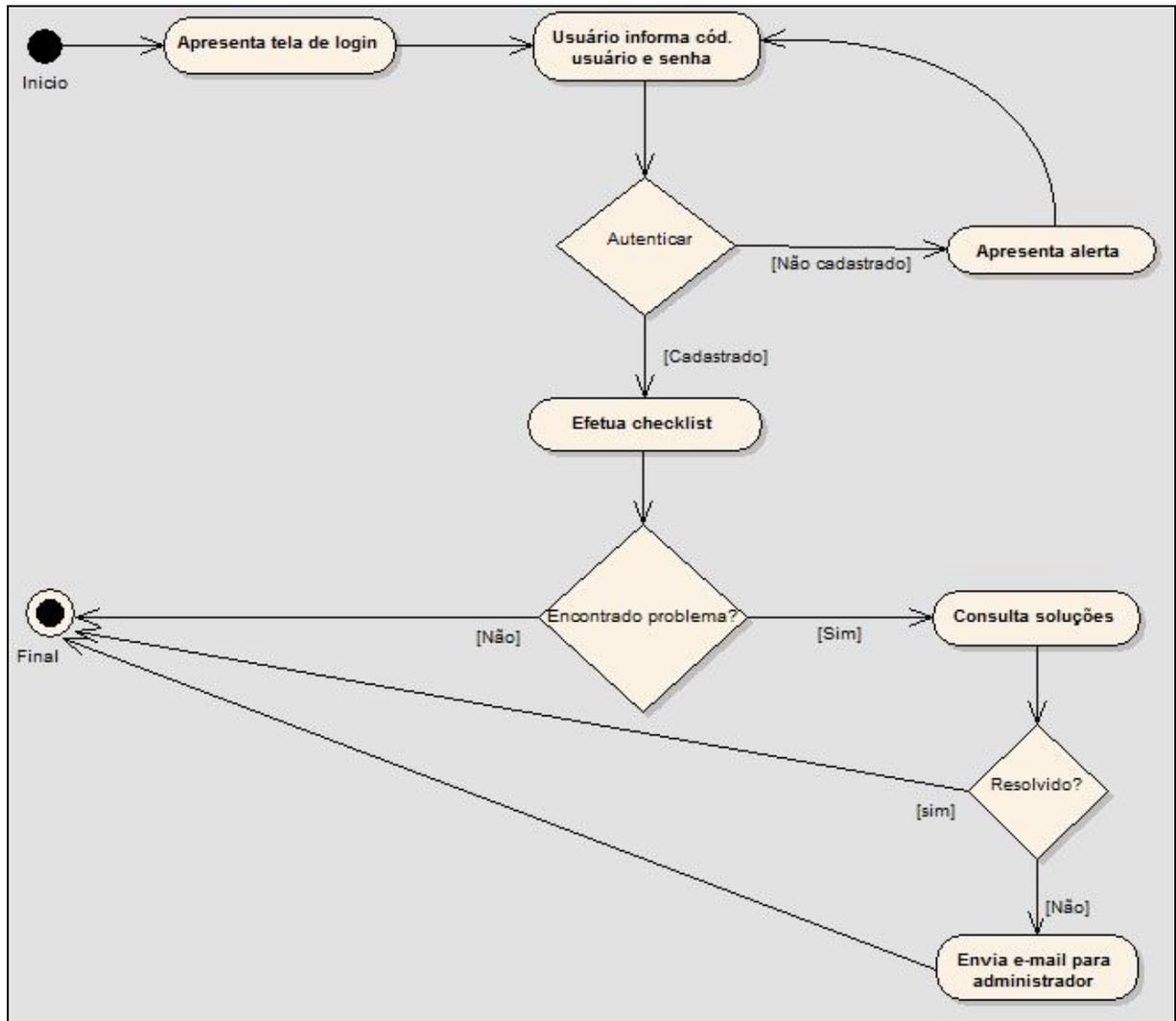


Figura 6 - Fluxo de preenchimento do *checklist*

3.2 REQUISITOS DO SISTEMA

Conforme a descrição do sistema, baseando-se no levantamento de informações e necessidades junto à empresa, definiu-se os requisitos funcionais e não funcionais para a ferramenta. O Quadro 1 apresenta os requisitos funcionais previstos para a ferramenta e sua rastreabilidade, ou seja, vinculação com os casos de uso associados.

Requisitos Funcionais	Caso de Uso
RF01: A ferramenta deverá ser acessada após efetuar <i>login</i> .	UC01
RF02: A ferramenta deverá permitir ao administrador o cadastro de usuários (inclusão, alteração, exclusão) com níveis.	UC02
RF03: A ferramenta deverá permitir a consulta de usuários.	UC03
RF04: A ferramenta deverá permitir ao administrador o cadastro de soluções.	UC04
RF05: A ferramenta deverá permitir a consulta de soluções.	UC05
RF06: A ferramenta deverá permitir a emissão de relatório de problemas por item do <i>checklist</i> .	UC06
RF07: A ferramenta deverá permitir a emissão de relatório de problemas por horas. Gráfico	UC07
RF08: A ferramenta deverá enviar um <i>e-mail</i> quando algum serviço apresentar problema e não for solucionado.	UC08
RF09: A ferramenta deverá permitir o preenchimento do <i>checklist</i> .	UC09

Quadro 1 - Requisitos funcionais

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

Requisitos Não Funcionais
RNF01: A ferramenta deverá utilizar JSP como linguagem de desenvolvimento.
RNF02: A ferramenta deverá ser executada no Internet Explorer 7 ou Firefox 4.
RNF03: A ferramenta deverá utilizar banco de dados MySql.
RNF04: A ferramenta deverá utilizar como servidor <i>web</i> o Apache Tomcat.

Quadro 2 - Requisitos não funcionais

As regras do negócio identificadas para o desenvolvimento da ferramenta incluem:

- a) somente o administrador poderá cadastrar os usuários da ferramenta (RN01);
- b) a ferramenta não deverá permitir o acesso quando for digitado o endereço da página e o usuário não ter feito *login* na sessão. (RN02);
- c) o usuário administrador deverá receber os *e-mails* de alerta quando algum evento for marcado com opção não (RN03);
- d) se o evento com problema for marcado como solucionado, a ferramenta não deverá enviar o *e-mail* (RN04);

- e) a ferramenta não permitirá o cadastro de usuário caso algum dos seguintes não esteja preenchido: nome do usuário, código de usuário, *e-mail*, nível e senha, o sistema deverá apresentar a mensagem informando que é necessário informar os dados (RN05);
- f) tempo da seção deverá ser de 20 minutos (RN06);
- g) os usuários nível 1 não poderão alterar os dados cadastrais (RN07);
- h) somente os usuários nível 0 poderão cadastrar soluções (RN08);
- i) a consulta de usuários deverá listar 10 entradas por página (RN09);
- j) a ferramenta deverá permitir a consulta parcial dos usuários (RN10);
- k) se um item do *checklist* já possuir um roteiro de solução cadastrado, a ferramenta não permitirá que seja adicionado novamente, apenas atualizado (RN11).

3.3 ESPECIFICAÇÃO

Esta seção apresenta os diagramas elaborados para o sistema. A ferramenta Enterprise Architect (EA) foi utilizada na geração dos diagramas através da linguagem de modelagem UML.

3.3.1 Diagramas de caso de uso

Esta subseção apresenta os diagramas de casos de uso da ferramenta, sendo que os detalhes dos casos de uso estão descritos no Apêndice A. Na figura 7, tem-se o diagrama de caso de uso do usuário administrador.

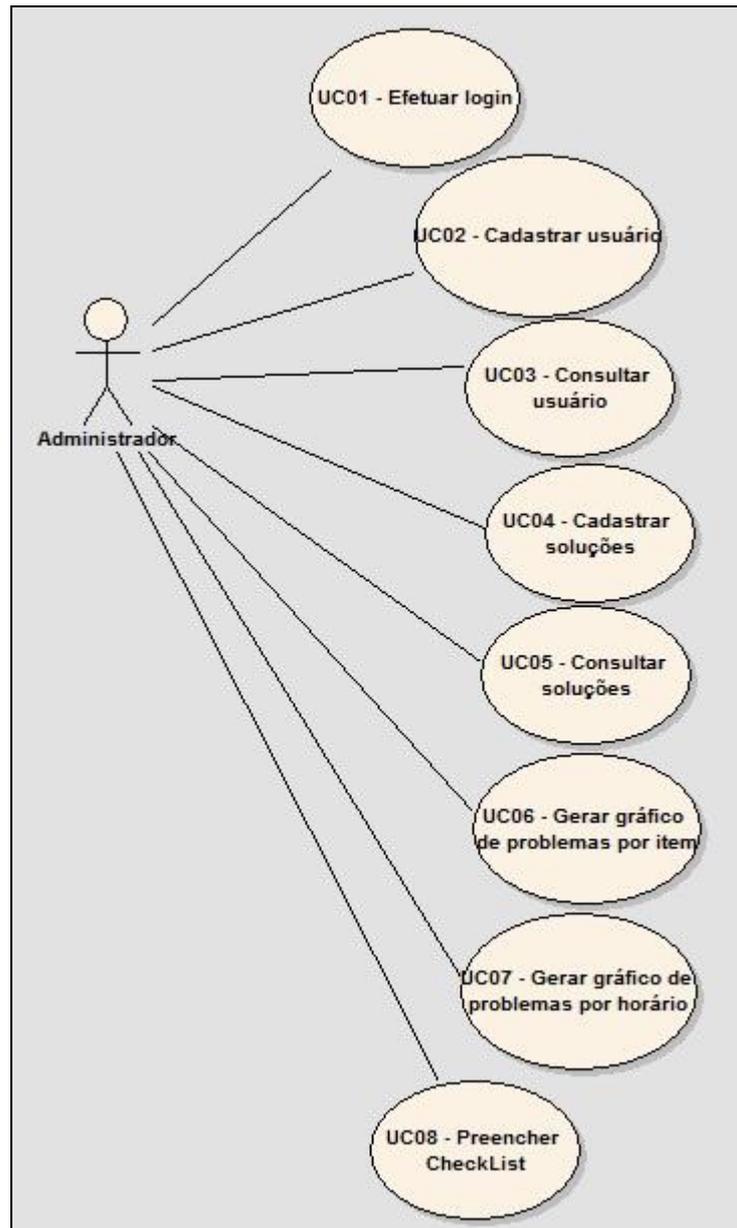


Figura 7 - Diagrama de caso de uso Administrador

Na figura 8, tem-se o diagrama de caso de uso referente ao usuário funcionário.

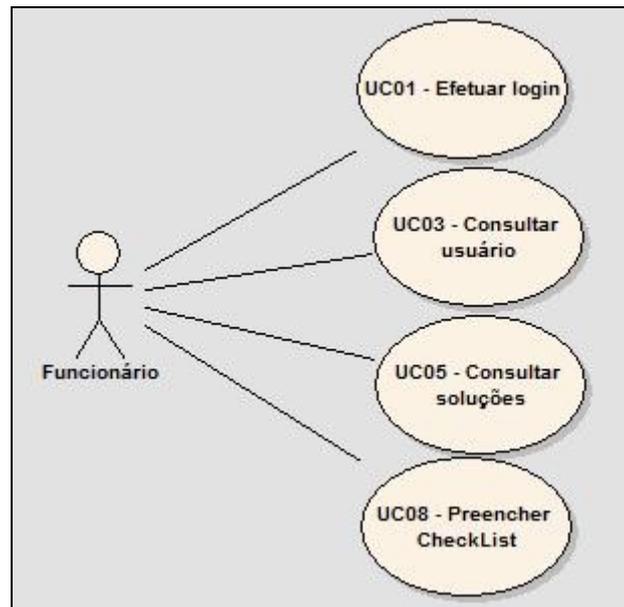


Figura 8 - Diagrama de caso de uso Funcionário

3.3.2 Diagrama Entidade Relacionamento

O Diagrama Entidade Relacionamento (DER) é um modelo em forma de diagrama que descreve o modelo de dados de um sistema com alto nível de abstração. Sua maior aplicação é visualizar o relacionamento entre tabelas de um banco de dados. Pode-se observar o DER da ferramenta na figura 9.

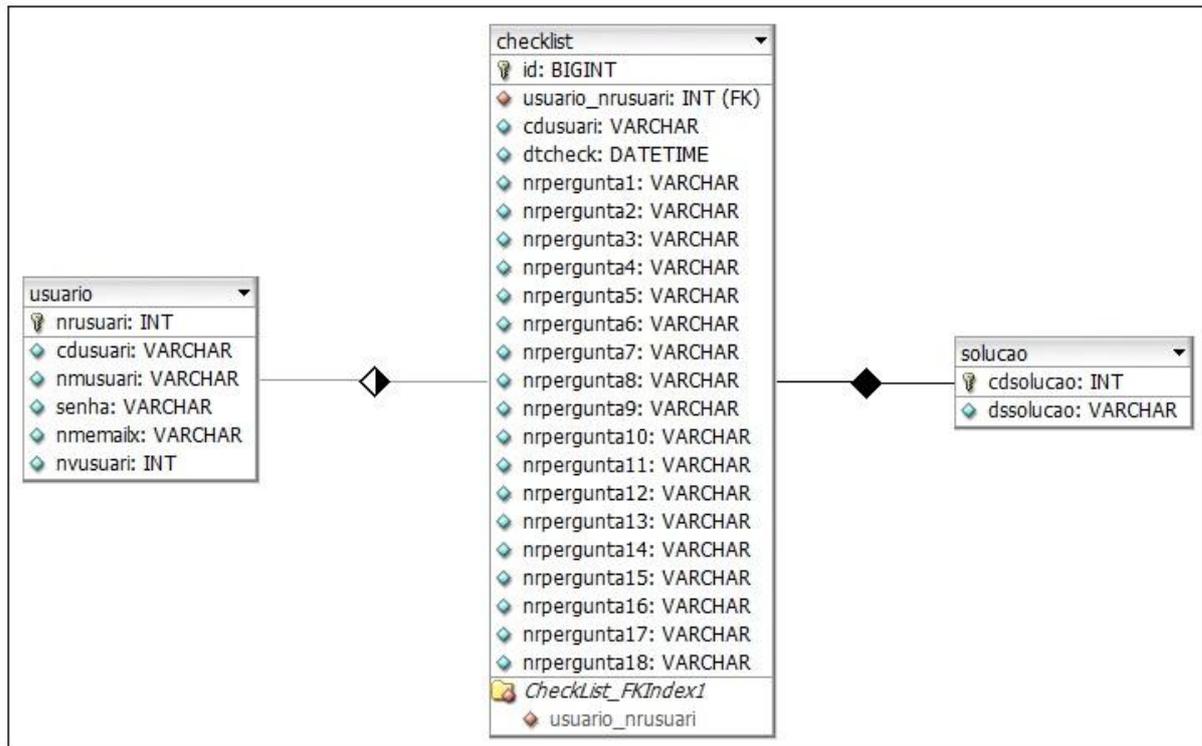


Figura 9 - Diagrama Entidade Relacionamento (DER)

3.3.3 Dicionário de Dados

A seguir os quadros apresentam o dicionário de dados das tabelas da ferramenta. O quadro 3 contém o dicionário de dados da tabela “usuário”.

Tabela: usuario				
Tabela responsável pelo armazenamento dos dados dos usuários da ferramenta.				
Campos:				
Nome	Tipo	Tamanho	Descrição	Obrigatório
nrusuari	Integer	10	Chave primária da tabela de usuários ao qual armazena um número identificador do usuário.	Sim
cdusuari	Varchar	45	Armazena o código do usuário.	Sim
nmusuari	Varchar	45	Armazena o nome do usuário.	Sim
senha	Varchar	45	Armazena a senha do usuário.	Sim
nmemailx	Varchar	45	Armazena o e-mail do usuário.	Sim
nvusuari	Integer	10	Armazena “0” para usuário administrador e “1” para funcionário.	Sim

Quadro 3 – Dicionário de dados da tabela “usuário”

No quadro 4 tem-se o dicionário de dados da tabela “*checklist*”.

Tabela: <i>checklist</i>				
Tabela responsável pelo armazenamento dos dados do <i>checklist</i> de validação.				
Campos:				
Nome	Tipo	Tamanho	Descrição	Obrigatório
Id	Integer	10	Chave primária da tabela de usuários ao qual armazena um número identificador do usuário	Sim
cdusuari	Varchar	45	Armazena o código do usuário.	Sim
dtcheck	Datetime		Armazena a data e hora que foi efetuado o <i>checklist</i> .	Sim
nrpergunta1	Varchar	1	Armazena “S” caso item esteja funcionando e “N” se não estiver.	Sim
nrpergunta2	Varchar	1	Armazena “S” caso item esteja funcionando e “N” se não estiver.	Sim
nrpergunta3	Varchar	1	Armazena “S” caso item esteja funcionando e “N” se não estiver.	Sim
nrpergunta4	Varchar	1	Armazena “S” caso item esteja funcionando e “N” se não estiver.	Sim
nrpergunta5	Varchar	1	Armazena “S” caso item esteja funcionando e “N” se não estiver.	Sim
nrpergunta6	Varchar	1	Armazena “S” caso item esteja funcionando e “N” se não estiver.	Sim
nrpergunta7	Varchar	1	Armazena “S” caso item esteja funcionando e “N” se não estiver.	Sim
nrpergunta8	Varchar	1	Armazena “S” caso item esteja funcionando e “N” se não estiver.	Sim
nrpergunta9	Varchar	1	Armazena “S” caso item esteja funcionando e “N” se não estiver.	Sim
nrpergunta10	Varchar	1	Armazena “S” caso item esteja funcionando e “N” se não estiver.	Sim
nrpergunta11	Varchar	1	Armazena “S” caso item esteja funcionando e “N” se não estiver.	Sim
nrpergunta12	Varchar	1	Armazena “S” caso item esteja funcionando e “N” se não estiver.	Sim
nrpergunta13	Varchar	1	Armazena “S” caso item esteja funcionando e “N” se não estiver.	Sim
nrpergunta14	Varchar	1	Armazena “S” caso item esteja funcionando e “N” se não estiver.	Sim
nrpergunta15	Varchar	1	Armazena “S” caso item esteja funcionando e “N” se não estiver.	Sim
nrpergunta16	Varchar	1	Armazena “S” caso item esteja funcionando e “N” se não estiver.	Sim
nrpergunta17	Varchar	1	Armazena “S” caso item esteja funcionando e “N” se não estiver.	Sim
nrpergunta18	Varchar	1	Armazena “S” caso item esteja funcionando e “N” se não estiver.	Sim

Quadro 4 – Dicionário de dados da tabela “*checklist*”

No quadro 5 tem-se o dicionário de dados da tabela “solução”.

Tabela: solucao				
Tabela responsável pelo armazenamento dos dados das soluções para cada item do <i>checklist</i> .				
Campos:				
Nome	Tipo	Tamanho	Descrição	Obrigatório
cdsolucao	Integer	10	Chave primária da tabela de soluções ao qual armazena um número identificador da solução.	Sim
dssolucao	Varchar	500	Armazena o roteiro de solução para cada item do <i>checklist</i> .	Sim

Quadro 5 – Dicionário de dados da tabela “solução”

3.4 IMPLEMENTAÇÃO

Nesta seção serão tratados os aspectos da implementação da ferramenta de gerenciamento de eventos, com a descrição das técnicas e ferramentas utilizadas e a operacionalidade da implementação e resultados alcançados.

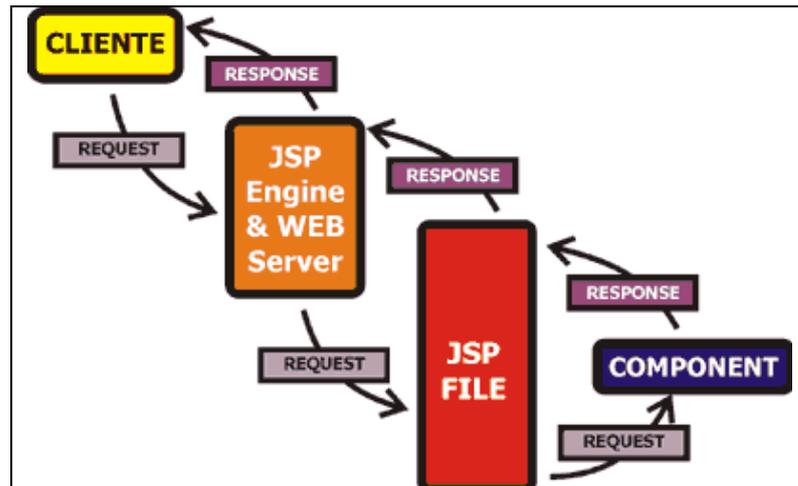
3.4.1 Técnicas e ferramentas utilizadas

No desenvolvimento deste trabalho foram utilizadas várias ferramentas e técnicas, que serão descritas a seguir.

Para desenvolvimento da aplicação foi utilizado o programa NetBeans IDE 6.7.1 que permite a criação de códigos da linguagem de desenvolvimento JSP e utilizado o servidor *web* Apache Tomcat 6.

De acordo com Delfes (2008), *Java Server Pages* (JSP) é uma tecnologia para desenvolvimento de aplicações *WEB* com portabilidade de plataforma podendo ser executada em vários sistemas operacionais. O JSP oferece uma vantagem por ser facilmente codificado, assim tornando mais simples a elaboração e a manutenção da aplicação.

Outra característica do JSP é produzir conteúdos dinâmicos que possam ser reutilizados. Quando a página é requisitada pelo cliente através do navegador, esta página será executada pelo servidor, e a partir daí será gerada uma página HTML que será devolvida ao navegador do cliente. A figura 10 mostra esse funcionamento.



Fonte: Delfes (2008).

Figura 10 - Funcionamento da requisição em JSP

As páginas da ferramenta foram estruturadas em *Hyper Text Markup Language* (HTML) e também a linguagem de estilos *Cascading Style Sheets* (CSS). Utilizou-se também a linguagem *JavaScript* para o tratamento dos componentes da página. Toda a construção dos formulários foi feita utilizando os objetos disponíveis na própria linguagem HTML como, por exemplo, *input button* (botão), o *input text* (caixa de texto de uma linha), o *input hidden* (campo oculto).

Xavier (2009), explica a linguagem HTML como uma linguagem de marcação (*tags*), ou seja, o navegador irá ler as marcações, interpretar e então gerar as formas de acordo com o que foi interpretado das marcações.

O sistema gerenciador de banco de dados utilizado foi o MySQL 5.1. Segundo Lima (2003), MySQL consome poucos recursos do sistema operacional, possui extrema velocidade e compatibilidade.

Segundo Niederauer e Prates (2006), o MySQL é um Sistema Gerenciador de Banco de Dados (SGBD) relacional que utiliza a linguagem padrão *Structured Query Language* (SQL) e é largamente utilizado em aplicações para a internet. Por se tratar de um banco com código-fonte aberto ele é muito popular em seu uso e também de grande atração porque seu custo é bastante baixo e pode ser usada como alternativa se comparado a grandes bancos de dados que têm custos muito mais elevados.

Na figura 11 pode-se observar um exemplo do código JSP implementado. Este código faz inserção do primeiro item do *checklist* no banco de dados.

```

<?
if (sdtptime.equals("ATU")==true) {
    String connectionURL = "jdbc:mysql://localhost:3306/rafaeldb";

    Class.forName("com.mysql.jdbc.Driver");
    Connection connection = DriverManager.getConnection(connectionURL, "root", "");
    Statement statement = connection.createStatement();
    String sql = "INSERT INTO CHECKLIST (CDUSUÁRIO, DTCHECK, NRPERGUNTA1, NRPERGUNTA2, NRPERGUNTA3, NRPERGUNTA4, NRPERGUNTA5, NRPERGUNTA6, NRPERGUNTA7, NRPERGUNTA8, NRPERGUNTA9, NRPERGUNTA10) VALUES ('" + usuario + "', now(),";
    sql += "('' + usuario + "', now(),";

    SendMail mail = new SendMail();

    /*
     *Pergunta 1 - Temperatura
     */

    String sdcheck00s = request.getParameter("check00s");
    if (sdcheck00s != null) {
        sql += "'s',";
    }
    String sdcheck00e = request.getParameter("check00e");
    if (sdcheck00e != null) {
        sql += "'n',";
    }
    String sdcheck00n = request.getParameter("check00n");
    if (sdcheck00n != null) {
        sql += "'n',";
        mail = new SendMail();
        try {
            mail.sendMail("echecklist@gmail.com", "echecklist@gmail.com", "Novo evento CheckList", "Temperatura");
        } catch (Exception e) {
            e.printStackTrace();
        }
    }
    if (sdcheck00s == null && sdcheck00n == null && sdcheck00e == null) {
        sql += "'s',";
    }
}

```

Figura 11 - Exemplo código fonte

3.4.2 Operacionalidade da implementação

A seguir, para melhor entendimento da ferramenta, as telas serão apresentadas através de estudo de caso baseado em dados fictícios.

Por se tratar de uma ferramenta de auxílio, na tela de *logon* não há nenhuma informação ou propaganda, será apenas solicitado o usuário e senha para *login*. Esta página será igual para todos os usuários conforme é apresentada na figura 12.

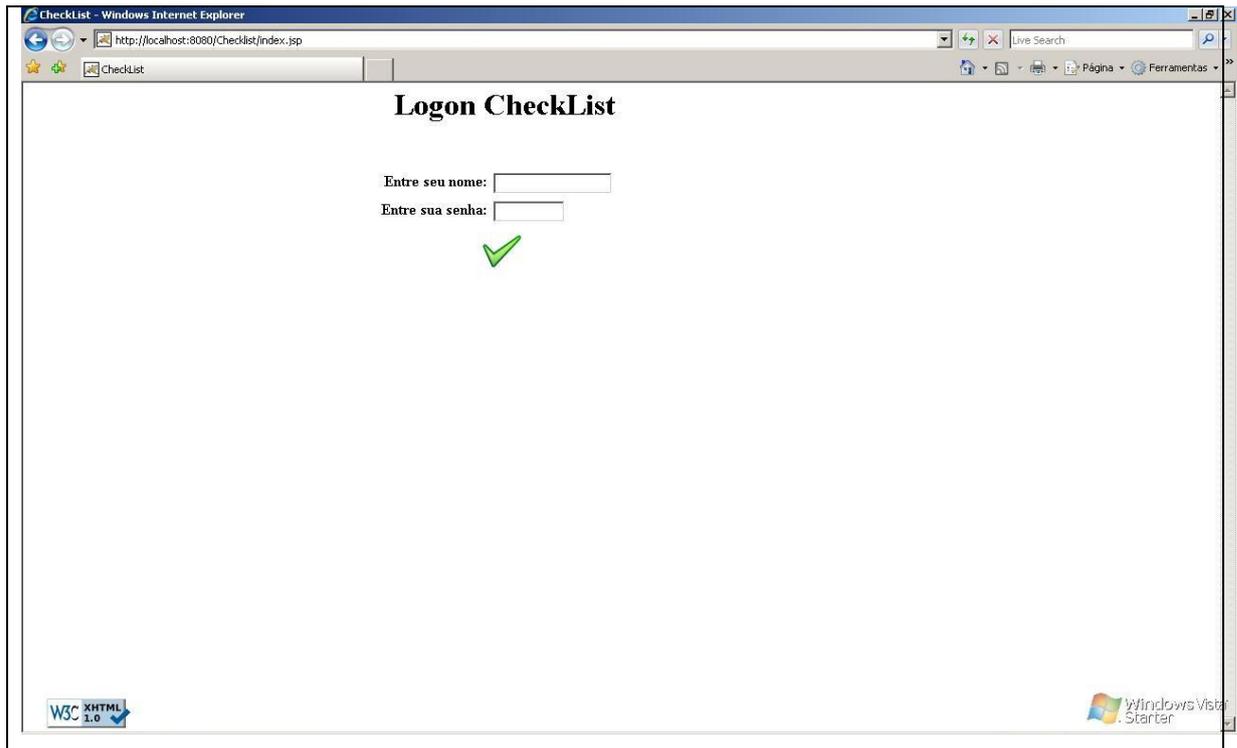


Figura 12 - Logon Checklist

Se não for informado o código de usuário, é apresentada uma mensagem solicitando as informações, conforme figura 13.

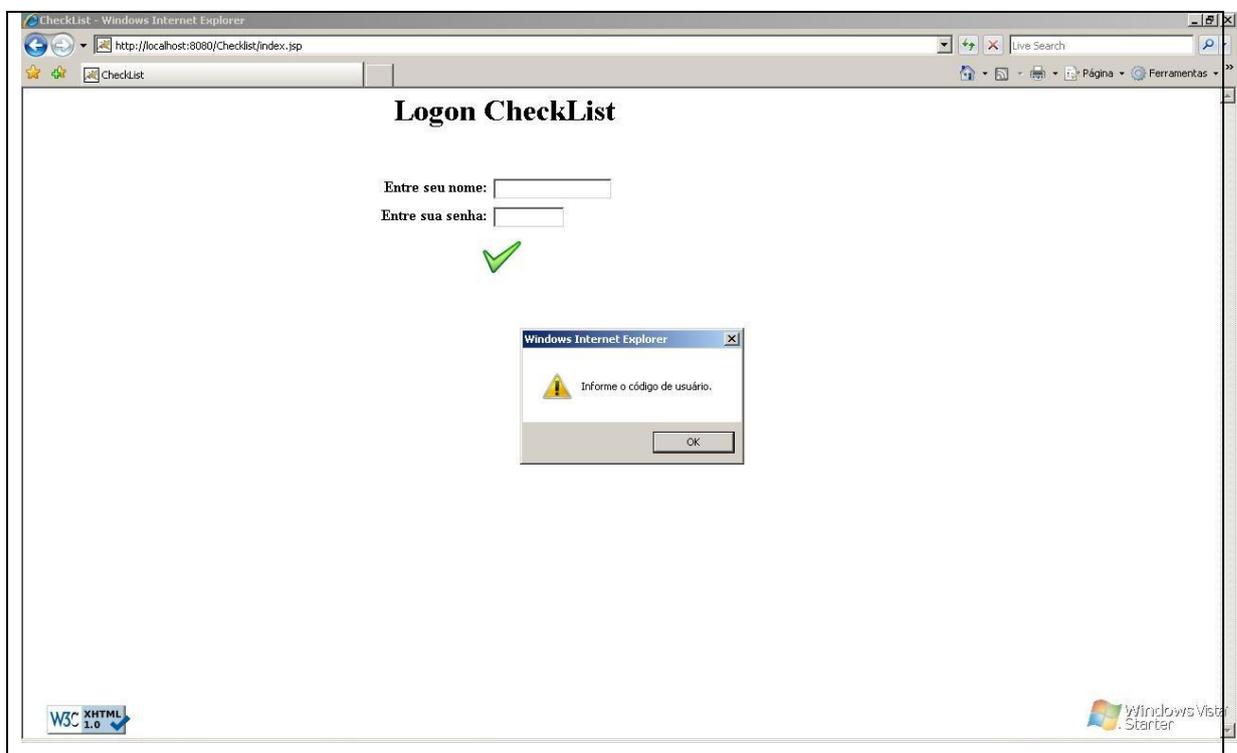


Figura 13 - Código de usuário não informado

Após a verificação de usuário e senha a ferramenta apresenta a tela inicial com as opções de menu que o tipo de usuário tiver liberado. A figura 14 apresenta os itens do menu. Seguido a sequencia temos o botão “Inicio”, que volta para o menu principal, “Cadastro novo usuário”, onde será feito os cadastros dos usuários. “Consulta de usuários”, para efetuar as consultas dos usuários cadastrados no sistema. “Cadastro de soluções”, para efetuar os cadastros das rotinas de soluções para os itens do *checklist*. “Acessar checklist”, que contem o *checklist* com os itens para validação. “Relatórios”, que terá os gráficos que serão gerados com os dados de funcionamento do ambiente.



Figura 14 - Itens do menu

Após efetuar o *login* na ferramenta, pode-se utilizá-la conforme o nível de liberação. Para efetuar as manutenções dos usuários da ferramenta, deve-se escolher a opção “Cadastrar novo usuário” que está localizado no menu de opções. Nesta tela é possível cadastrar, alterar e excluir as contas de usuários, conforme figura 15.

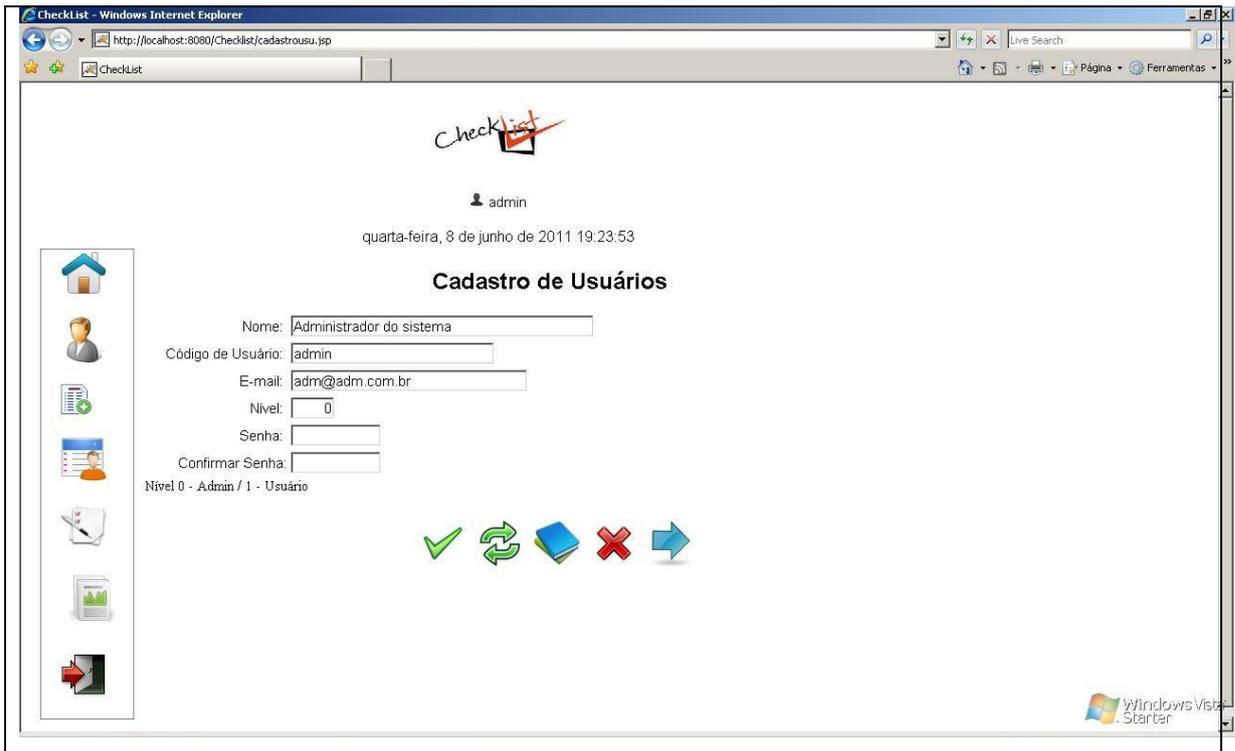


Figura 15 - Cadastrar novo usuário

Todos os campos são obrigatórios. Para garantir isso, o sistema informa através de mensagens na própria página o campo que faltou preencher. Esse tratamento de exceção foi feito através de roteiros *javascript*, conforme figura 16.



Figura 16 - Verificação dos campos obrigatórios

A figura 17 apresenta o trecho do código *javascript* que faz a verificações dos campos do cadastro de usuários.

```

/*Quando axtpmanute for igual a 'ADD' (adicionar)*/
if (axtpmanute=='ADD') {
/*Validacao dos campos da pagina de cadastro de usuarios*/
  if (formulario.nmusuari.value=="") {
    msg = "Nome do usuário deve ser informado.\n"+msg
  }
  if (formulario.cdusuari.value=="") {
    msg = "Código do usuário deve ser informado.\n"+msg
  }
  if (formulario.nmemailx.value=="") {
    msg = "E-mail do usuário deve ser informado.\n"+msg
  }
  if (formulario.nvusuari.value=="") {
    msg = "Nível do usuário deve ser informado.\n"+msg
  }
  if (formulario.nvusuari.value > "1") {
    msg = "Nível do usuário pode ser 0 ou 1.\n"+msg
  }
  if (formulario.dssenha1.value=="") {
    msg = "Senha deve ser informada.\n"+msg
  }
  /*Se o campo "dssenha1" for informado, pede para confirmar a senha
  informando o campo "dssenha2"*/
  if (formulario.dssenha1.value!="") {
    if (formulario.dssenha2.value=="") {
      msg = "Senha deve ser confirmada.\n"+msg
    }
  }
}

```

Figura 17 - Código *javascript* da verificação dos campo

Para cadastrar os roteiros de soluções de cada item do *checklist*, deve-se escolher a opção “Cadastrar solução” que está localizado no menu de opções. Nesta tela é possível cadastrar e alterar os roteiros de soluções, conforme figura 18.

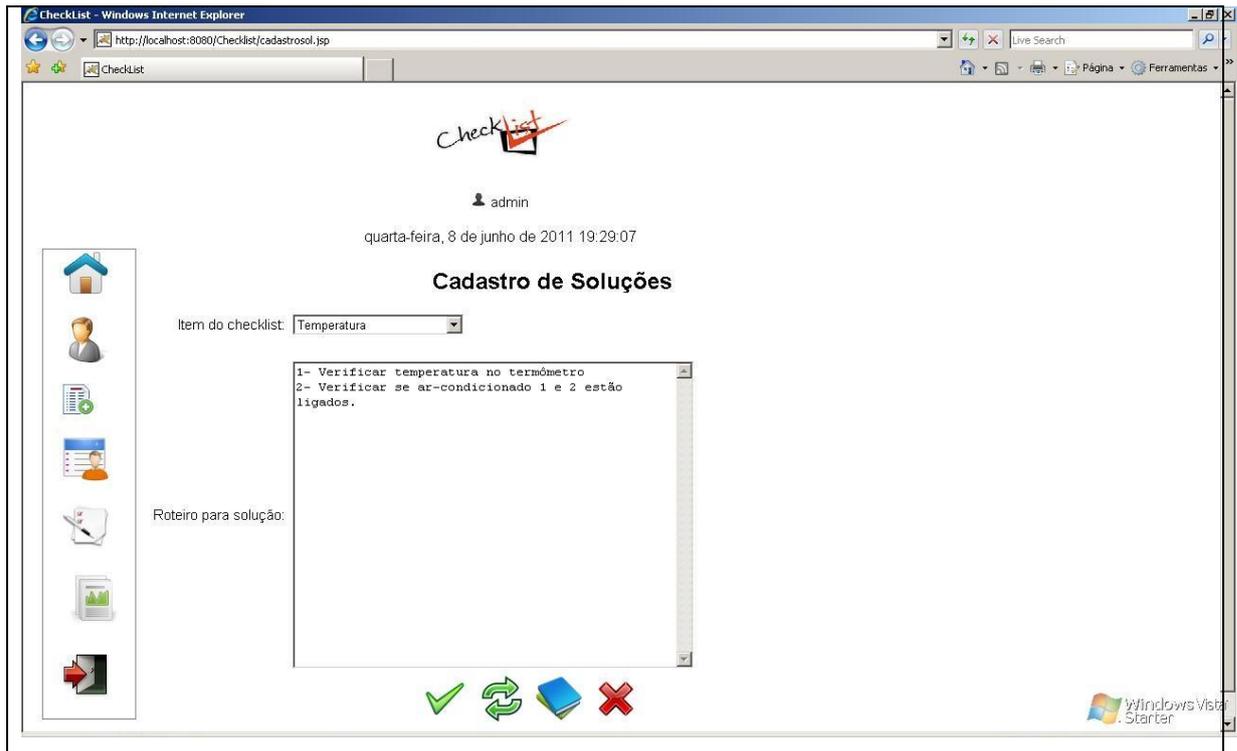


Figura 18 - Cadastro de soluções

Para efetuar consulta de usuários cadastrados na ferramenta, deve-se escolher a opção “Consultar usuários” que está localizado no menu de opções. Nesta tela não há campos obrigatórios: se nenhum campo for preenchido a consulta trará todos os usuários cadastrados, listando 10 entradas por página. A consulta também pode ser feita utilizando o nome ou código de usuário, ou parte deles, conforme figura 19 e 20.

Nome:

Código de Usuário:

Nr. do usuário	Código do usuário	Nome do usuário	Nível
1	admin	Administrador do sistema	0
4	rafael.pires	Rafael Antonio Pires	0
3	nivel0	Teste Nivel 0	0
2	nivel1	Teste Nivel 1	1
5	nivel1a	Teste Nivel 1 a	1
6	nivel1b	Teste Nivel 1 b	1
7	nivel1c	Teste Nivel 1 c	1
8	nivel1d	Teste Nivel 1 d	1
9	nivel1e	Teste Nivel 1 e	1
10	nivel1f	Teste Nivel 1 f	1

Figura 19 - Consulta de usuários

Nome:

Código de Usuário:

Nr. do usuário	Código do usuário	Nome do usuário	Nível
1	admin	Administrador do sistema	0
4	rafael.pires	Rafael Antonio Pires	0

Figura 20 - Consulta de usuário parcial

Para efetuar o *checklist*, deve-se escolher a opção “Acessar checklist” que está localizado no menu de opções, conforme figura 21. O usuário deverá apenas preencher “Sim” e “Não”, a data e hora será inserida a do momento da confirmação.

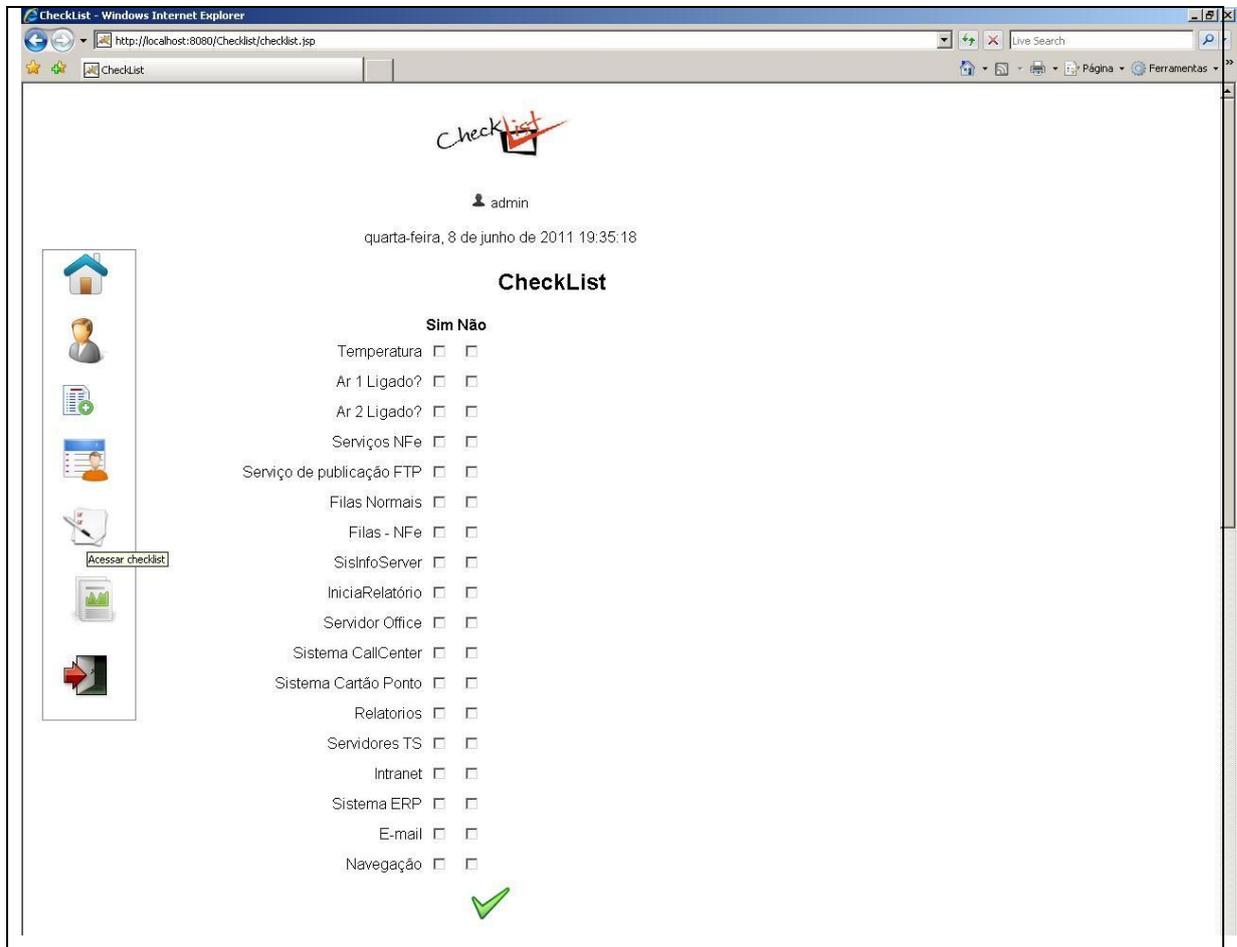


Figura 21 - Itens Checklist

Conforme figura anterior os campos “Sim” e “Não” do checklist vem desmarcados. Após o operador efetuar a verificação do item, ao preencher só é possível marcar uma das opções. Caso a opção “Não” seja marcada, serão apresentadas outras opções. Primeiro tem-se o botão que mostrará o roteiro de solução para o item em questão. Caso o problema seja resolvido deve-se marcar a opção “Resolvido?”, conforme figura 22.

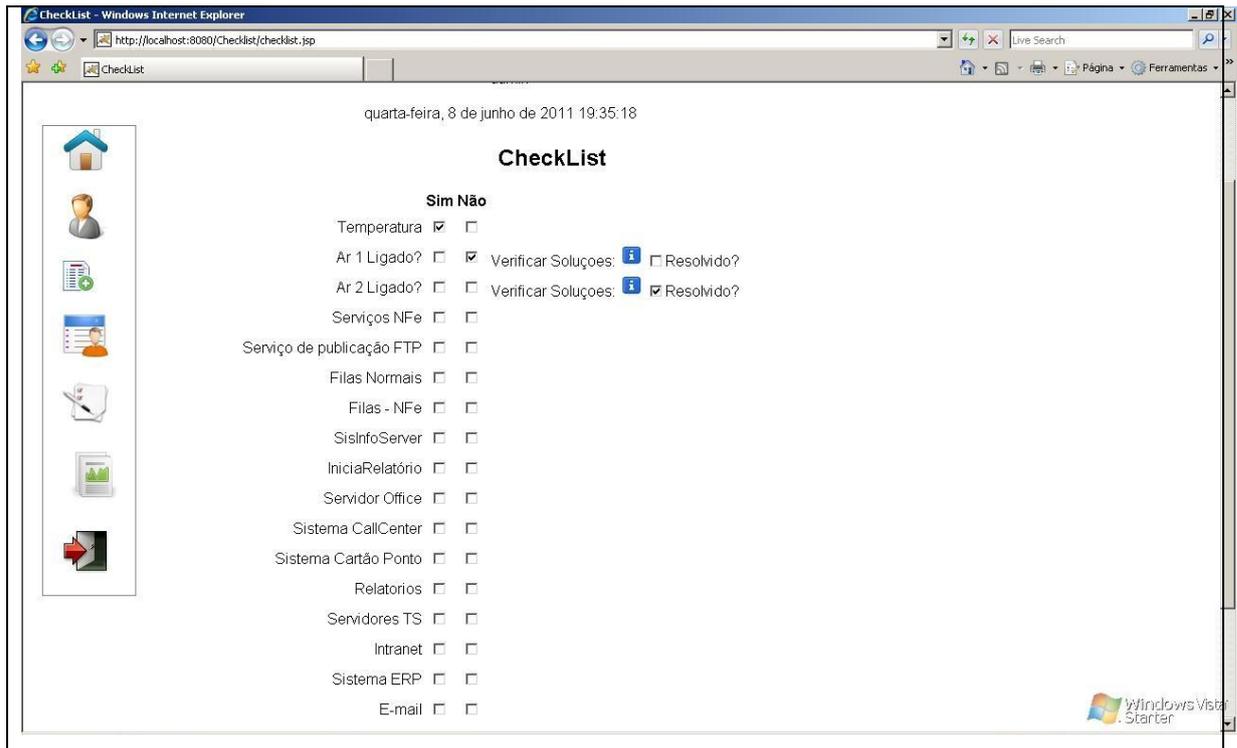


Figura 22 - Opções de *check*

Quando um dos itens apresentarem problema e não ter sido resolvido, a ferramenta irá enviar uma mensagem de alerta para o *e-mail* cadastrado para o recebimento. Na figura 23 tem-se um *e-mail* que foi recebido.

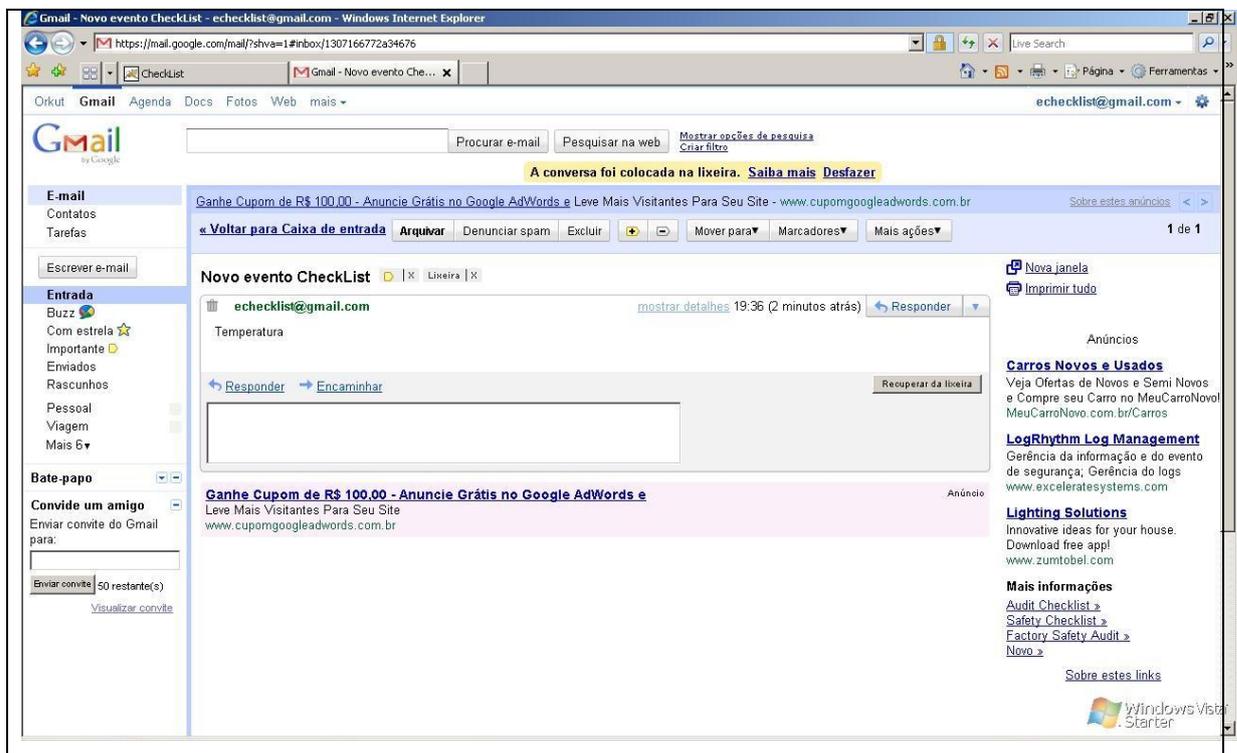


Figura 23 - *E-mail* recebido

Os usuários nível 0 poderão gerar relatórios estatísticos dos problemas ocorridos e que foram marcados nos *checklists*. Para isso deve-se escolher a opção “Relatórios” que está localizado no menu de opções. Esta tela dará a opção de gráfico de pizza e de colunas. O gráfico de pizza mostrará os problemas ocorridos por item do *checklist* no período definido, conforme figura 24.

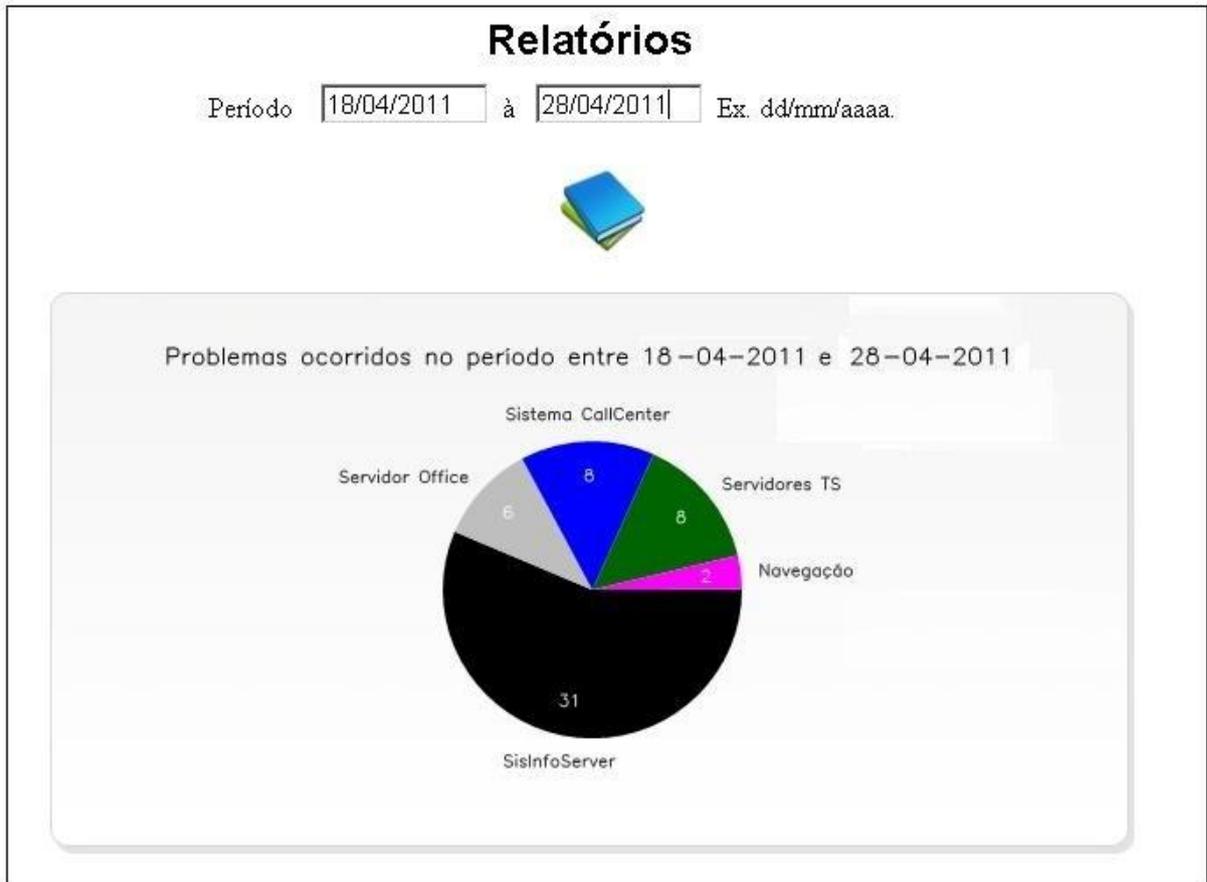


Figura 24 - Gráfico pizza problemas por itens

O gráfico de colunas mostrará os problemas ocorridos por horário no período definido, conforme figura 25.

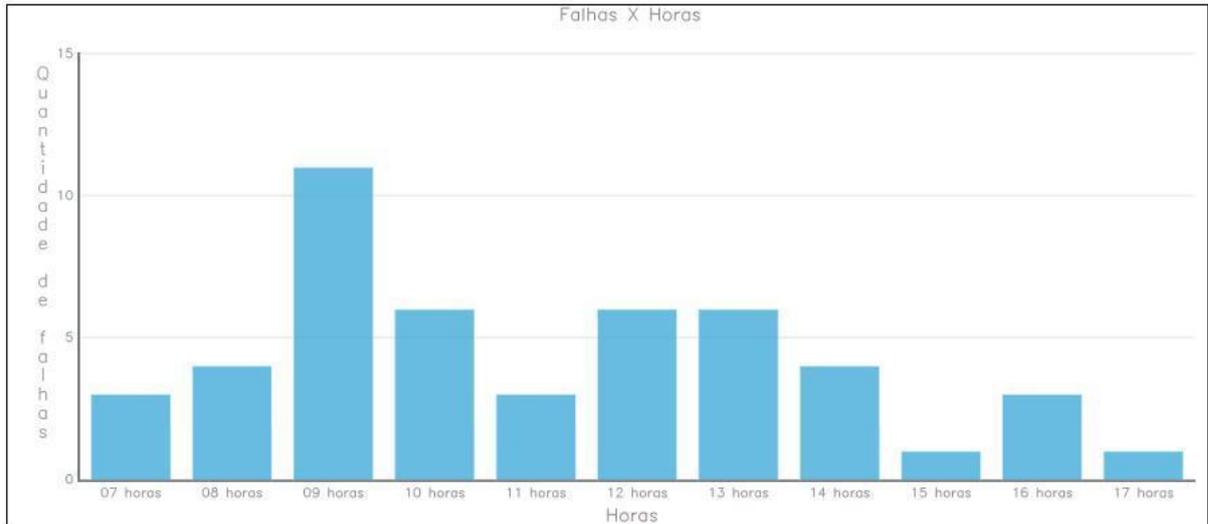


Figura 25 - Gráfico coluna problemas por horário

O tempo de sessão da ferramenta é de vinte minutos, caso o usuário fique inativo neste tempo, será necessário efetuar novamente o *logon*. Figura 26 mostra a tela de acesso negado.



Figura 26 - Acesso negado

3.5 RESULTADOS E DISCUSSÃO

O desenvolvimento deste trabalho foi concluído com sucesso, e o objetivo que era originalmente desenvolver uma ferramenta *web* que possibilitasse o controle do ambiente de TI foi alcançado. Todos os requisitos levantados foram desenvolvidos. Os resultados obtidos através do trabalho foram muito satisfatórios nos quesitos gerenciamento de ambiente TI genéricos e agilidade para resposta a um eventual problema, pois com uma abordagem proativa, pode-se identificar falhas antes que elas ocorram. Outro item que irá tornar o processo de solução mais ágil é o roteiro de solução, pois com isso o operador da validação conseguirá resolver alguns problemas imediatamente, sem a necessidade da intervenção de um analista de suporte experiente.

Não foi possível testar a ferramenta no ambiente da empresa devido algumas políticas internas. Diante disto, foram solicitados os arquivos dos *checklist* antigos para importá-los na ferramenta. Foram importados os dados dos meses de novembro e dezembro de 2010. Após a conclusão da importação a ferramenta foi mostrada para o coordenador da área de suporte, que gostou da ferramenta e das funcionalidades, como o envio de *e-mails* e o roteiro de solução. Para um próximo projeto da empresa, a ferramenta será adequada ao ambiente e ao sistema gerencia que hoje existe e é desenvolvido internamente.

A ferramenta, além do benefício de poder dimensionar os problemas do ambiente de TI, evita as falhas que eram causadas no controle do *checklist* feito no papel, devido ao grande volume gerado pelas folhas de papel e pastas de arquivos que não eram analisados.

Nos trabalhos correlatos, foi apresentado o software *Microsoft Operations Manager* que é bem completo para uma gestão de infraestrutura de TI, mas é complexo e possuem várias opções que não seriam usadas em uma empresa de pequeno ou médio porte. Pode-se citar também seu valor, que pode ser proibitivo para o orçamento de muitas organizações. Ele também não atende a todas as funções, como o controle serviços de catracas, temperatura do CPD e alguns softwares terceirizados.

A utilização de um ambiente *web* com desenvolvimento de páginas em linguagem JSP e o banco MySQL, fez com que os custos do sistema sejam acessíveis a várias empresa de pequeno a médio porte. Os relatórios são específicos e de fácil entendimento para o gerente que controlará os ambientes.

4 CONCLUSÕES

O presente trabalho pretende colaborar com o conceito abordado na fundamentação teórica, desenvolvendo uma ferramenta responsável pelo controle e monitoramento genérico das funcionalidades de um ambiente de TI, automatizando o processo de validação, dando uma resposta rápida a solução de problemas e apresentando informações referentes ao ambiente. Por se tratar de uma ferramenta genérica, o custo se tornou mais baixo, sendo acessível à grande maioria das empresas de pequeno e médio porte.

O trabalho foi concluído com sucesso, atingindo todos os seus objetivos iniciais, com o desenvolvimento de uma ferramenta de informação *web* que possibilitasse o controle e monitoramento dos diversos serviços e processos da área de TI e sendo parte de um SAD. Em relação aos objetivos específicos pode-se afirmar que com o controle e monitoramento corretos, pode-se garantir a qualidade no serviço prestado pela área de suporte da empresa. Com o preenchimento do *checklist* é possível validar as diversas funcionalidades do ambiente de TI, e com as informações geradas a partir destas validações tem-se dados que ajudarão o administrador de suporte a traçar melhorias para o ambiente.

Por fim, este trabalho veio a engrandecer em termos de conhecimentos pessoais, sobre controle e monitoramento de sistemas, pois trabalho na área de suporte, e este assunto é um problema atual. Sobre as linguagens de programação para a *web*, tinha-se pouco conhecimento, principalmente em JSP e teve-se que aprender e também superar as dificuldades encontradas para realização e conclusão deste trabalho. O aprendizado de uma linguagem de programação exigiu um esforço maior nas pesquisas à internet e aos livros, e que proporcionou o conhecimento necessário para o desenvolvimento da ferramenta.

4.1 EXTENSÕES

Para trabalhos futuros, sugere-se implementar a criação de seqüência de validações do *checklist*, com isso a responsabilidade de adicionar os itens passaria para o administrador quando houvesse necessidade, em casos de novas funcionalidades implantadas no ambiente.

Outra funcionalidade que poderia ser acrescentada à ferramenta, é a criação de rotinas de validação em tempo real, onde diminuiria ainda mais o tempo de resposta aos problemas.

REFERÊNCIAS BIBLIOGRÁFICAS

ATTIE, William. **Auditoria: conceitos e aplicações**. 2. ed. São Paulo: Atlas, 1986.

BARBOSA JUNIOR, Euler de A.. **Sistemas de Informação**. São Paulo, [2009?]. Disponível em: <http://guiadoestudante.abril.com.br/profissoes/profissoes_280113.shtml>. Acesso em: 30 out. 2010.

CHAVES, Eduardo. **Sistemas de Informação e Sistemas de Apoio a Decisão**. [S.l.], 2004. Disponível em <<http://www.chaves.com.br/textself/comput/sad.htm>>. Acesso em: 25 abr. 2011.

CYBERCONN BCLM SUITE. **Gerenciamento de riscos**. [S.l.], [2007?]. Disponível em: <<http://www.cyberconn.com/SOFTWARE/Governan%C3%A7aRiscoeConformidade/tabid/148/Default.aspx>>. Acesso em: 27 ago. 2010.

DELFE, Jaider. **Linguagem de Programação II – Desenvolvimento Web – Java**. [S.l.], [2008?] Disponível em: <http://www.fortium.com.br/faculdefortium.com.br/jaider_delfes/material/5191.pdf>. Acesso em: 16 abr. 2011.

HEINZLE, Roberto. **Um modelo de engenharia do conhecimento para sistemas de apoio a decisão com recursos para raciocínio abdutivo**. Florianópolis, 2011. Disponível em: <http://btd.egc.ufsc.br/wp-content/uploads/2011/05/Roberto_Heinzle.pdf>. Acesso em: 20 mai. 2011.

HILSDORF, Carlos. **51 Atitudes essenciais para vencer na vida e na carreira**. ed. Carlos Hilsdorf – ed. Laselva Negócios, [S.l.], 2009.

LAUDON, K. C.; LAUDON, J. P. **Sistemas de informação gerenciais: administrando a empresa digital**. 5. ed. São Paulo: Prentice Hall, 2004.

LIMA, Adilson da Silva. **MySQL Server: Versões Open Source 4.X: Soluções para desenvolvedores e administradores de bancos de dados**. 1. Ed. São Paulo: Érica, 2003.

LUIZ, Vanessa Andréia. **Software de Controle e Gerenciamento de Documentos de Projetos**. Blumenau, 2009. Disponível em: <<http://www.inf.furb.br/~pericas/orientacoes/PMBOK2009.pdf>>. Acesso em: 27 ago. 2010.

MACORATTI, José Carlos. **Desenvolvendo para desktop ou para WEB?**. [S.l.], [2010?]. Disponível em: <http://www.macoratti.net/vbn_dkw.htm>. Acesso em: 12 mar. 2011.

MARQUES, Cleber F. **MOF: Monitoramento e Controle de Serviços – SMF (Service Monitoring and Control)**. [S.l.], 2007. Disponível em:
<<http://www.linhadecodigo.com.br/artigo/1517/mof-monitoramento-e-contr.aspx>>. Acesso em: 30 out. 2010.

MICROSOFT. **Descrição geral do Microsoft Operations Manager 2005**. [S.l.], 2004. Disponível em:
<<http://www.microsoft.com/portugal/mom/evaluation/overview/default.msp>>. Acesso em: 20 set. 2010.

MOF 4. **Microsoft Operations Framework 4**. [S.l.], 2008. Disponível em:
<<http://technet.microsoft.com/en-us/library/cc506049.aspx>>. Acesso em: 30 out. 2010.

NETWORK WORLD. **Monitoramento de aplicações vira aliado dos departamentos de TI**. [S.l.], 2010. Disponível em:
<<http://computerworld.uol.com.br/tecnologia/2010/09/27/monitoramento-de-aplicacoes-vira-aliado-dos-departamentos-de-ti/>>. Acesso em: 29 mar. 2011.

NIEDERAUER, Juliano; PRATES, Rubens. **Guia de consulta rápida Mysql 5**. São Paulo: Novatec, 2006. Disponível em:
<<http://www.martinsfontespaulista.com.br/anexos/produtos/capitulos/209813.pdf>>. Acesso em: 16 abr. 2011.

O'BRIEN, J. A. **Sistemas de informação: e as decisões gerenciais na era da Internet**. 3. ed. São Paulo: Saraiva 2004.

OLIVEIRA, D. P. R. **Sistemas de informações gerenciais: estratégica, tática, operacionais**. 5. ed. São Paulo: Atlas 1998.

ORACLE. **Getting Started with Web Applications**. [S.l.], [2006?]. Disponível em:
<<http://download.oracle.com/javae/1.4/tutorial/doc/WebApp.html>>. Acesso em: 20 set. 2010.

ORLANDINI, Leandro. **Administração e Tecnologia**. [S.l.], 2005. Disponível em:
<http://www.bonde.com.br/bonde.php?id_bonde=1-14--1646-20050407>. Acesso em: 30 out. 2010.

PEREIRA, Alex de Oliveira. **Fundamentos em Gerenciamento de Serviços de TI**. [S.l.], [2007?]. Disponível em:
<<http://pareaki.com/Gerenciamentos%20de%20Servicos%20TI.pdf>>. Acesso em: 9 ago. 2010.

POWER, D.J. **A Brief History of Decision Support Systems**. [S.l.], [2002]. Disponível em:
<<http://DSSResources.com/history/dsshhistory.html>>. Acesso em: 15 mai. 2011.

SEGUNDO, Fábio. **Programação para Web: JSP**. Apostila da disciplina de Programação *WEB*, FURB. Blumenau, 2008.

WATSON, Hugh J. **Sistemas de Apoio à Decisão**. Michigan: Ed. Irwin, 1991.

TOMAEL, Maria Inês. **Fontes de informação na internet**. [S.l.], [2008?]. Disponível em: <<http://snbu.bvs.br/snbu2000/docs/pt/doc/t138.doc>>. Acesso em: 10 nov. 2010.

TURBAN, Efrain; MCLEAN, Ephraim; WETHERBE, James. **Tecnologia da Informação para gestão**. 3 ed. Porto Alegre: Bookman, 2004.

UFRN. **Sistemas de informação**. Rio grande do Norte, 2007. Disponível em: <<http://pt.scribd.com/doc/4419063/Sistemas-de-Infomacao-Unidade-01-747>>. Acesso em: 29 mar. 2011.

XAVIER, Denys William. **Estrutura básica HTML**. [S.I.] 2009. Disponível em: <<http://www.tiexpert.net/ver.php?page=85>>. Acesso em: 16 abr. 2011.

APÊNDICE A – Detalhamento dos casos de uso.

No Quadro 6 apresenta-se o caso de uso "Efetuar Login" (UC01).

Caso de uso – Efetuar login

Ator: Usuário

Objetivo: Efetuar autenticação na ferramenta

Pré-condições: A ferramenta deve estar hospedada no servidor *web*, Usuário deve estar cadastrado no banco de dados.

Pós-condições: Usuário ferramenta liberada para o uso.

Cenário Principal:

1. Usuário preenche seu login e sua senha
2. A ferramenta valida os dados de login e senha do usuário
3. A ferramenta direciona o Usuário para a página inicial

Cenário Alternativo:

No passo 1, caso seja informado nome de usuário e/ou senha inválidos

- 1.1 Ferramenta apresenta um alerta de usuário ou senha inválido.
- 1.2 Usuário informa dados corretos
- 1.3 Volta ao cenário principal

Quadro 6 – Descrição do caso de uso Efetuar Login

No Quadro 7 apresenta-se o caso de uso "Cadastrar usuário" (UC02).

Caso de uso – Cadastrar usuário

Ator: Usuário nível 0

Objetivo: Efetuar cadastro de usuários na ferramenta

Pré-condições: A ferramenta deve estar hospedada no servidor *web*, usuário deve estar logado.

Pós-condições: Novo usuário cadastrado.

Cenário Principal:

1. Usuário seleciona a opção “cadastrar novo usuário”
2. Usuário informa todos os dados
3. Usuário informa o nível do usuário da ferramenta
4. A ferramenta valida informações do cadastro
5. A ferramenta armazena os dados

Cenário Alternativo:

No passo 4, caso algum dado não seja informado ou cadastre com mesmo código

- 1.1 Ferramenta apresenta um alerta de quais dados faltam ser informados.
- 1.2 Usuário informa dados corretos
- 1.3 Volta ao cenário principal

Quadro 7 – Descrição do caso de uso Cadastrar de usuário

No Quadro 8 apresenta-se o caso de uso "Consultar usuário" (UC03).

Caso de uso – Consultar usuário

Ator: Usuário

Objetivo: Efetuar consulta de usuários

Pré-condições: A ferramenta deve estar hospedada no servidor *web*, Usuário deve estar cadastrado no banco de dados.

Pós-condições: Lista com os usuários cadastrados.

Cenário Principal:

1. Usuário seleciona a opção "Consulta de usuário"
2. Usuário informa parte do nome ou deixa em branco para consulta completa

Quadro 8 – Descrição do caso de uso Consultar usuário

No Quadro 9 apresenta-se o caso de uso "Cadastrar soluções" (UC04).

Caso de uso – Cadastrar soluções

Ator: Usuário nível 0

Objetivo: Efetuar cadastro de rotinas soluções na ferramenta

Pré-condições: A ferramenta deve estar hospedada no servidor *web*, usuário deve estar logado.

Pós-condições: Cadastro das rotinas de solução para consultas posteriores.

Cenário Principal:

1. Usuário seleciona a opção "cadastrar soluções"
2. Usuário seleciona o item do *checklist*
3. Usuário preenche o roteiro de solução
4. A ferramenta armazena os dados

Quadro 9 – Descrição do caso de uso Cadastrar soluções

No Quadro 10 apresenta-se o caso de uso "Emitir relatórios" (UC06).

Caso de uso – Emitir relatório

Ator: Usuário nível 0

Objetivo: Gerar gráficos com os indicativos de falhas do ambiente

Pré-condições: A ferramenta deve estar hospedada no servidor *web*, usuário deve estar logado.

Pós-condições: Apresenta gráficos.

Cenário Principal:

1. Usuário seleciona a opção "Relatórios"
2. Usuário informa o período desejado para consulta
3. A ferramenta apresenta os gráficos de falhas por problemas e de falhas por hora.

Quadro 10 – Descrição do caso de uso Emitir relatórios

No Quadro 11 apresenta-se o caso de uso "Enviar *e-mail*" (UC07).

Caso de uso – Enviar *e-mail*

Ator: Ferramenta

Objetivo: Enviar *e-mail* caso algum item validado apresente problema e não seja resolvido

Pré-condições: A ferramenta deve estar hospedada no servidor *web*, usuário deve estar logado, algum item do *checklist* deve apresentar problema.

Pós-condições: Administrados recebe *e-mail* com informando o item com problema.

Cenário Principal:

1. Usuário preenche *check list*;
2. Usuário transmite *check list*;
3. A ferramenta verifica se alguma opção foi selecionada com a opção não e sem solução e envia um *e-mail* para o usuário Administrador

Quadro 11 – Descrição do caso de uso Envio de *e-mail*

No Quadro 12 apresenta-se o caso de uso "Preencher *check list*" (UC08).

Caso de uso – Preencher *checklist*

Ator: Usuário

Objetivo: Efetuar a validação do ambiente através do *checklist*

Pré-condições: A ferramenta deve estar hospedada no servidor *web*, usuário deve estar logado.

Pós-condições: Ambiente validado

Cenário Principal:

- 1 Usuário verifica no *checklist* quais os itens deverá validar;
- 2 Usuário deverá selecionar a opção sim, se o serviço estiver funcionando, e não, caso não esteja funcionando corretamente;
- 3 Usuário clica em atualizar;

Cenário Alternativo:

No passo 2, caso seja selecionado a opção não

2.1 Habilita um campo com as possíveis soluções ou procedimentos para aquele determinado serviço

2.2 Se o problema não for resolvido, deve-se selecionar opção "Resolvido?"

2.3 A ferramenta enviará um *e-mail* caso não seja resolvido o problema

2.4 Voltar ao cenário principal

Quadro 12 – Descrição do caso de uso Preencher *check list*