

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

**APLICAÇÃO DA TÉCNICA RBC NO SISTEMA DE
INFORMAÇÃO PARA SUPORTE AOS CLIENTES DA
EMPRESA MULTITHERM SISTEMAS E AUTOMAÇÃO
LTDA.**

JOÃO RODRIGO DA SILVA

BLUMENAU
2011

2011/1-12

JOÃO RODRIGO DA SILVA

**APLICAÇÃO DA TÉCNICA RBC NO SISTEMA DE
INFORMAÇÃO PARA SUPORTE AOS CLIENTES DA
EMPRESA MULTITHERM SISTEMAS E AUTOMAÇÃO
LTDA.**

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. Roberto Heinzle, Doutor – Orientador

**BLUMENAU
2011**

2011/1-12

**APLICAÇÃO DA TÉCNICA RBC NO SISTEMA DE
INFORMAÇÃO PARA SUPORTE AOS CLIENTES DA
EMPRESA MULTITHERM SISTEMAS E AUTOMAÇÃO
LTDA.**

Por

JOÃO RODRIGO DA SILVA

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

Presidente: _____
Prof. Roberto Heinzle, Doutor – Orientador, FURB

Membro: _____
Prof. Antonio Carlos Tavares, Mestre – FURB

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

Blumenau, 30 de junho de 2011.

Dedico este trabalho a toda minha família, e a todas as pessoas que direta ou indiretamente me ajudaram na realização deste.

AGRADECIMENTOS

À minha família, que sempre deu o apoio e incentivo necessário na formação acadêmica juntamente com a formação humana.

Aos meus amigos, pelos empurrões, cobranças e por estarem ao meu lado sempre quando necessário.

Ao meu orientador, Roberto Heinzle, pelas importantes contribuições, advertências e sugestões.

A todos que, direta ou indiretamente me auxiliaram na concretização deste trabalho, meus sinceros agradecimentos.

Noventa por cento do sucesso se baseia simplesmente em insistir.

Woody Allen

RESUMO

Este trabalho de conclusão de curso tem por objetivo integrar a tecnologia de Raciocínio Baseado em Casos utilizando a técnica do vizinho mais próximo (*Nearest Neighbour*), em um sistema de informação utilizado pelo setor de suporte ao cliente da empresa Multitherm Sistemas e Automação. Sistemas RBC simulam o ato humano de relembrar um episódio para resolver um determinado problema através da identificação de similaridades entre os mesmos, para tanto se utilizou uma fórmula matemática para cálculo desta similaridade. Esta integração visa facilitar o trabalho, organizar e armazenar o conhecimento, diminuir o tempo total dos atendimentos e fornecer soluções mais precisas aos novos problemas relatados.

Palavras-chave: Sistemas de informação. Raciocínio baseado em casos. Inteligência artificial. Conhecimento.

ABSTRACT

This work aims to integrate the technology Case-Based Reasoning using the technique of Nearest Neighbor in a information system used by the customer support sector of the company Multitherm Sistemas e Automação. CBR systems simulate the human act of remembering a previous episode in order to solve a determinate problem according the identification of the affinities between them, it's used a mathematics formula to calculate this similarity. This integration is able to make the job easier, improve storing and organizing knowledge data base, reduce the customer support time and provide more accurate solutions to the problems.

Key-words: Information systems, Case-based reasoning, Artificial intelligence, Knowledge.

LISTA DE ILUSTRAÇÕES

Figura 1 - Ambiente de um Sistema de Informação.....	14
Figura 2 - Ciclo RBC.....	18
Figura 3 – Cálculo Similaridade.....	22
Figura 4 – Diagrama de Atividades Atual.....	24
Figura 5 – Tela Inicial Sistema Suporte	25
Figura 6 – Tela de Consulta Atual.....	27
Figura 7 – Diagrama de atividades a ser seguido	28
Quadro 1 - Requisitos funcionais	29
Quadro 2 - Requisitos não funcionais	29
Figura 8 – Diagrama de Caso de uso ampliação	30
Figura 9 – Principais casos de uso do sistema.....	31
Figura 10 – Modelo entidade-relacionamento.....	32
Quadro 3 – Dicionário de dados para a tabela “Funcionario”	32
Quadro 4 – Dicionário de dados para a tabela “Atividade”	33
Quadro 5 – Dicionário de dados para a tabela “Pendencia”	33
Quadro 6 – Dicionário de dados para a tabela “Tempo”	34
Quadro 7 – Dicionário de dados para a tabela “Cliente”	34
Quadro 8 – Dicionário de dados para a tabela “Sistema”.....	34
Quadro 9 – Dicionário de dados para a tabela “ClienteSistema”	35
Quadro 10 – Dicionário de dados para a tabela “Atualizacao”	35
Quadro 11 – Dicionário de dados para a tabela “Versao”	35
Quadro 12 – Dicionário de dados para a tabela “Pendencia”.....	36
Quadro 13 – Dicionário de dados para a tabela “Previsao”	36
Quadro 14 – Dicionário de dados para a tabela “PalavraChave”	37
Quadro 15 – Dicionário de dados para a tabela “PalavraChave”	37
Figura 11 – Diagrama de Classes do sistema	39
Quadro 16 – Base de Casos	40
Quadro 17 – Resultados Obtidos	40
Quadro 18 – Comparação entre os casos.....	41
Figura 12– Lista de usuários para login.....	41
Figura 13 – Usuário selecionado para login	42

Figura 14 – Tela inicial do sistema.....	42
Figura 15 – Cadastro de palavras-chave.....	43
Figura 16 – Cadastro de um novo Atendimento.....	44
Figura 17 – Cadastro de Pendência	45
Figura 18 – Confirmar inserção na base de casos ao finalizar	46
Figura 19 - Indexação com palavras-chave	46
Figura 20 – Seleção de palavras-chave para busca RBC.....	47
Figura 21 – Resultado Busca RBC	48
Figura 22 - Cálculo de similaridade do atendimento.....	49
Quadro 19 – Descrição do caso de uso Manter Caso	55
Quadro 20 – Descrição do caso de uso Manter Palavras-Chave	56
Quadro 21 – Descrição do caso de uso Efetuar Busca por Similaridade	56

LISTA DE SIGLAS

ERP – *Enterprise Resourcing Planing*

IA – Inteligência Artificial

RBC – Raciocínio Baseado em Casos

SI – Sistemas de Informação

SUMÁRIO

1. INTRODUÇÃO.....	12
1.1. OBJETIVOS DO TRABALHO	13
1.2. ESTRUTURA DO TRABALHO	13
2. FUNDAMENTAÇÃO TEÓRICA	14
2.1. SISTEMAS DE INFORMAÇÃO.....	14
2.2. TIPOS DE SISTEMAS DE INFORMAÇÃO	15
2.3. RACIOCÍNIO BASEADO EM CASOS.....	17
2.4. DEFINIÇÃO.....	17
2.5. CICLO DE RBC.....	17
2.6. DESENVOLVIMENTO DE UM SISTEMA RBC.....	18
2.6.1. Aquisição do Conhecimento	19
2.6.2. Representação de Casos	19
2.6.3. Indexação	20
2.6.4. Recuperação	20
2.6.5. Método Vizinho mais Próximo.	21
2.6.6. Adaptação dos casos.	22
2.7. SISTEMA ATUAL	23
2.7.1 Área de suporte da Multitherm Sistemas e Automação.	23
2.8. TRABALHOS CORRELATOS	25
3. DESENVOLVIMENTO.....	27
3.1. LEVANTAMENTO DE INFORMAÇÕES	27
3.2. REQUISITOS PRINCIPAIS DO SISTEMA	29
3.3. ESPECIFICAÇÃO	29
3.4. DIAGRAMAS DE CASO DE USO	30
3.5. MODELO CONCEITUAL DE DADOS	31
3.6. IMPLEMENTAÇÃO	37
3.6.1. Técnicas e ferramentas utilizadas.....	38
3.6.2. Aplicação do RBC no Sistema.....	39
3.6.3. Operacionalidade da implementação	41
3.6.4. Estudo de Caso.....	47
3.7. RESULTADOS E DISCUSSÃO	49

4. CONCLUSÕES	51
4.1. EXTENSÕES	51
REFERÊNCIAS BIBLIOGRÁFICAS	53
APÊNDICE A – Detalhamento dos casos de uso.	55

1. INTRODUÇÃO

A Inteligência Artificial (IA) é uma área de pesquisa dedicada a buscar métodos ou dispositivos computacionais que simulem a capacidade humana de resolver problemas, pensar ou, de forma ampla, ser inteligente. A IA adotou a idéia de reproduzir faculdades humanas como criatividade, auto-aperfeiçoamento e uso da linguagem. Segundo Delpizzo (1997), pode-se definir Inteligência Artificial como “o campo de estudo da ciência que persegue a meta de fazer o raciocínio do computador similar ao raciocínio humano” tentando dar explicações algorítmicas aos processos mentais humanos.

O Raciocínio Baseado em Casos (RBC) é uma técnica dentro da Inteligência Artificial que busca resolver novos problemas buscando soluções que já foram utilizadas para resolver problemas anteriores. Segundo Kolodner (1993), o RBC é uma técnica que busca resolver novos problemas adaptando soluções usadas para resolver problemas anteriores. Usa-se uma experiência passada, que é representada e armazenada na forma de casos que, posteriormente são recuperados e adaptados para resolução de um novo problema. A idéia básica em um sistema RBC é que, para um domínio particular, os problemas a serem resolvidos tendem a ser recorrentes e repetir-se com pequenas alterações em relação a sua versão original. Dessa forma, soluções anteriores podem ser reaplicadas também com pequenas modificações.

A utilização de RBC em Sistemas de Informação (SI) de Suporte a Decisão (SAD) tem crescido nos últimos anos, segundo Wangenheim e Wangenheim (2003), o “Raciocínio Baseado em Casos estabeleceu-se nos últimos anos como uma das tecnologias mais populares e disseminadas para o desenvolvimento de Sistemas Baseados em Conhecimento.”

Frente a isto, o presente trabalho propõe a ampliação do atual Sistema de Informação chamado “Suporte”, utilizado pelo setor de suporte na empresa Multitherm Sistemas e Automação, inserindo a técnica de Raciocínio Baseado em Casos. Técnica esta que armazena no sistema o conhecimento de um especialista da área de suporte, possibilitando definir procedimentos a serem executados em função de uma determinada situação. Pretende-se que a aplicação do RBC neste sistema permita encontrar rapidamente soluções, possibilitando uma interação adequada com o usuário. Além disso, RBC suporta a transferência de experiência de atendentes experientes, para novos funcionários sem a necessidade de treinamento, auxiliando na construção da memória corporativa da empresa.

1.1. OBJETIVOS DO TRABALHO

O objetivo deste trabalho é a integração da tecnologia RBC ao sistema de informação utilizado pelo setor de suporte ao usuário da empresa Multitherm Sistemas e Automação aplicando a técnica de RBC. Os objetivos específicos são:

- a) disponibilizar um sistema de apoio à resolução de problemas;
- b) validar o sistema através de um estudo de caso.

1.2. ESTRUTURA DO TRABALHO

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

No segundo capítulo tem-se a fundamentação teórica, onde apresenta-se os conceitos sobre RBC, SI, o sistema atual e trabalhos correlatos.

No terceiro capítulo é apresentado o desenvolvimento da ampliação, incluindo detalhes sobre a especificação, implementação realizada e operacionalidade do mesmo.

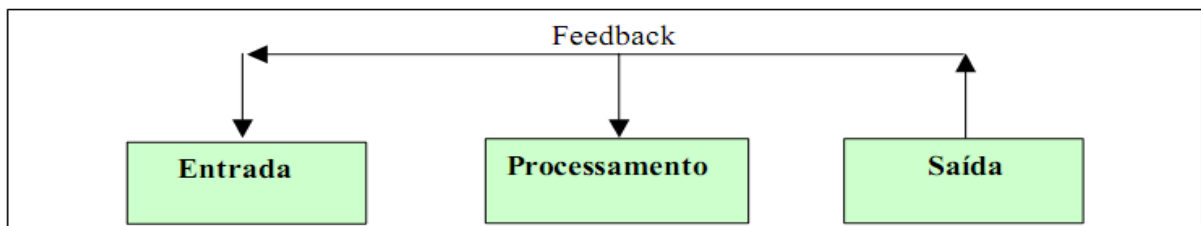
No quarto capítulo tem-se a conclusão e sugestões para trabalhos futuros.

2. FUNDAMENTAÇÃO TEÓRICA

Este capítulo aborda assuntos a serem apresentados nas seções a seguir, tais como os conceitos de Sistemas de Informação, o Raciocínio Baseado em Casos, o sistema atual, além dos trabalhos correlatos.

2.1. SISTEMAS DE INFORMAÇÃO

Um Sistema de Informação (SI) é um sistema cujo principal elemento é a informação. Seu objetivo é armazenar, tratar e fornecer informações de tal modo a apoiar as funções ou processos de uma organização. De acordo com Dalfovo e Amorim (2000), SI são sistemas que permitem a coleta, o armazenamento, o processamento, a recuperação e a disseminação de informações. Na concepção de Oliveira (1996) “é um conjunto de partes integrantes e interdependentes que conjuntamente formam um todo unitário com determinado objetivo e efetuam determinada função”. Para Stair (1998), “é um conjunto de elementos ou componentes que interagem para se atingir objetivos”. O ambiente de um sistema é constituído por entradas, processamento, e saídas conforme mostra Figura 1.



Fonte: Adaptado de Stair (1998).

Figura 1 - Ambiente de um Sistema de Informação

Os componentes de um sistema estão explicados a seguir:

- a) entradas: tudo o que o processador do sistema recebe para processar, armazenar e gerar saídas;
- b) processamento: componentes do sistema que transformam as entradas em saídas podendo efetuar várias operações como juntar, calcular, transformar, armazenar e selecionar. Este processador é a maneira pela qual haja interação no sentido de produzir as saídas desejadas;

- c) saídas: correspondem aos resultados do processamento. É o produto final resultante do processamento das entradas;
- d) *feedback*: processo de comunicação que reage a cada entrada de informação incorporando o resultado da ação reposta desencadeada por meio de nova informação, a qual afetará seu comportamento subsequente e assim sucessivamente. Seu objetivo é reduzir as divergências ao mínimo.

Os objetivos dos SI são de fornecer aos interessados, empreendedores, executivos, informações relacionadas com determinado assunto que interessa a organização no momento. Os SI são hoje elemento indispensável para dar apoio às operações e à tomada de decisão na empresa, antes de seu surgimento as empresas utilizavam uma grande quantidade de mão de obra e tempo quando necessitavam levantar um grande número de informação.

2.2. TIPOS DE SISTEMAS DE INFORMAÇÃO

Existem várias propostas de classificação de SI as quais estão relacionadas sua importância na organização. Entre várias propostas destacam-se as seguintes:

De acordo com Dalfovo e Amorim (2000), os SI podem ser divididos em quatro categorias de acordo com o nível em que atuam:

- a) sistemas de informação em nível operacional: são os sistemas que monitoram as atividades elementares e transacionais da organização. Sendo que o propósito é de responder a questão de rotina e fluxo de transações, como por exemplo, vendas, recibos, depósito de dinheiro e folha;
- b) sistemas de informação em nível de conhecimento: são os SI de suporte a funcionários especializados e de dados em uma organização. O propósito destes sistemas é ajudar a empresa a integrar novos conhecimentos ao negócio e ajudar a organização a controlar o fluxo de papéis, que são os trabalhos burocráticos. Fazem parte desta categoria os SI de Tarefas Especializadas e os Sistemas de Automação de Escritórios;
- c) sistemas de informação em nível administrativo: são os SI que suportam o monitoramento de administradores em nível médio, o propósito do sistema deste nível é controlar e prover informações para a direção setorial de rotina. Os SI

Gerenciais são um tipo de sistema que fazem parte desta categoria;

- d) sistemas de informação em nível estratégico: são os SI que suportam a atividade de planejamento de longo prazo dos administradores seniores. Sendo que seu propósito é compatibilizar mudanças no ambiente externo com as capacidades organizacionais existentes. Os Sistemas de Informações Executivas são um tipo de sistema que faz parte desta categoria.

Segundo Laudon e Laudon (1999), como existem diferentes interesses, especialidades e níveis em uma organização, existem diferentes tipos de sistemas, e por isto propõe a seguinte classificação:

- a) sistemas de processamento de transações (SPT): são os sistemas empresariais básicos que servem o nível operacional da organização;
- b) sistemas de trabalho do conhecimento (STC) e de Automação de Escritório (SAE): suprem as necessidades de informação no nível de conhecimento da empresa. Os sistemas de trabalho do conhecimento ajudam os trabalhadores do conhecimento, enquanto os sistemas de automação de escritório ajudam principalmente os trabalhadores de dados;
- c) sistemas de informação gerenciais (SIG): dão suporte ao nível gerencial da empresa, proporcionando aos gerentes relatórios e, em alguns casos, acesso on-line para o desempenho atual e registros históricos da empresa;
- d) sistemas de suporte de decisões (SSD): também dão suporte ao nível gerencial da organização;
- e) sistemas de suporte executivo (SSE): gerentes seniores usam uma categoria de sistemas de informação chamada “Sistemas de suporte executivo para a tomada de decisão”. Os SSE dão suporte ao nível estratégico da empresa. Eles enfocam decisões não-estruturadas e criam um ambiente computadorizado e de comunicações generalizado, em vez de fornecerem qualquer aplicação fixa ou capacidade específica.

2.3. RACIOCÍNIO BASEADO EM CASOS

O Raciocínio Baseado em Casos (RBC) é uma tecnologia para a solução de problemas e para o aprendizado com base em experiências passadas. De uma forma simplificada, entende-se RBC como a solução de novos problemas por meio da utilização de casos anteriores já conhecidos. Um novo problema que é apresentado é resolvido com a reutilização da solução de um problema anterior similar ao atual.

2.4. DEFINIÇÃO

Segundo Aamodt e Plaza (1994), RBC é uma técnica de IA com enfoque para a solução de problemas e para o aprendizado baseado em experiências passadas. RBC resolve problemas ao recuperar e adaptar experiências passadas. Estas experiências são chamadas de casos, e ficam armazenadas num repositório chamado base de casos. Um novo problema é resolvido com base na adaptação de soluções de casos similares já conhecidos e armazenados.

Para o desenvolvimento de sistemas RBC é necessário inicialmente utilizar soluções já aplicadas anteriormente com sucesso e determinar qual das experiências mais se assemelha ao problema atual. Para isso é necessário que essas experiências passadas sejam analisadas e armazenadas de maneira organizada. O sistema RBC é responsável por realizar a pesquisa nestas experiências armazenadas e verificar se existem casos semelhantes com as características do problema atual, podendo encontrar um ou mais casos, que serão utilizados na resolução do problema atual.

2.5. CICLO DE RBC

O Ciclo de Funcionamento de um sistema de RBC é composto por quatro etapas de execução, conforme definido por Aamodt & Plaza (1994) e explicadas a seguir:

- a) recuperar: a partir da apresentação ao sistema de um novo problema é feita a

- recuperação na base de casos daquele mais parecido com o problema em questão;
- reutilizar: a partir do caso recuperado é feita a reutilização da solução associada ao caso recuperado. Geralmente a solução é transferida ao novo problema diretamente como sua solução;
 - revisar: uma revisão é feita quando a solução não pode ser aplicada diretamente ao novo problema;
 - reter: é o processo de armazenar o novo caso e sua respectiva solução para futuras reutilizações. Deve ser definido qual informação armazenar e de que forma.

Dado um novo problema, a base de casos é pesquisada à procura de tarefas anteriormente resolvidas cuja descrição seja similar à da situação atual.

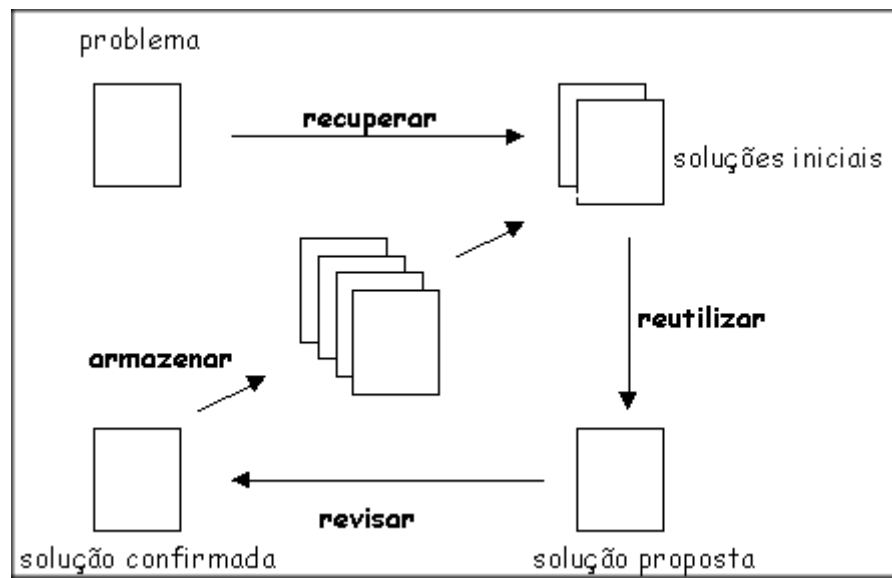


Figura 2 - Ciclo RBC

2.6. DESENVOLVIMENTO DE UM SISTEMA RBC

Conforme Wangenheim e Wangenheim (2003), as etapas mais importantes do processo de desenvolvimento de um sistema RBC são:

- aquisição do Conhecimento;
- representação de Caso;
- indexação;
- recuperação de Casos;
- adaptação de Casos.

2.6.1. Aquisição do Conhecimento

A aquisição do conhecimento é uma etapa que consiste na coleta de dados necessários para formar a base de casos, que contenha implicitamente o conhecimento necessário na resolução de problemas. Conforme Kolodner (1993), ela é considerada o componente crítico no desenvolvimento de sistemas RBC.

2.6.2. Representação de Casos

O conhecimento em um sistema RBC é principalmente armazenado sob a forma de casos. Para Kolodner (1993), um caso é uma peça do conhecimento contextualizado representando um episódio ou experiência em que um problema foi parcialmente ou totalmente resolvido.

Casos podem ser encontrados sob diversas formas e tamanhos, um caso pode ter diferentes conteúdos e representações dependendo da área de aplicação, por exemplo, venda de imóveis, atendimento a cliente.

Um caso representa tipicamente a descrição de uma situação (problema) juntamente com as experiências adquiridas durante a sua resolução (solução). Para Wangenheim e Wangenheim (2003), caso é composto dos seguintes componentes principais:

- a) descrição do problema que foi resolvido ou a situação que foi compreendida;
- b) solução que descreve como o problema foi resolvido em uma situação particular.

Para que os casos estejam à disposição para serem reutilizados, eles são organizados e armazenados em uma base de casos. Geralmente uma base de casos contém experiências descrevendo os passos efetuados para resolver o problema em questão.

2.6.3. Indexação

A indexação determina quais características devem ser comparadas entre os casos para avaliar e quantificar o grau de similaridade, com o propósito de recuperar os casos mais úteis e auxiliar a resolução do problema atual. A indexação deve ser considerada conforme a estrutura e o conteúdo da memória. A memória é indexada para proporcionar uma recuperação e reutilização eficiente, ou seja, se define como índices os atributos que serão utilizados para realizar a comparação entre o caso e a situação presente, sendo estes atributos conhecidos como discriminantes.

Para Kolodner (1993), indexação de casos é a associação de rótulos em casos, de maneira a caracterizá-los para posteriormente recuperá-los em uma base de casos. Portanto a escolha de um índice deve ser realizada cuidadosamente. A seleção dos índices pode ser realizada manualmente ou automaticamente. A seleção manual analisa caso a caso para determinar quais características descritas que determinam as variações sobre as conclusões. A seleção automática busca quantificar as diferenças entre os casos e os relacionamentos entre o problema e soluções adotadas.

Segundo Lee (1998), existe métodos automatizados que auxiliam na escolha de bons índices, porém na prática, os sistemas cujos índices foram definidos manualmente tendem a ter melhor desempenho do que aqueles puramente processados.

2.6.4. Recuperação

A recuperação de casos é um dos processos mais importantes do paradigma RBC. Ele tem como propósito recuperar, de uma memória de casos, o caso mais adequado a uma nova situação e sugerir a solução desse caso ou uma adaptação dela como solução do novo caso.

De acordo com Aamodt & Plaza (1994), o processo de recuperação do caso consiste em recuperar os casos candidatos, e após isto, aplica-se um processo mais elaborado aonde é feita a seleção do melhor caso entre os casos candidatos. Os algoritmos de recuperação baseiam-se nos índices e na organização da memória para direcionar a busca nos casos mais relevantes. Segundo Wangenheim & Wangenheim (2003), o processo de recuperação de casos

pode ser descrito por meio de um conjunto de subtarefas que devem ser realizadas pelo sistema RBC:

- a) assessoramento da situação objetivando a formulação de uma consulta representada por um conjunto de descritores (índices) relevantes da situação ou problema atual;
- b) casamento, objetivando a identificação de um conjunto de casos suficientemente similares à consulta;
- c) seleção, que escolhe o melhor casamento ou casamentos com base no conjunto de casos selecionado.

Para a execução destas subtarefas existem várias técnicas como Vizinho mais Próximo, método de recuperação indutiva, algoritmo de indução, indução guiada por conhecimento, recuperação de padrões, flat memory, entre outros.

2.6.5. Método Vizinho mais Próximo.

Quando o RBC é responsável pela execução somente de um tipo de objetivo de raciocínio, o cálculo do grau de escolha pode ser um processo correto que usa uma função de evolução numérica. Um simples conjunto global de valores de importância é associado às dimensões da biblioteca de casos. Toda característica do caso de entrada é comparada às características correspondentes que são armazenadas, e o grau de comparação de cada par é calculado, baseando-se na importância associada a cada dimensão. Desta forma é calculado e agregado a pontuação ao caso. O caso que tem maior pontuação é utilizado primeiro.

Conforme Abel (1996), neste método de recuperação utiliza-se uma soma ponderada das características entre um novo caso e um armazenado no banco de dados, sendo que cada um dos atributos que compõem possuem um peso, de acordo com sua relevância.

Segundo Abel (1996) a semelhança pode ser encontrada pela fórmula apresentada na Figura 3.

$$\text{Similaridade (T,S)} = \left\{ \sum_{i=1}^n f(T_i, S_i) * W_i \right\}$$

Figura 3 – Cálculo Similaridade

O significado de cada variável é:

- a) T é o novo caso;
- b) S são os casos existentes na memória de casos;
- c) n é o número de atributos;
- d) i é um atributo individual;
- e) f é a função de similaridade para o atributo i os casos T e S;
- f) W é o peso do atributo i.

A maioria das ferramentas RBC utiliza algoritmos como este, normalmente o resultado deve ser entre zero (0) e um (1), onde zero não tem nenhuma similaridade e um é exatamente similar.

2.6.6. Adaptação dos casos.

Uma vez que um caso adequado é recuperado da base de casos, a solução sugerida por este caso é objeto de tentativa de reutilização para a solução do problema atual. Durante este passo, dá-se uma reutilização de conhecimento por meio da transferência da solução do caso anterior, conhecido, para o caso atual, ainda não solucionado.

A reutilização dos casos recuperados no contexto dos novos casos tem foco em dois aspectos: as diferenças entre o caso passado e o caso corrente, e que parte do caso recuperado pode ser transferida para o novo caso. Conforme Kolodner (1993), “a etapa de adaptação pode tomar várias formas entre elas: incluir parte do novo caso na solução do caso antigo; a substituição do valor de algum item, ou a transformação de alguma parte da solução do caso antigo”.

Para Abel (1996), definir a melhor forma de adaptação para alguns problemas pode ser uma tarefa extremamente complexa que pode comprometer a confiabilidade do sistema. Em

muitas aplicações, a solução é simplesmente apresentar o melhor caso e deixar o problema da adaptação para o usuário do sistema. Esta estratégia vem sendo utilizada na maior parte dos sistemas comerciais.

2.7. SISTEMA ATUAL

A Multitherm é uma empresa especialista em software de gestão e automação para a indústria têxtil. Está a vinte anos no mercado e atualmente conta com 26 colaboradores. Seu software de *Enterprise Resource Planing* (ERP) Gestum é um sistema de gestão desenvolvido com o objetivo de integrar todos os setores da empresa por meio da informação.

Para Mendes e Escrivão Filho (2002), o entendimento de ERP compreende desde um conjunto de programas de computador até um sistema de informação gerencial que visa apoiar as decisões estratégicas da empresa.

2.7.1 Área de suporte da Multitherm Sistemas e Automação.

A área de suporte da Multitherm Sistemas e Automação possui seis pessoas envolvidas, incluindo técnicos de suporte e implantadores. Não existe um especialista por módulo, ou seja, o mesmo atendente soluciona questões que podem ir desde a área de produção até o financeiro. As informações sobre o atual sistema foram obtidas através de entrevistas informais com as pessoas envolvidas no setor de suporte ao cliente.

O processo de atendimento atual tem início com o cliente que expõem sua dúvida através de um telefonema ou e-mail, se o atendente souber a solução então responde de imediato o cliente, caso não, pergunta para o desenvolvedor do módulo em questão. Porém, este processo nem sempre é simples, pois o atendente sabe que já respondeu sobre esta mesma dúvida a outro cliente, mas muitas vezes não recorda a solução anteriormente apresentada e não encontra este mesmo atendimento no atual sistema de controle das atividades.

A Figura 4 apresenta o diagrama de atividades com o processo de atendimento ao cliente da empresa Multitherm Sistemas e Automação.

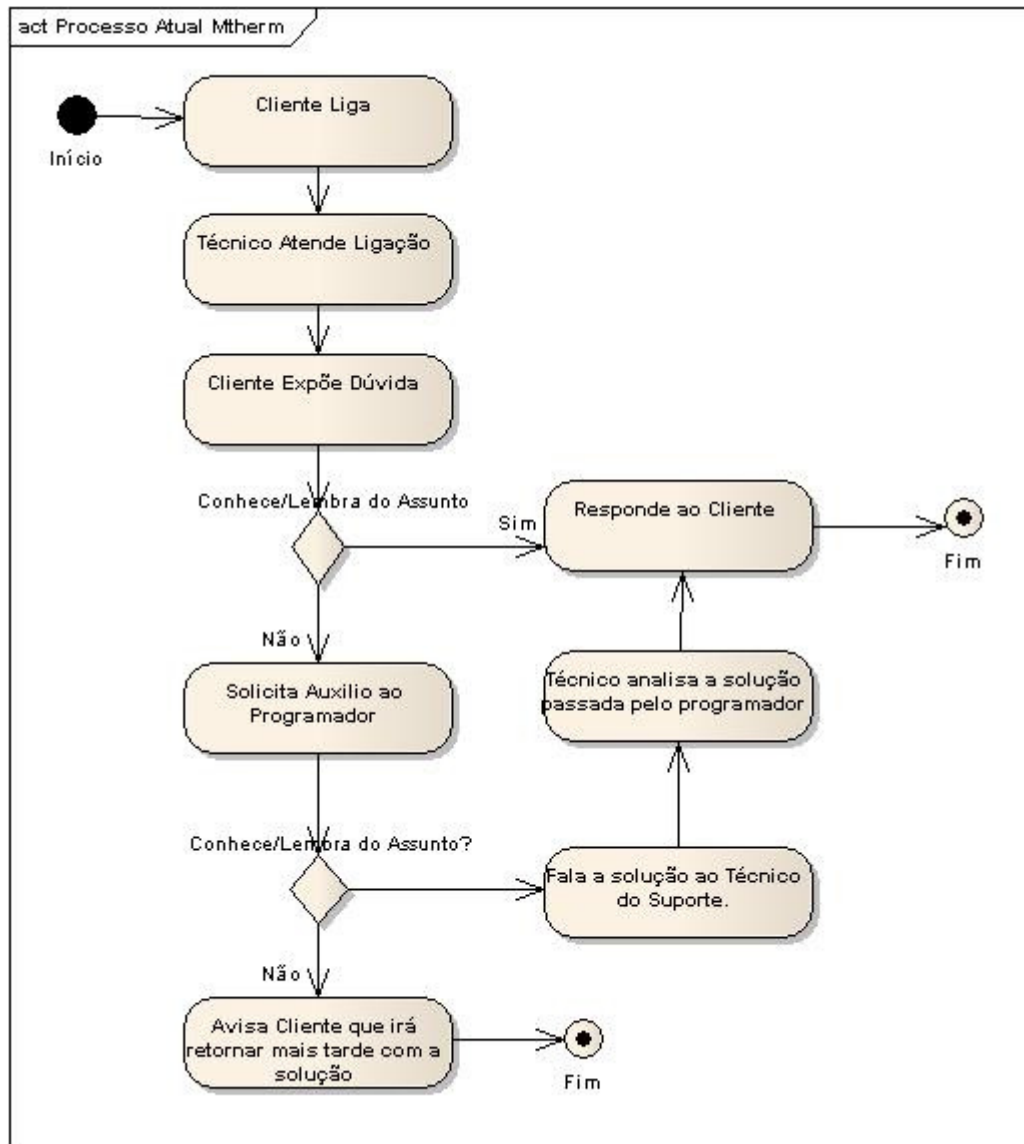


Figura 4 – Diagrama de Atividades Atual

Ocorre com frequência situações onde o atendente está atendendo um cliente e surge uma dúvida de outro cliente. Neste caso o segundo cliente ficará aguardando o atendimento ou um retorno posterior. Esta situação causa insatisfação para o cliente e um desconforto para o técnico que atendeu a ligação, por não ter uma resposta ou solução para ser apresentada.

Para controlar as atividades utiliza-se um sistema de suporte de decisões (SSD) chamado 'Suporte' desenvolvido pela própria empresa. Cada atendimento é inserido no sistema com a descrição da dúvida e os passos para sua solução. Caso o atendimento tenha que ser enviado para outro usuário, este repasse é feito criando uma nova pendência dentro do atendimento. Uma pendência é criada a fim de dar continuidade ao processo, ao iniciar a resolução de uma pendência o usuário deve inserir o horário de início, no término deve também ser inserido o horário final.

Após resolver o problema em questão o atendente descreve no atendimento a resolução do problema, ou seja, os passos que levaram para a solução e então finaliza o atendimento. Na Figura 5 é possível visualizar a tela inicial do sistema evidenciando um atendimento e suas pendências:

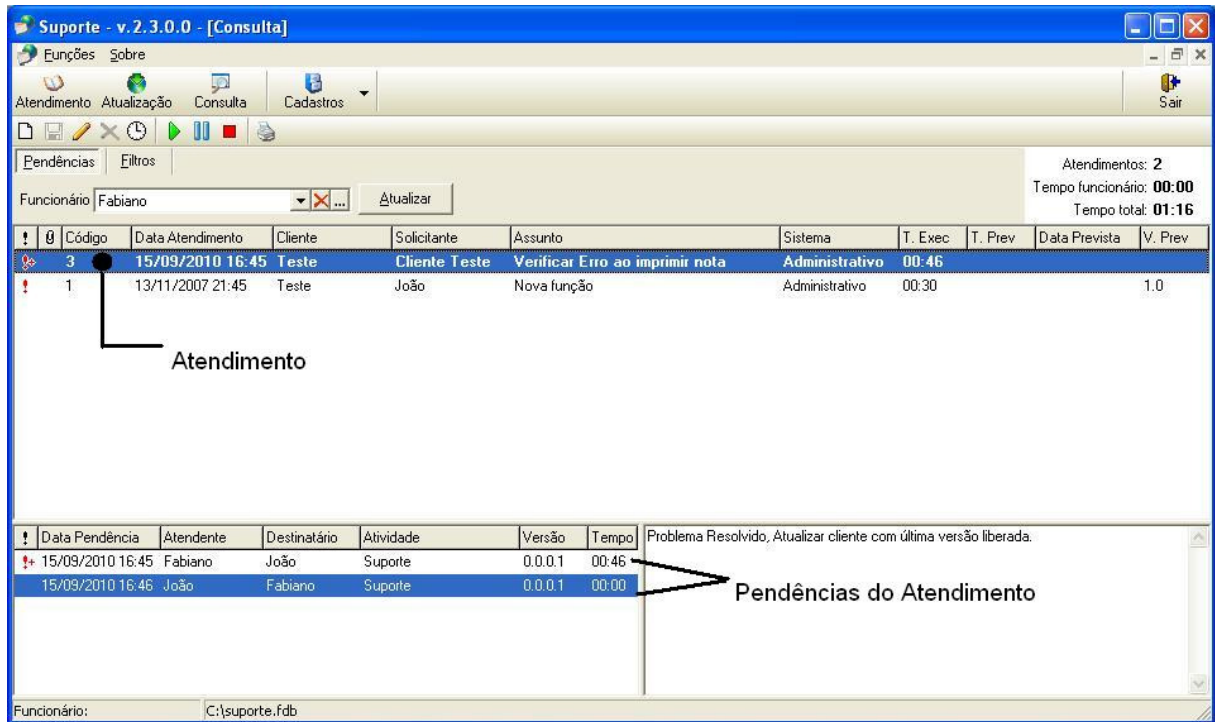


Figura 5 – Tela Inicial Sistema Suporte

Por conter o cadastro dos sistemas desenvolvidos, clientes, versões por cliente e cadastro de usuários, o software Suporte é utilizado também no setor de desenvolvimento de software e implantação. No desenvolvimento de software é utilizado para registrar o trabalho efetuado pelos programadores, analistas e testadores enquanto que no setor de implantação é utilizado para controlar as versões atualizadas nos clientes.

2.8. TRABALHOS CORRELATOS

Vários trabalhos envolvendo Raciocínio Baseado em Casos foram encontrados, entre os mesmos apresentam-se a seguir três trabalhos.

Lira (2002) expôs em seu TCC de BCC, um Sistema de Informação utilizando a técnica RBC para auxílio na elaboração de preços para esquadrias de alumínio. Para tal foi utilizado o ambiente de programação Delphi e o banco de Dados Microsoft Access.

Kienen (2003), propôs em seu Trabalho de Conclusão do Curso (TCC) de Bacharelado em Ciência da Computação (BCC), uma ferramenta que utiliza RBC a fim de encontrar processos jurídicos de acordo com o grau de similaridade. Neste trabalho foi utilizada a ferramenta Delphi 5 e o banco de dados MySQL.

Ehlert (2004), através de seu TCC de BCC, implementou um sistema para sugestão da melhor opção de transporte e rota para transportadoras utilizando a técnica de RBC, para implementar este sistema utilizou-se o ambiente de programação Delphi e o banco de Dados Interbase.

3. DESENVOLVIMENT110

Neste capítulo são descritos as especificações do sistema, as suas características, os principais requisitos, os diagramas de caso de uso, o modelo entidade relacionamento, a operacionalidade do sistema e ao final os resultados e discussão.

3.1. LEVANTAMENTO DE INFORMAÇÕES

Através de entrevistas realizadas com os usuários do atual sistema, foi possível identificar as dificuldades encontradas na tentativa de buscar atendimentos antigos. Atualmente a busca por atendimentos passados é feita através de uma série de filtros conforme mostra a Figura 6. A reclamação recorrente é de que são muitos filtros a serem feitos e caso o usuário tente pesquisar por “Assunto” a busca não é eficaz, pois, nem sempre os atendentes ao finalizar o atendimento colocam o assunto que condiz com o problema.

Figura 6 – Tela de Consulta Atual

Para que a ampliação desenvolvida funcione de maneira eficaz, devem ser seguidos os seguintes procedimentos padrões, a serem efetuados pelo setor de suporte ao cliente:

- a) técnico atende a ligação do cliente;
- b) cliente expõe dúvida;
- c) efetua-se a busca por casos similares;
- d) caso a busca seja efetuada com sucesso repassar solução ao cliente, se houve necessidade de uma adaptação da solução, armazenar um novo caso;
- e) caso a busca não retornou resultados, buscar solução e após resolver problema do cliente armazenar um novo caso com todos os passos para a solução do problema informado.

Abaixo segue o diagrama de atividades dos procedimentos a serem seguidos.

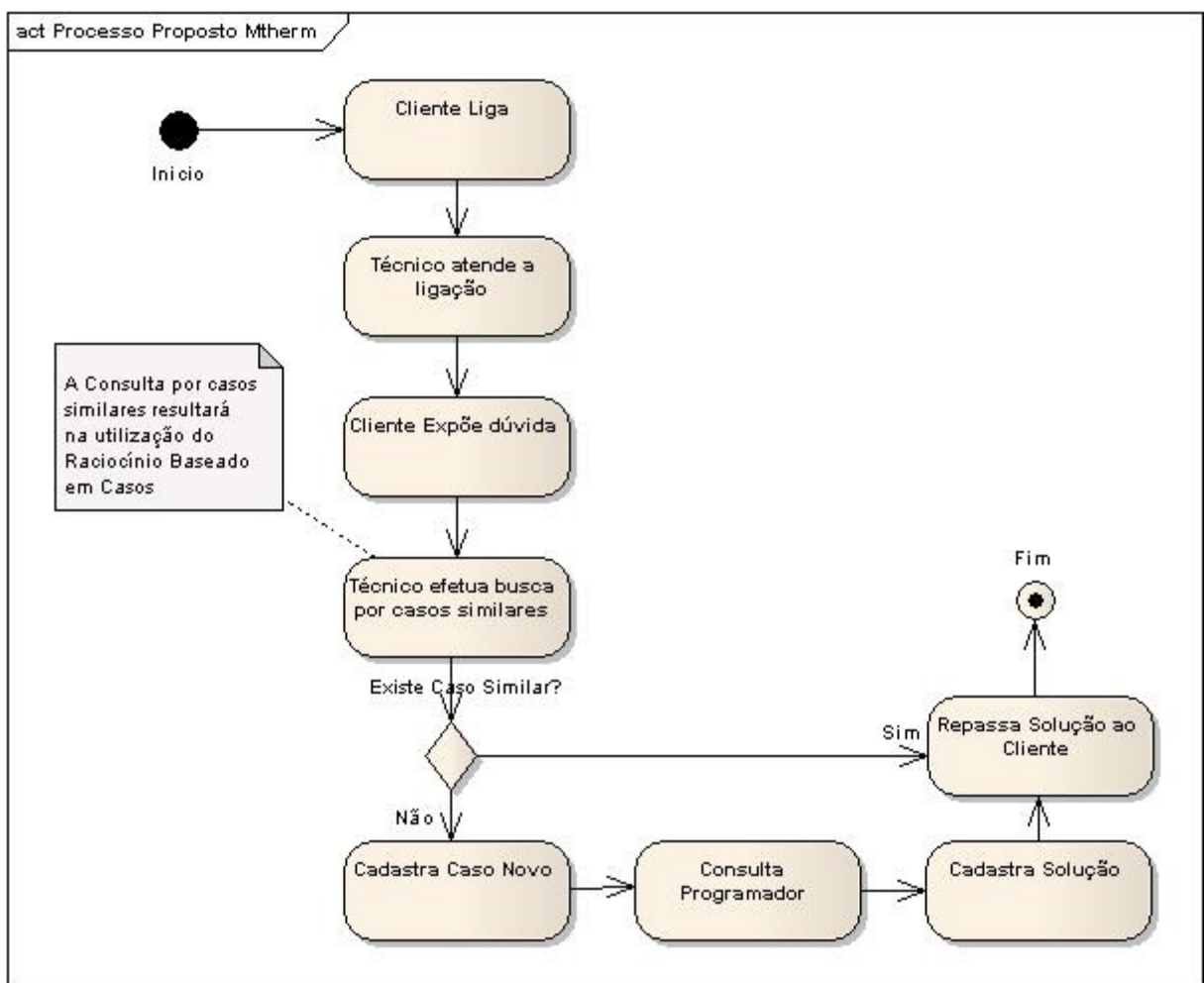


Figura 7 – Diagrama de atividades a ser seguido

3.2. REQUISITOS PRINCIPAIS DO SISTEMA

Nesta seção serão apresentados os principais requisitos funcionais (RF), requisitos não funcionais (RNF), sua rastreabilidade com seus respectivos casos de uso. No Quadro 1 são apresentados os requisitos funcionais do sistema.

Requisitos Funcionais	Caso de Uso
RF01: O sistema deverá possibilitar a indexação dos atendimentos.	UC01
RF02: O sistema deverá possibilitar a criação, alteração e exclusão de palavras-chave.	UC02
RF03: O sistema deverá possibilitar anexar palavras-chave ao caso.	UC03

Quadro 1 - Requisitos funcionais

O Quadro 2 lista os requisitos não funcionais do sistema.

Requisitos Não Funcionais
RNF01: O sistema deverá ser implementado na linguagem Delphi.
RNF02: O sistema deverá utilizar o banco de dados Firebird.

Quadro 2 - Requisitos não funcionais

3.3. ESPECIFICAÇÃO

Esta seção apresenta os diagramas que serão necessários para o entendimento da ampliação feita no atual sistema. Para as especificações utilizou-se a notação UML, sendo os diagramas de caso de uso gerados através da ferramenta *Enterprise Architect*, o modelo conceitual de dados gerado através da ferramenta *DBDesigner* e o diagrama de classes do sistema gerado através da ferramenta *Power Designer*.

3.4. DIAGRAMAS DE CASO DE USO

Abaixo estão apresentados o diagrama de casos de uso do sistema, sendo que os detalhes dos principais estão descritos no Apêndice A. Na Figura 8 tem-se o diagrama de caso de uso para a ampliação.

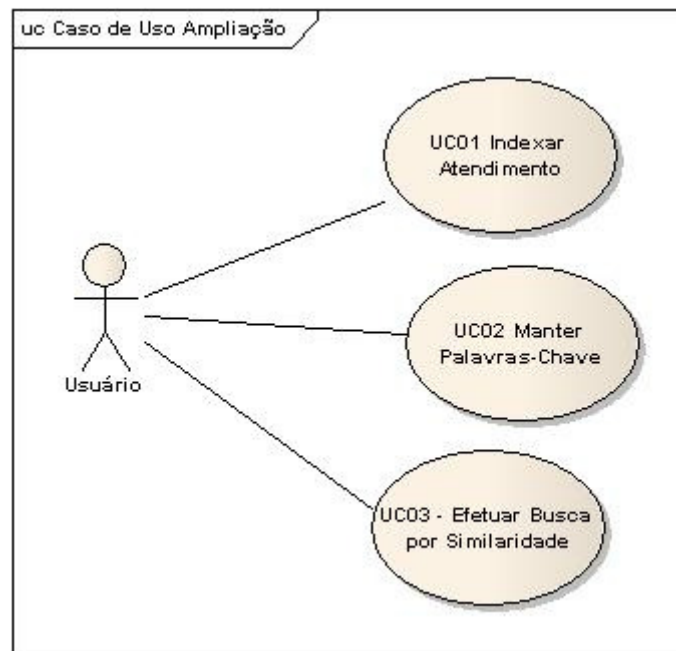


Figura 8 – Diagrama de Caso de uso ampliação

Na Figura 9 tem-se o diagrama dos principais casos de uso do sistema.

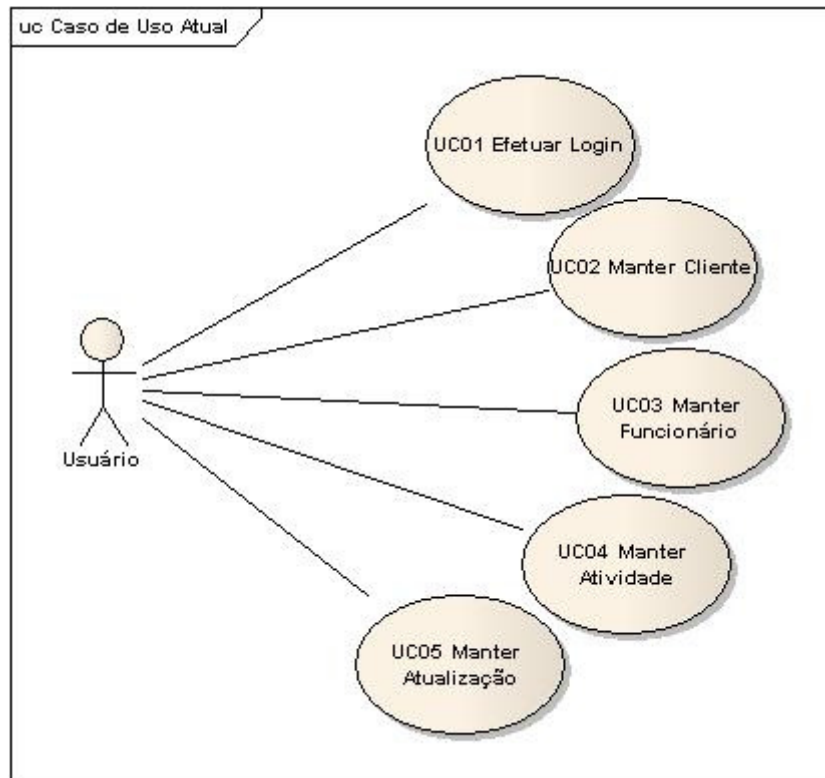


Figura 9 – Principais casos de uso do sistema

3.5. MODELO CONCEITUAL DE DADOS

Na Figura 10 se apresenta o modelo entidade relacionamento que representam as entidades que serão persistidas no bando de dados. Modelo entidade relacionamento é um modelo abstrato que busca descrever de maneira conceitual, os dados a serem utilizados em um sistema.

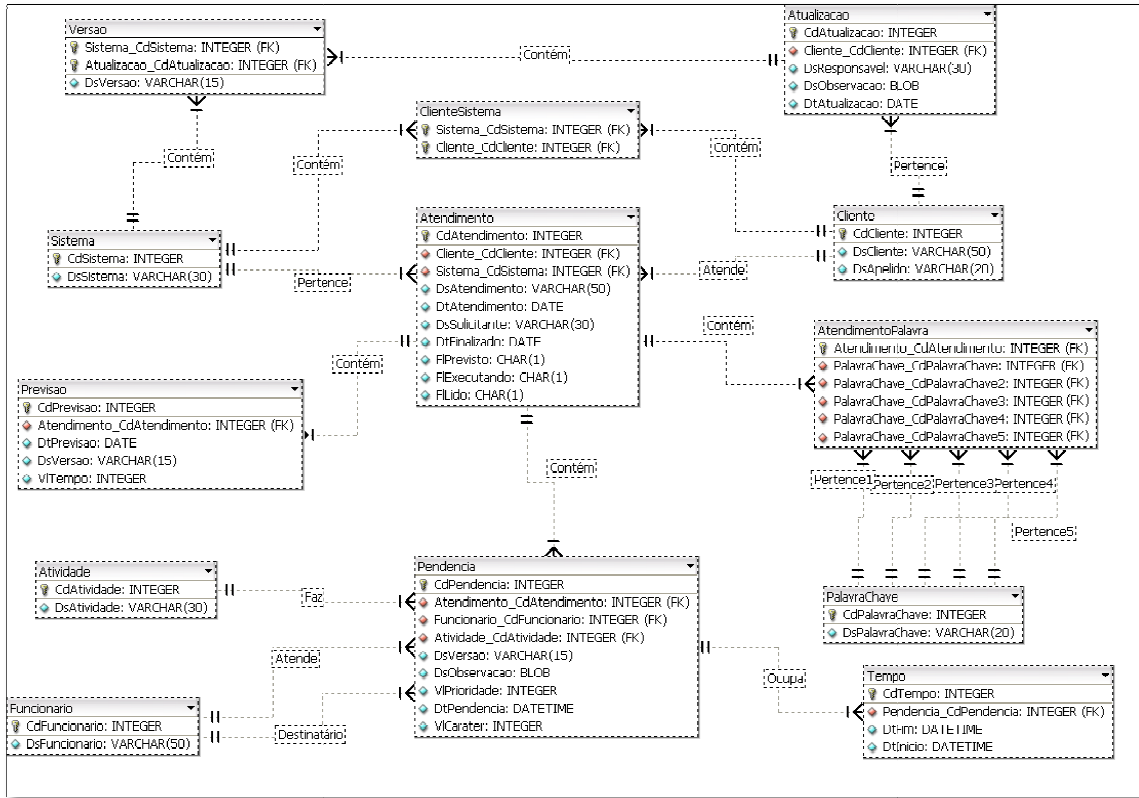


Figura 10 – Modelo entidade-relacionamento

A seguir é apresentado o dicionário de dados das tabelas do sistema, que visa oferecer uma explicação de forma textual sobre os campos contidos nas tabelas. Os tipos Varchar são strings de tamanho variável, os campos com o tipo Integer contêm números inteiros com ou sem sinal, os tipos Date armazenam datas no formato “AAAA-MM-DD”, os tipos Datetime são tipos de campos que armazenam data e hora no formato “AAAA-MM-DD HH:MM:SS”, os tipos Char são um conjunto de string fixo, seu tamanho é definido no momento da criação do campo, o tipo Blob é um campo criado para o armazenamento de qualquer tipo de informação em formato binário. Chave primária é o conjunto de um ou mais campos que não aceitam valores nulos e nem repetição, chave estrangeira é um campo, que aponta para a chave primária de outra tabela ou da mesma tabela, criando um relacionamento entre elas.

No Quadro 3 se apresenta o dicionário de dados referente à tabela Funcionário que é responsável pelo armazenamento de dados dos usuários do sistema.

Tabela: Funcionario				
Nome	Tipo	Tamanho	Descrição	Obrigatório
CdFuncionario	Integer		Chave Primária da tabela, armazena o código do usuário do sistema.	Sim
DsFuncionario	Varchar	50	Armazena o nome do usuário.	Não

Quadro 3 – Dicionário de dados para a tabela “Funcionario”

No Quadro 4 observa-se o dicionário de dados da tabela Atividade que é responsável pelo armazenamento das atividades efetuadas nas pendências.

Tabela: Atividade				
Nome	Tipo	Tamanho	Descrição	Obrigatório
CdAtividade	Integer		Chave Primária da tabela, armazena o código da atividade.	Sim
DsAtividade	Varchar	50	Armazena a descrição da atividade.	Não

Quadro 4 – Dicionário de dados para a tabela “Atividade”

No Quadro 5 observa-se o dicionário de dados da tabela Pendencia que é responsável pelo armazenamento de dados das pendências de cada atendimento.

Tabela: Pendencia				
Nome	Tipo	Tamanho	Descrição	Obrigatório
CdPendencia	Integer		Chave Primária, armazena o código da Pendência.	Sim
CdAtendimento	Integer		Chave Estrangeira da tabela Atendimento, armazena o código do atendimento que a pendência pertence.	Sim
CdFuncionario	Integer		Chave Estrangeira da Tabela Funcionário, armazena o código do funcionário que executou a pendência.	Sim
CdAtividade	Integer		Chave Estrangeira ligando a tabela Atividade, armazena o código da atividade informada na pendência.	Sim
VICarater	Integer		Armazena o tipo de urgência da pendência.	Não
CdAtendente	Integer		Chave Estrangeira ligando a tabela Funcionario, armazena o código do usuário remetente da pendência.	Sim
CdDestinatario	Integer		Chave Estrangeira ligando a tabela Funcionario, armazena o código do usuário destinatário da pendência.	Sim
DsObservacao	Blob		Armazena a descrição do trabalho feito pelo destinatário da pendência.	Não
DsVersao	Varchar	15	Armazena a versão liberada pelo programador em que o problema relatado pelo cliente foi resolvido.	Não
DtPendencia	Date		Armazena a data que a pendência foi criada.	Não
VIPrioridade	Integer		Armazena o código de urgência da pendência.	Não

Quadro 5 – Dicionário de dados para a tabela “Pendencia”

No Quadro 6 observa-se o dicionário de dados da tabela Tempo que é responsável pelo armazenamento do tempo trabalhado em cada pendência.

Tabela: Tempo				
Nome	Tipo	Tamanho	Descrição	Obrigatório
CdTempo	Integer		Chave Primária da tabela, armazena o código do registro.	Sim
CdPendencia	Integer		Chave Estrangeira ligando a tabela Pendencia, armazena o código da pendência que o tempo foi marcado.	Sim
DtInicio	DateTime		Armazena a Data/Hora que o usuário começou a resolver o problema relacionado à pendência.	Não
DtFim	DateTime		Armazena a Data/Hora que o usuário finalizou a resolução da pendência.	Não

Quadro 6 – Dicionário de dados para a tabela “Tempo”

No Quadro 7 observa-se o dicionário de dados da tabela Cliente que armazena os dados do cadastro de cliente.

Tabela: Cliente				
Nome	Tipo	Tamanho	Descrição	Obrigatório
CdCliente	Integer		Chave Primária da tabela, armazena o código do cliente.	Sim
DsCliente	Varchar	50	Armazena o nome do cliente	Não
DsApelido	Varchar	20	Armazena o Apelido	Não

Quadro 7 – Dicionário de dados para a tabela “Cliente”

No Quadro 8 observa-se o dicionário de dados da tabela Sistema que é responsável pelo armazenamento dos dados do cadastro de sistemas desenvolvidos pelo Multitherm Sistemas e Automação.

Tabela: Sistema				
Nome	Tipo	Tamanho	Descrição	Obrigatório
CdSistema	Integer		Chave Primária da tabela Sistema, armazena o código o sistema.	Sim
DsSistema	Varchar	30	Armazena a descrição do sistema cadastrado desenvolvido.	Não

Quadro 8 – Dicionário de dados para a tabela “Sistema”

No Quadro 9 observa-se o dicionário de dados da tabela ClienteSistema que é responsável pelo armazenamento dos dados provenientes da ligação da tabela cliente com a tabela sistema.

Tabela: ClienteSistema				
Nome	Tipo	Tamanho	Descrição	Obrigatório
CdSistema	Integer		Chave Primária e Estrangeira ligando a tabela Sistema, armazena o código do sistema cadastrado para o cliente.	Sim
CdCliente	Integer		Chave Primária e Estrangeira ligando a tabela Cliente, armazena o código do cliente cujo sistema foi cadastrado.	Sim

Quadro 9 – Dicionário de dados para a tabela “ClienteSistema”

No Quadro 10 observa-se o dicionário de dados da tabela Atualizacao que é responsável pelo armazenamento dos dados das atualizações feitas nos clientes.

Tabela: Atualizacao				
Nome	Tipo	Tamanho	Descrição	Obrigatório
CdAtualizacao	Integer		Chave Primária da tabela, armazena o código da atualização.	Sim
CdCliente	Integer		Chave Estrangeira ligando a tabela cliente, armazena o código do cliente atualizado.	Sim
DtAtualizacao	Date		Armazena a data que o cliente foi atualizado.	Não
DsResponsavel	Varchar	30	Armazena o nome da pessoa que atualizou o sistema.	Não
DsObservacao	Blob		Armazena alguma observação que possa ser relevante referente à atualização efetuada.	Não

Quadro 10 – Dicionário de dados para a tabela “Atualizacao”

No Quadro 11 observa-se o dicionário de dados da tabela Versao que é responsável pelo armazenamento dos dados provenientes da ligação das tabelas atualização e sistema.

Tabela: Versao				
Nome	Tipo	Tamanho	Descrição	Obrigatório
CdSistema	Integer		Chave Primária e Estrangeira ligando a tabela Sistema, armazena o código do sistema atualizado.	Sim
CdAtualizacao	Integer		Chave Primária e Estrangeira ligando a tabela Atualizacao, armazena o código a atualização que o sistema foi atualizado.	Sim
DsVersao	Varchar	15	Armazena o código da release atualizada do sistema atualizada no cliente.	Não

Quadro 11 – Dicionário de dados para a tabela “Versao”

No Quadro 12 observa-se o dicionário de dados da tabela Atendimento que é responsável pelo armazenamento dos dados do atendimento cadastrado.

Tabela: Atendimento				
Nome	Tipo	Tamanho	Descrição	Obrigatório
CdAtendimento	Integer		Chave Primária, armazena o código do atendimento.	Sim
CdSistema	Integer		Chave Estrangeira ligando a tabela Sistema, armazena o código do sistema a que será prestado atendimento.	Sim
CdCliente	Integer		Chave Estrangeira ligando a tabela Cliente, armazena o código do cliente que solicitou atendimento.	Sim
DsAtendimento	Varchar	50	Armazena o assunto a que se refere o atendimento.	Não
DtAtendimento	DateTime		Armazena a data/hora que o atendimento foi criado.	Não
DtFinalizado	DateTime		Armazena a data/hora que o atendimento foi finalizado.	Não
DsSolicitante	Varchar	30	Armazena o nome do solicitante da pendência.	Não
FlExecutando	Char	1	Identifica se o atendimento esta sendo executado.	Não
FlPrevisto	Char	1	Identifica se o atendimento tem uma data prevista para execução.	Não
FlLido	Char	1	Identifica se o atendimento foi lido.	Não
CdPrevisao	Integer		Chave Estrangeira ligando a tabela Previsao, armazena o código da previsão do atendimento.	Não

Quadro 12 – Dicionário de dados para a tabela “Pendencia”

No Quadro 13 observa-se o dicionário de dados da tabela Previsao que é responsável pelo armazenamento dos dados referentes à previsão de realização de uma determinada pendência.

Tabela: Previsao				
Nome	Tipo	Tamanho	Descrição	Obrigatório
CdPrevisao	Integer		Chave Primária da tabela Sistema, armazena o código da previsão.	Sim
CdAtendimento	Integer		Chave Estrangeira ligando a tabela atendimento, armazena o código do atendimento que a previsão esta ligada.	Não
DtPrevisao	Date		Armazena a data da previsão de realização do atendimento.	Não
DsVersao	Varchar	15	Armazena a Release que foi liberado com a correção/alteração constante no atendimento.	Não
VITempo	Integer		Armazena o tempo previsto para realização do atendimento.	Não

Quadro 13 – Dicionário de dados para a tabela “Previsao”

No Quadro 14 observa-se o dicionário de dados da tabela PalavraChave responsável pelo armazenamento dos dados do cadastro de palavras-chave.

Tabela: PalavraChave				
Nome	Tipo	Tamanho	Descrição	Obrigatório
CdPalavraChave	Integer		Chave Primária da tabela Sistema, armazena o código da palavra chave.	Sim
DsPalavraChave	Varchar	20	Armazena a descrição da palavra chave.	Não

Quadro 14 – Dicionário de dados para a tabela “PalavraChave”

No Quadro 15 observa-se o dicionário de dados da tabela AtendimentoPalavra que é responsável pelo armazenamento dos dados da indexação de cada atendimento. É nesta tabela que é feita a busca por casos comparando as palavras-chave informadas com as cadastradas para cada atendimento.

Tabela: AtendimentoPalavra				
Nome	Tipo	Tamanho	Descrição	Obrigatório
CdAtendimento	Integer		Chave Primária e Estrangeira ligando a tabela Atendimento, armazena o código do atendimento.	Sim
CdPalavraChave1	Integer		Chave Estrangeira ligando a tabela PalavraChave, armazena o código da primeira palavra chave do atendimento.	Sim
CdPalavraChave2	Integer		Chave Estrangeira ligando a tabela PalavraChave, armazena o código da segunda palavra chave do atendimento.	Sim
CdPalavraChave3	Integer		Chave Estrangeira ligando a tabela PalavraChave, armazena o código da terceira palavra chave do atendimento.	Sim
CdPalavraChave4	Integer		Chave Estrangeira ligando a tabela PalavraChave, armazena o código da quarta palavra chave do atendimento.	Sim
CdPalavraChave5	Integer		Chave Estrangeira ligando a tabela PalavraChave, armazena o código da quinta palavra chave do atendimento.	Sim

Quadro 15 – Dicionário de dados para a tabela “PalavraChave”

3.6. IMPLEMENTAÇÃO

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

3.6.1. Técnicas e ferramentas utilizadas

Para a ampliação do sistema foi usada a ferramenta Delphi na sua versão XE, o atual sistema foi desenvolvido nesta mesma plataforma em sua versão 7. Então foram atualizados os fontes do atual sistema de Delphi 7 para a versão utilizada na ampliação.

O sistema gerenciador de banco de dados utilizado foi o Firebird, que também já é utilizado pelo sistema atual e foi atualizado da versão 2.1 para a 2.5.

No desenvolvimento da ampliação foi utilizado o conceito de orientação a objetos com camadas, sendo uma camada de acesso ao banco de dados, outra camada para acesso dos dados recuperados e uma terceira que guarda a lista dos registros retornados do banco, a seguir uma breve explicação das classes exibidas na figura 10:

- a) classes de acesso ao banco de dados: TPalavraChaveDb, TSistemaDb, TAtividadeDb, TFuncionarioDb, TClienteDb, TAtendimentoD, TPendenciaDb;
- b) classes para acesso aos atributos: TPalavraChave, TSistema, TAtividade, TFuncionario, TCliente, TAtendimento, TPendencia;
- c) classes que guardam lista de objetos: TListPalavraChaveDb, TListSistemaDb, TListAtividadeD, TListFuncionarioDb, TListClienteDb, TListAtendimentoDb, TListPendenciaDb.

A Classe MSuporte guarda lista de determinados objetos em memória para tornar o sistema mais rápido.

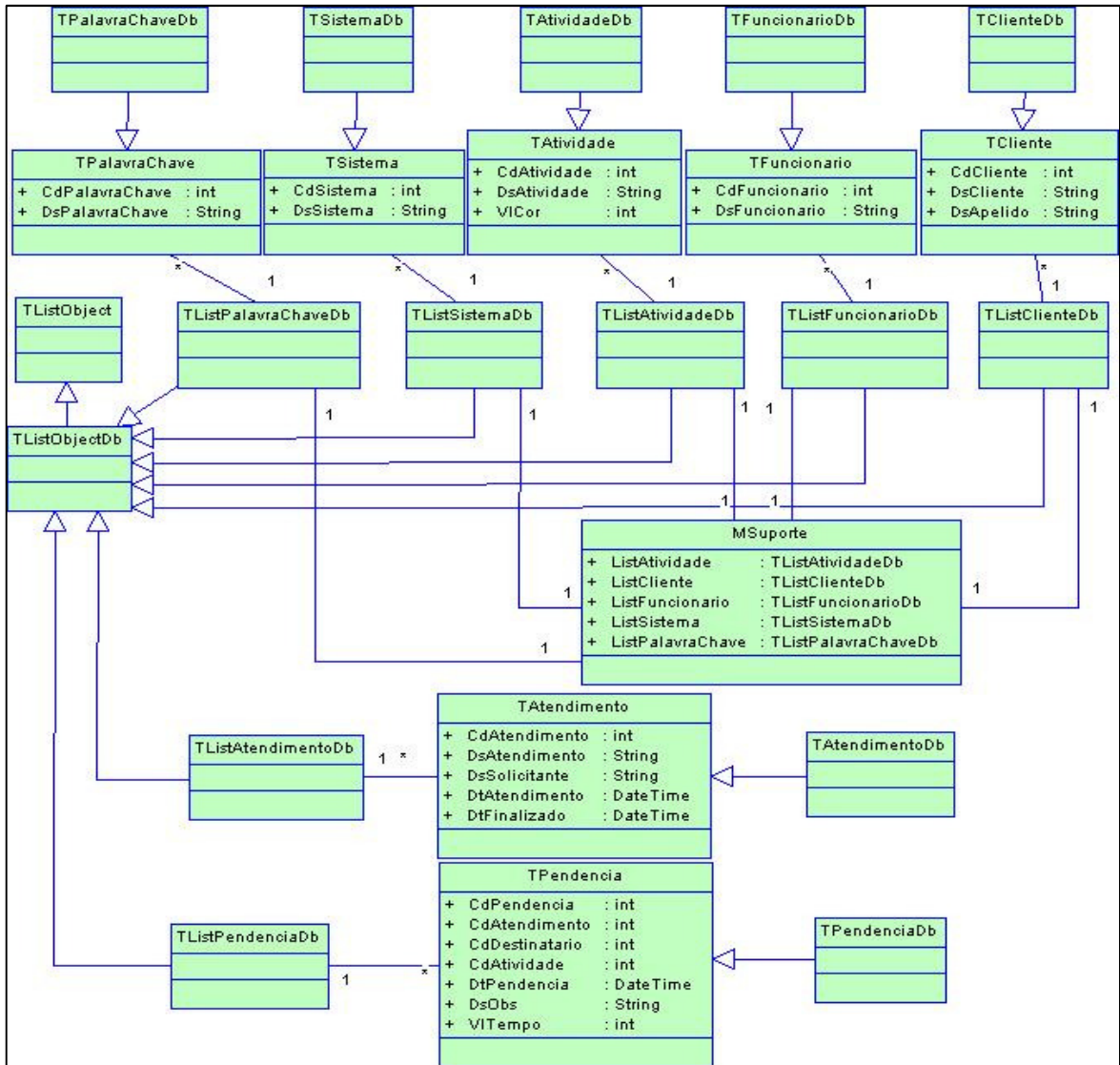


Figura 11 – Diagrama de Classes do sistema

3.6.2. Aplicação do RBC no Sistema

A rapidez na resposta dada ao cliente na resolução de um problema é essencial para manter confiança no sistema e a satisfação do mesmo. A aplicação do RBC no atual sistema tem foco nesta rapidez e no aumento da produtividade dos atendentes, que poderão encontrar rapidamente soluções para problemas relatados, compartilhar com os outros colaboradores novas soluções e atualizações nas existentes.

O exemplo a seguir demonstra a aplicação do RBC na pesquisa de similaridades feita durante o atendimento telefônico de uma ocorrência. O cliente liga para o setor de suporte

relatando o seguinte caso “*Estou tentando emitir o relatório da op através da tela de OP, mas está aparecendo uma mensagem de erro. Enviei a mensagem por email agora, o que pode ter acontecido?*”. O atendente apenas tem o trabalho de ver a mensagem de erro, analisar a situação exposta, e separar os termos principais do problema que esta sendo relatado. No exemplo, pode-se afirmar que os termos principais são:

- a) relatório da Op;
- b) *sql Error*;
- c) tela de OP;
- d) *udf_copy*;
- e) *access violation*.

No Quadro 16 tem-se uma base de casos simplificada contendo casos que podem ajudar a solucionar o problema através do RBC.

Palavra-Chave1	Palavra-Chave2	Palavra-Chave3	Palavra-Chave4	Palavra-Chave5	Solução
Gerar OP	<i>Sql Error</i>	Tinturaria	Selecionar 2 itens	Itens com menos de 1 kg	Erro de SQL corrigido na versão 5.54.0.10.
Cadastro Processo	Inserção	<i>Sql Error</i>	<i>Unknown parameter</i>	Fecha cadastro	Erro de SQL corrigido na versão 5.55.0.32.
<i>Access Violation</i>	Relatório da OP	<i>Udf_copy</i>	<i>Sql Error</i>	Tela de OP	Ao instalar os arquivos no servidor da base faltou o Midas.dll.

Quadro 16 – Base de Casos

Atribuindo os respectivos pesos 1 para as palavras coincidentes e 0 para não coincidentes temos o Quadro 17:

Atendimento	Palavra-Chave1	Palavra-Chave2	Palavra-Chave3	Palavra-Chave4	Palavra-Chave5
1	0	1	0	1	0
2	0	0	1	0	0
3	1	1	1	1	1

Quadro 17 – Resultados Obtidos

Considerando o quadro anterior a comparação entre os casos será conforme cálculo apresentado no Quadro 18.

ConsultaSimilaridade(1) =	$\frac{0 + 1 + 0 + 1 + 0}{5} = \frac{2}{5} =$	0,40
ConsultaSimilaridade(2) =	$\frac{0 + 0 + 1 + 0 + 0}{5} = \frac{1}{5} =$	0,20
ConsultaSimilaridade(3) =	$\frac{1 + 1 + 1 + 1 + 1}{5} = \frac{5}{5} =$	1

Quadro 18 – Comparação entre os casos

O atendimento três, já registrado anteriormente é o mais semelhante, pois é o que mais se aproxima de um.

3.6.3. Operacionalidade da implementação

Nesta subseção é apresentada a sequência de telas e operações que o usuário efetua para efetuar o login no sistema, cadastrar as palavras-chave, cadastrar um atendimento, resolvê-lo, indexá-lo e efetuar a busca por atendimentos anteriores através da técnica RBC. Também serão apresentados trechos dos códigos fonte de algumas rotinas implementadas.

Nas figuras 12 e 13 pode ser visualizada a tela de login do sistema. Como não é necessário senha de acesso, o usuário escolhe seu login entre os listados.

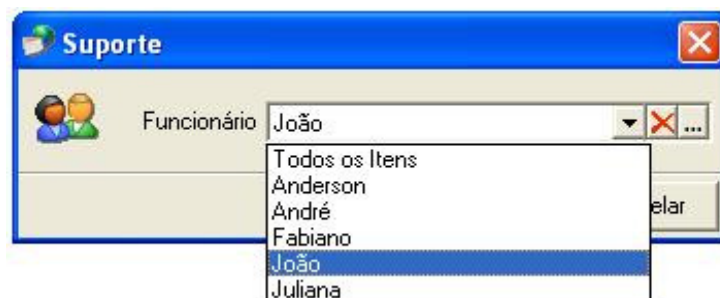


Figura 12– Lista de usuários para login



Figura 13 – Usuário selecionado para login

Após o usuário ser validado, este terá acesso à tela principal onde serão exibidos seus atendimentos em aberto, também terá acesso a telas de cadastros conforme mostra a Figura 14.

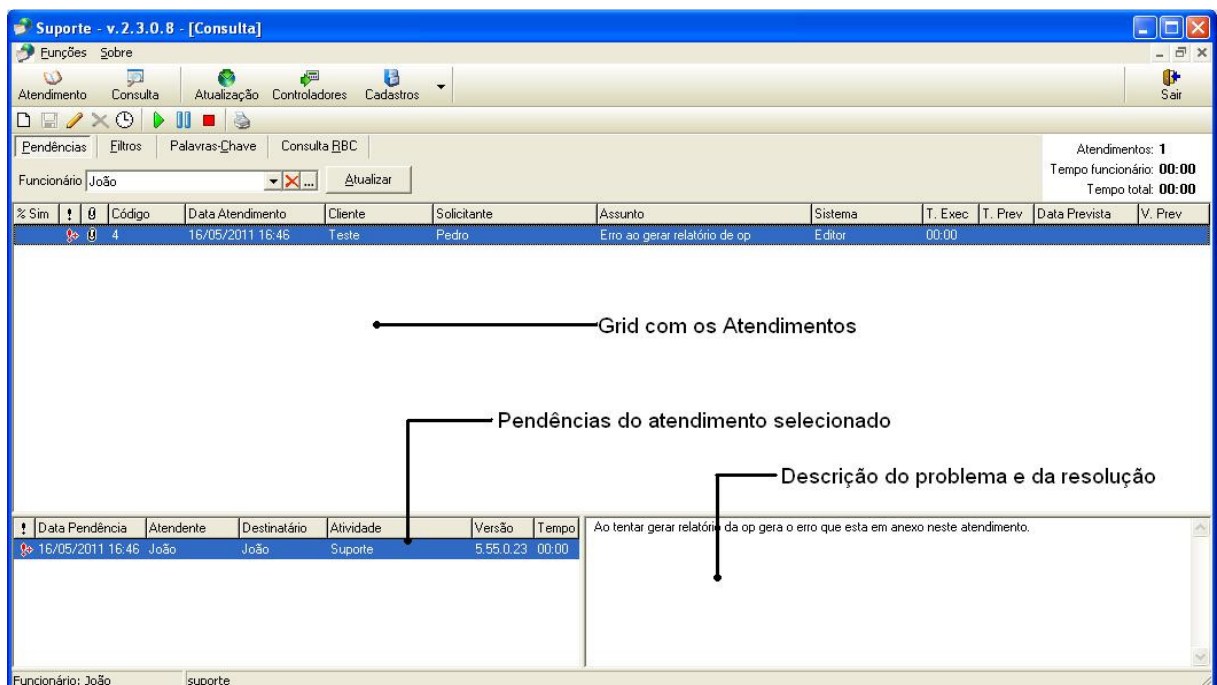


Figura 14 – Tela inicial do sistema

Ao acessar a opção Cadastros - Palavras-Chave, o usuário pode cadastrar as palavras-chave que serão posteriormente utilizadas na indexação dos casos.

A Figura 15 demonstra o cadastro de palavras-chave.

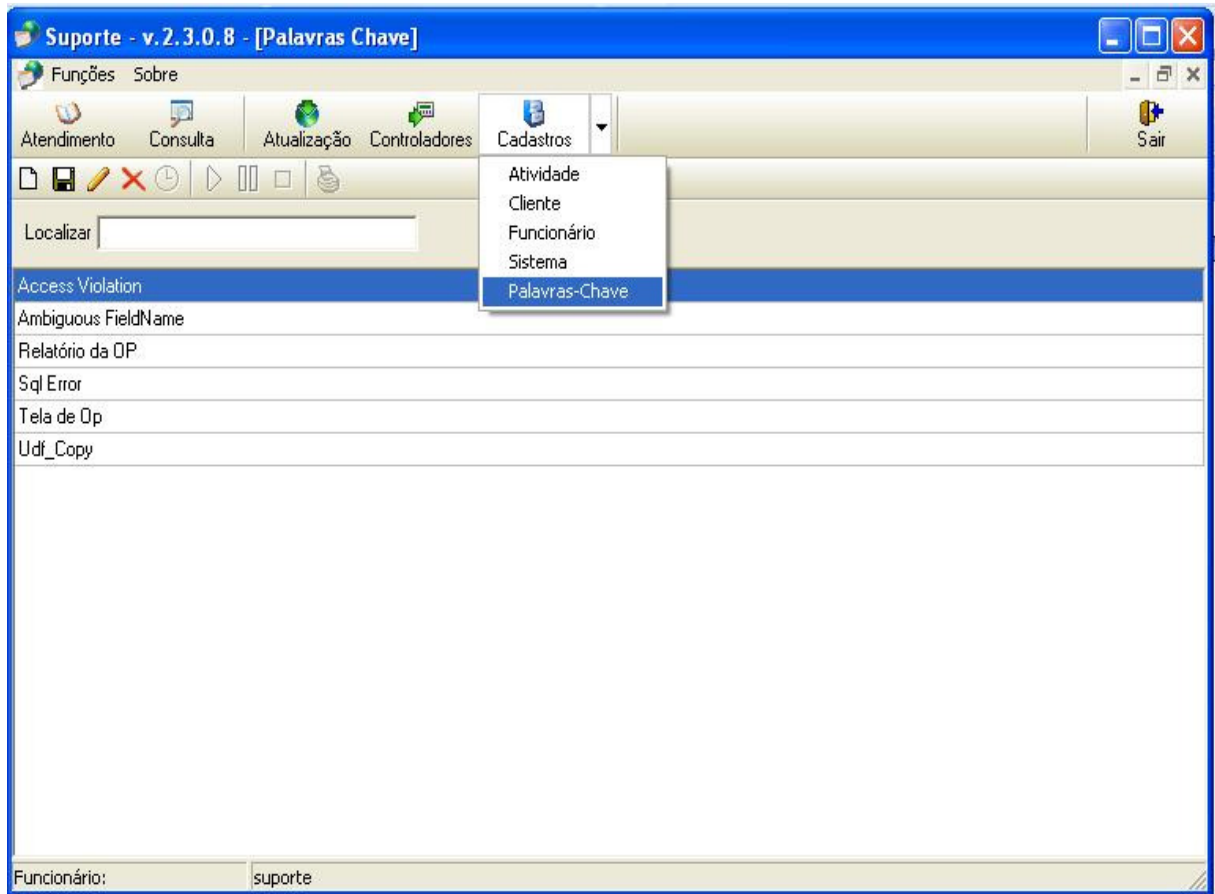


Figura 15 – Cadastro de palavras-chave

A seguir, na Figura 16 é apresentada a tela de cadastro dos atendimentos. Nela o usuário descreve o problema a ele relatado.

Atendente: João

Destinatário: João

Cliente: Cliente Teste Ltda

Solicitante: Pedro

Sistema: Editor Versão: 5.55.0.23

Atividade: Suporte

Assunto: Erro ao gerar relatório de op

Prioridade: Altíssima

Caráter: Normal

Atendimento previsto

Previsão

Data: / /

Versão:

Tempo: 00 : 00

erro.JPG


Ao tentar gerar relatório da op gera o erro que esta em anexo neste atendimento,

Descrição do problema relatado

Arquivos anexos referentes ao problema.

Funcionário: suporte

Figura 16 – Cadastro de um novo Atendimento

Uma vez que o atendimento está cadastrado, o atendente da sequência no processo registrando a solução ou enviando para o desenvolvimento resolver caso haja necessidade, isto é feito inserindo uma nova pendência no mesmo atendimento. É neste ponto que o usuário irá registrar as informações de forma padronizada e organizada, compartilhando-as com os outros atendentes. Para isto, na tela principal o usuário seleciona o atendimento desejado e clica no botão , será aberta a tela de cadastro de pendência conforme mostra a Figura 17.

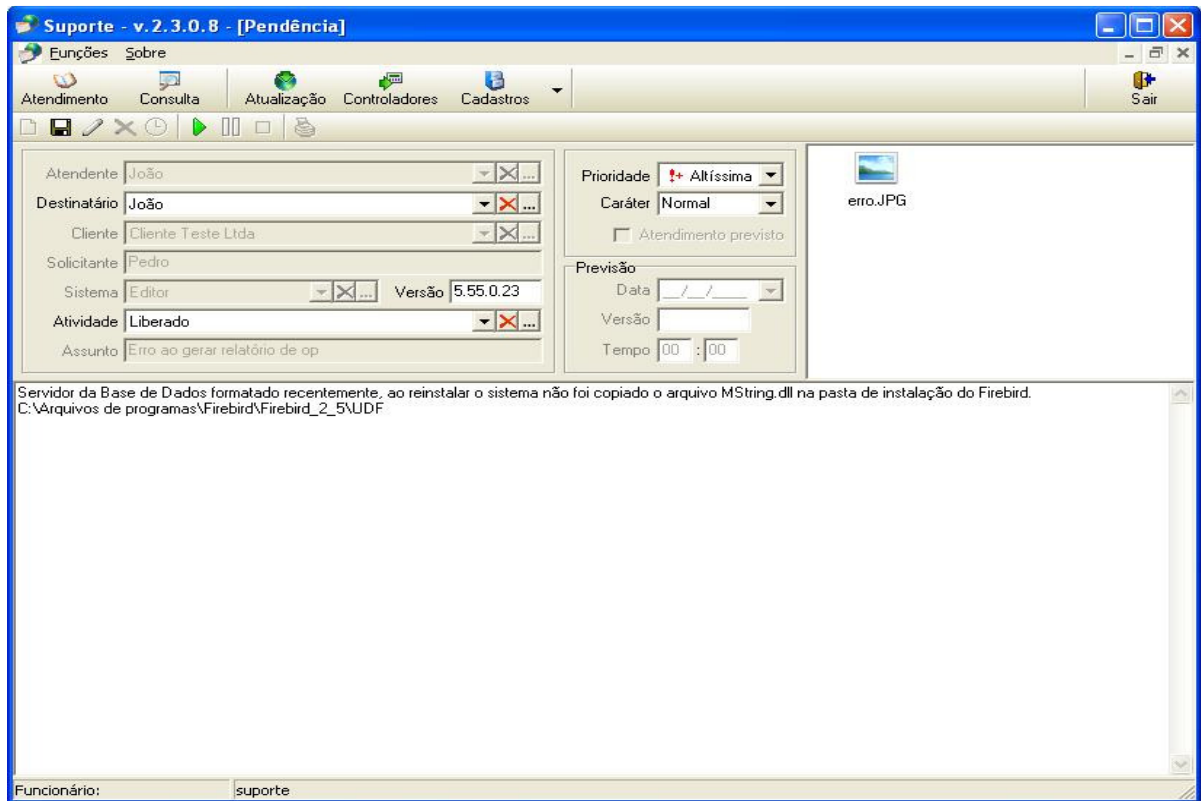


Figura 17 – Cadastro de Pendência

A indexação do atendimento pode ser feita de duas maneiras, uma delas é a partir da tela inicial, onde o usuário seleciona o atendimento e clica na aba Palavras-Chave. Outra forma consiste em perguntar ao usuário, ao término de um atendimento, se o mesmo deseja armazenar na base de casos a solução encontrada (Figura 18). Em caso afirmativo, é apresentada a tela de indexação com as palavras-chave. Nas duas formas apresenta-se a mesma tela que pode ser visualizada na Figura 19.

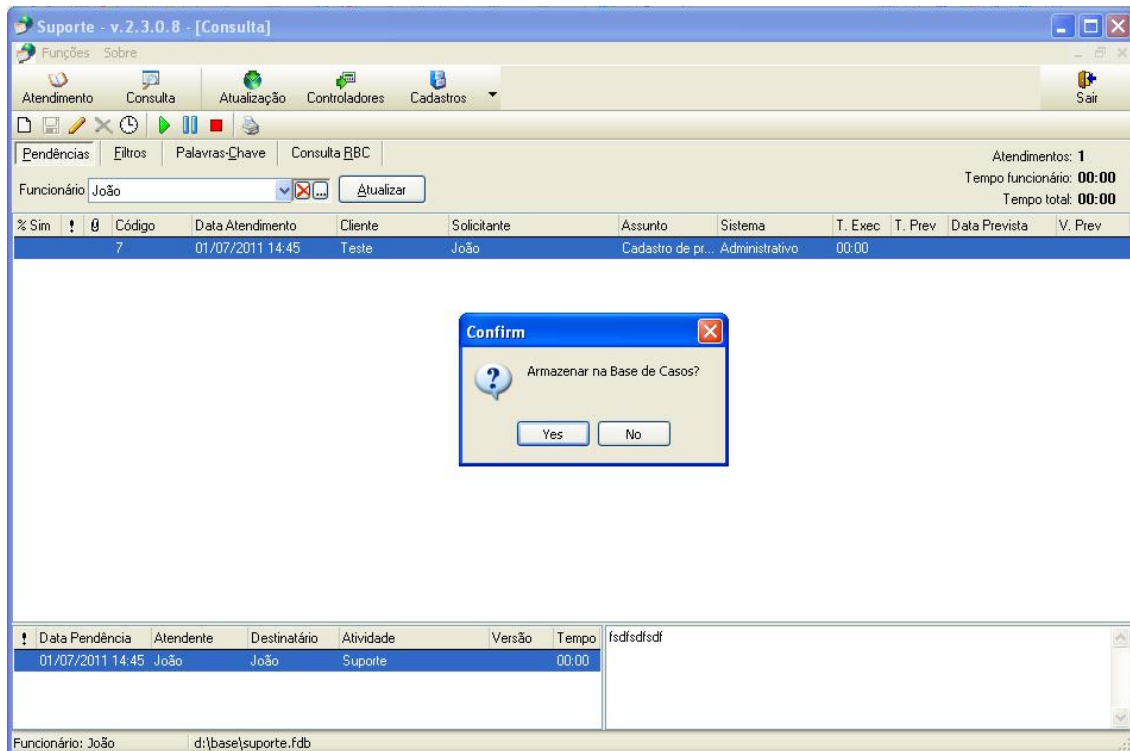


Figura 18 – Confirmar inserção na base de casos ao finalizar

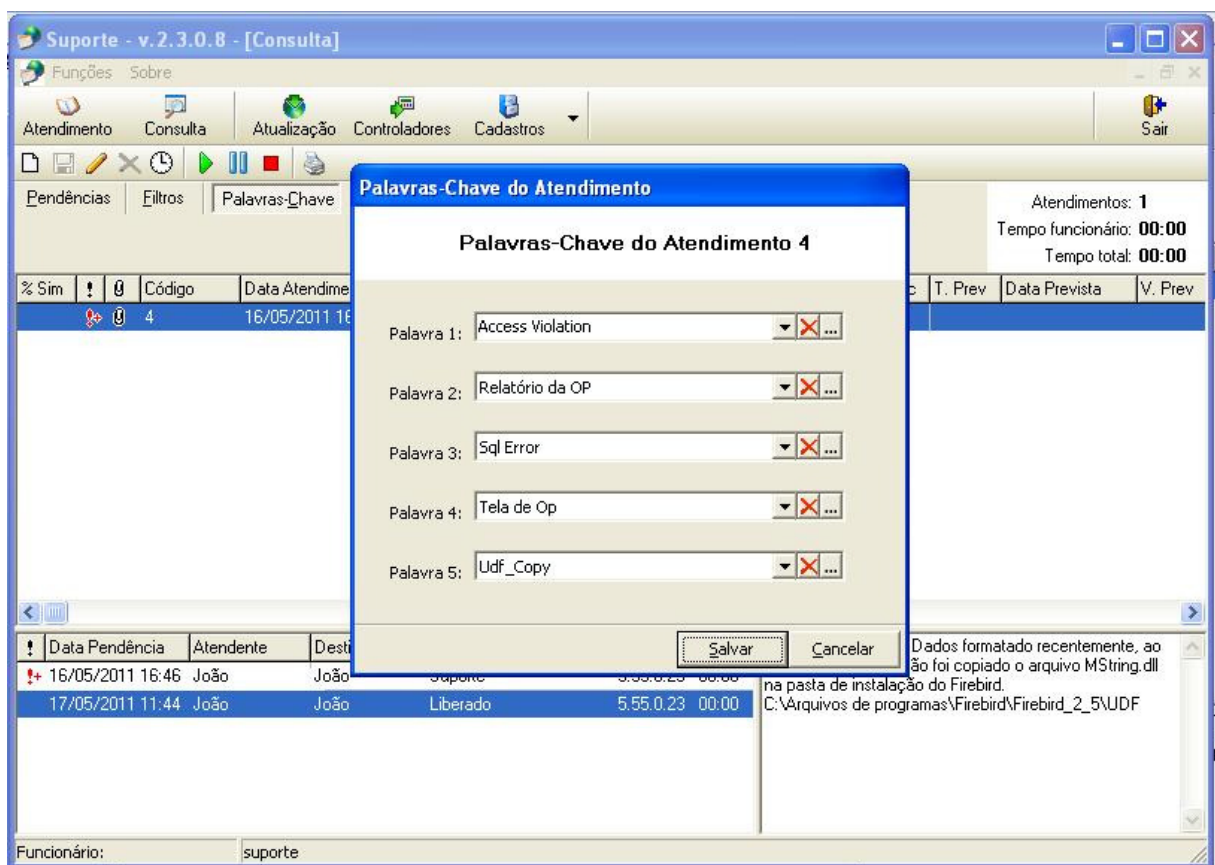


Figura 19 - Indexação com palavras-chave

3.6.4. Estudo de Caso

Na tela inicial do sistema encontra-se a aba Consulta RBC, onde através dela é possível consultar casos antigos na base. Para ilustrar esta operacionalidade será utilizado um estudo de caso simulando a ligação de um cliente, que reporta a seguinte situação, “*Estou tentando cadastrar um novo processo e esta gerando uma mensagem de erro, a mensagem é a seguinte: violation of PRIMARY or UNIQUE KEY constraint "PK_SUBPROCESSO" on table "PROCESSOSUBPROCESSO".*”

A Figura 20 mostra a tela utilizada na seleção das palavras para a consulta na base através da técnica RBC, é necessário informar todas as palavras sem repetições.

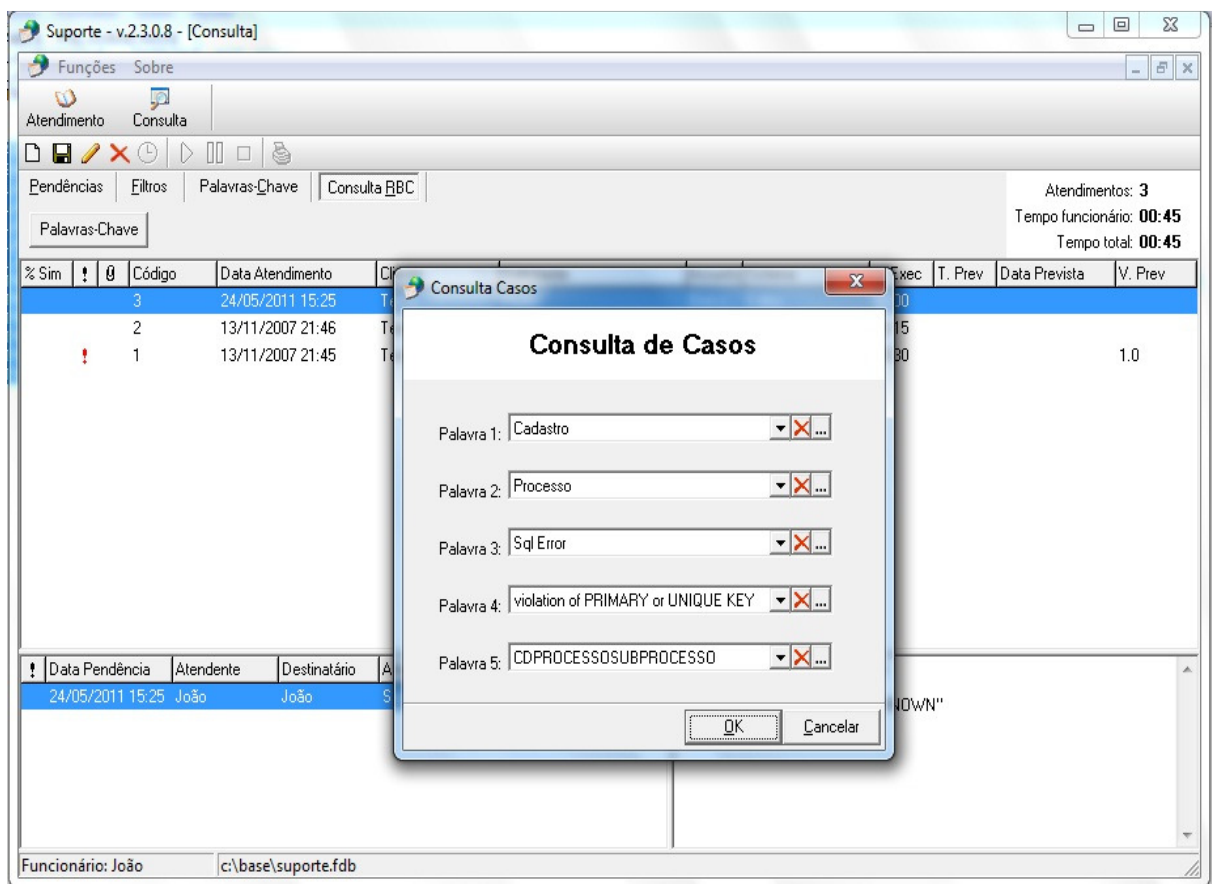


Figura 20 – Seleção de palavras-chave para busca RBC

Ao clicar em “ok” é feita a consistência das palavras informadas. Caso sejam válidas o sistema busca os atendimentos na base de casos, efetua o cálculo de similaridade com as

palavras da consulta e exibe na tela por ordem de similaridade. Selecionando o atendimento que desejar, o usuário tem acesso à descrição da resolução e pode verificar se ela se encaixa no problema atual. Tudo isto pode ser visualizado na Figura 21.

The screenshot shows a software window titled "Suporte - v.2.3.0.8 - [Consulta]". It features a menu bar with "Funções" and "Sobre", and a toolbar with icons for "Atendimento", "Consulta", "Atualização", "Controladores", and "Cadastros". Below the toolbar are tabs for "Pendências", "Filtros", "Palavras-Chave", and "Consulta BBC". A status bar on the right indicates "Atendimentos: 5", "Tempo funcionário: 02:03", and "Tempo total: 02:03".

The main area displays a table of search results with the following data:

% Sim	!	0	Código	Data Atendimento	Cliente	Solicitante	Assunto	Sistema	T. Exec	T. Prev	Data Prevista	V. Prev
100			5	24/05/2011 16:17	Teste	Manoel	Erro cadastro de processo	Editor	00:30			
80			1	13/11/2007 21:45	Teste	Paulo	Erro cadastro de Processo	Administrativo	00:30			1.0
40			3	24/05/2011 15:25	Teste	Pedro	Erro cadastro de cliente	Editor	00:00			
40			2	13/11/2007 21:46	Teste	João	Erro Cadastro de máquinas	Produção	00:40			
20			4	24/05/2011 16:02	Teste	Fabricia	Etapas não aparecem no subprocesso	Editor	00:23			

An arrow points from the text "Casos retornados pela consulta" to the table above. Below the table, the section "Pendências do atendimento selecionado" is shown, with an arrow pointing to a table of pending tasks:

!	Data Pendência	Atendente	Destinatário	Atividade	Versão	Tempo
	24/05/2011 16:17	João	João	Suporte	5.58.0.1	00:17
	24/05/2011 16:18	João	João	Suporte	5.58.0.1	00:13

To the right of this table, a text box contains the message: "Cliente está tentando inserir o mesmo subprocesso duas vezes no processo. Alertado que isto não pode ser feito no sistema." An arrow points from the text "Descrição do problema/solução" to this text box. At the bottom of the window, the status bar shows "Funcionário: João" and "c:\base\suporte.fdb".

Figura 21 – Resultado Busca RBC

A Figura 22 mostra trecho do código para o cálculo da similaridade de cada atendimento encontrado na base.

```

function TfrmConsulta.CalculaSimilaridade(cdAtendimento, CdPalavra1, CdPalavra2, CdPalavra3, CdPalavra4, CdPalavra5: integer): integer;
var
  i, TotIguais: integer;
  CalculoSimilaridade: double;
  AtendimentoPalavra: TAtendimentoPalavra;
  ListAtendimetoPalavraDb: TListAtendimentoPalavraDb;
  List: TListInt;
820 begin
  TotIguais := 0;
  ListAtendimetoPalavraDb := TListAtendimentoPalavraDb.Create(dmSuporte.SQLConnection);
  {Carrega palavras-chave do atendimento em questão}
  ListAtendimetoPalavraDb.Refresh(CdAtendimento);
  AtendimentoPalavra := ListAtendimetoPalavraDb.GetByCodigo(CdAtendimento);
  List := TListInt.Create;
  try
    {armazena as palavras-chave da busca em lista}
    List.Add(CdPalavra1);
830    List.Add(CdPalavra2);
    List.Add(CdPalavra3);
    List.Add(CdPalavra4);
    List.Add(CdPalavra5);
    {compara cada palavra informada com todas as palavras do atendimento}
    for i := 0 to List.Count - 1 do
      if (AtendimentoPalavra.CdPalavraChave1 = List[i]) or
        (AtendimentoPalavra.CdPalavraChave2 = List[i]) or
        (AtendimentoPalavra.CdPalavraChave3 = List[i]) or
        (AtendimentoPalavra.CdPalavraChave4 = List[i]) or
840        (AtendimentoPalavra.CdPalavraChave5 = List[i]) then
        TotIguais := TotIguais + 1;
    finally
      List.Free;
    end;
    {Efetua o calculo de similaridade}
    CalculoSimilaridade := TotIguais / 5 * 100;
    result := Trunc(CalculoSimilaridade);
  end;
end;

```

Figura 22 - Cálculo de similaridade do atendimento

3.7. RESULTADOS E DISCUSSÃO

Através de entrevistas informais com técnicos do setor de suporte ao usuário, foi possível perceber que o desenvolvimento deste trabalho permitiu agilizar o processo de resolução dos problemas relatados ao setor de suporte, através da tecnologia RBC. Sendo assim o objetivo desta ampliação foi implementar o paradigma de RBC no atual sistema, fazendo com que o atendente do setor de suporte ao cliente possa fornecer de forma mais rápida e eficiente uma resolução do problema em questão, e também armazenar o conhecimento que ele adquiriu ao resolver uma situação nova.

Como resultado da implantação do sistema desenvolvido realizou-se um acompanhamento junto aos técnicos do setor de suporte para avaliar a viabilidade do sistema. Foi constatado que:

- a) diminuição no tempo total para resolução dos atendimentos;
- b) mais atendimentos atendidos em menos tempo;
- c) conseqüente diminuição de custo custo para a resolução dos problemas.

Isto é possível pois grande parte das situações relatadas são similares as já cadastradas previamente no sistema, e com a consulta por RBC esses atendimentos similares são apresentados como sugestões de resolução para os atendentes. Não foi realizada uma comparação estatística, mas a melhora é visivelmente perceptível.

4. CONCLUSÕES

Durante o presente trabalho foi demonstrada a utilização de um paradigma da Inteligência Artificial (IA) o Raciocínio Baseado em Casos (RBC), como uma ferramenta de auxílio aos técnicos de suporte ao cliente para fornecer soluções rápidas e eficazes.

Os Sistemas de RBC possibilitam a utilização do conhecimento especialista no apoio a decisões, pois eles permitem a extração, organização e reuso de conhecimento utilizado para tomada de decisões no passado, e permitindo o seu aperfeiçoamento.

A utilização dessa técnica limita-se ao acesso à base de dados completa, correta e confiável que contenha entre as informações armazenadas, a descrição completa do problema e das soluções que foram aplicadas em algum momento. Esta é a matéria prima inicial e básica para a construção de sistemas baseados em casos.

Uma característica importante que faz com que o RBC seja eficaz no sistema atual, é de possuir uma metodologia simples para captura de casos. A base de dados usada na aplicação é mantida pelos próprios usuários e o sistema é alimentado com novos casos constantemente.

Com a alta competitividade as empresas se destacam pela qualidade do atendimento ao cliente. Em empresas de software essa realidade se reflete no suporte ao usuário, ao ligar para o suporte o cliente espera que seu problema seja resolvido ou sua dúvida sanada de forma rápida e eficaz.

Porém, este é um setor que não recebe muita atenção, acarretando no não uso de um sistema para compartilhar o conhecimento adquirido pelos atendentes. Isto leva a demora para resolução de problemas, explicação de dúvidas e treinamento de novos funcionários.

Notou-se que a utilização da técnica RBC em sistemas para controle de atendimentos pode trazer vários benefícios, o mais importante talvez seja que a partir de uma base de casos o usuário pode rapidamente diagnosticar e resolver problemas, antes resolvido por outro atendente. Isso resulta em uma redução no custo do suporte técnico. Outro benefício que pode ser citado é a externalização do conhecimento, possibilitando que novos técnicos possam resolver os mesmos problemas que os técnicos que já trabalham a mais tempo na empresa. Estes novos funcionários podem também receber treinamento pesquisando pelos problemas e suas soluções.

Portanto, os objetivos deste trabalho, de integrar a tecnologia RBC no sistema de informação utilizado pelo setor de suporte, disponibilizando um sistema de apoio à resolução

de problemas e validar este sistema através de um estudo de caso foram alcançados.

4.1. EXTENSÕES

A seguir apresenta-se algumas sugestões para o desenvolvimento de trabalhos futuros:

- a) criar uma versão *web* do aplicativo incluindo a busca através de RBC;
- b) implementar outra técnica de recuperação e comparar com a utilizada neste trabalho e avaliar qual é mais veloz;
- c) implementação de *text mining* para pré-classificação automática das palavras-chave;
- d) determinar a frequência de uso do caso para enviar para correção os casos que são mais usados como solução na busca RBC.

REFERÊNCIAS BIBLIOGRÁFICAS

AAMODT, A; PLAZA, E. **Case-Based Reasoning: Foundational Issues, Methodological Variations, and Systems Approaches**. In: Artificial Intelligence Communications, 7., 1994, Catalonia: Proceedings... Catalonia: IOS Press, 1994. p.39-59.

ABEL, Mara. **Um estudo sobre Raciocínio Baseado em Casos**. Porto Alegre, 1996.

Disponível em:

<http://www.inf.ufrgs.br/bdi/administrator/components/com_jresearch/files/publications/CBR-TI60.pdf>. Acesso em: 05 jun. 2011.

BARRONE, Dante (Org.). **Sociedades Artificiais: a nova fronteira da inteligência das máquinas**. Porto Alegre: Bookman, 2003.

DALFOVO, Oscar; AMORIM, Sammy Newton. **Quem tem informação é mais competitivo**. Blumenau: Acadêmica, 2000.

DELPIZZO, Vanessa L. Francalacci. **Prescrição de atividades físicas através do uso da inteligência artificial**. Florianópolis, 1997. Tese (Doutorado em Engenharia de Produção) - Programa de Pós-Graduação em Engenharia de Produção, Universidade Federal de Santa Catarina, Florianópolis. Disponível em: <<http://www.eps.ufsc.br/disserta98/delpizzo/>>. Acesso em: 29 ago. 2010

EHLERT, Gisele. **Sistema aplicativo para controle de fretes utilizando raciocínio baseado em casos**. 2004. 63 f. Trabalho de Conclusão de Curso (Bacharel em Ciências da Computação) – Centro de Ciências Exatas e Naturais, Universidade Regional de Blumenau, Blumenau.

KOLODNER, Janet L ed. **Case-based learning**. Boston : Kluwer, 1993.

KIENEN, Paulo César. **Sistema de informação aplicado na advocacia utilizando raciocínio baseado em casos**. 2003. 66 f. Trabalho de Conclusão de Curso (Bacharel em Ciências da Computação) – Centro de Ciências Exatas e Naturais, Universidade Regional de Blumenau.

LAUDON, Kenneth C.; LAUDON, Jane Price. **Sistemas de informação com Internet**. Rio de Janeiro: LTC, 1999.

LEE, Rosina Weber, **Pesquisa jurisprudencial inteligente**. Florianópolis, 1998. Tese (Doutorado em Engenharia de Produção) - Programa de Pós-Graduação em Engenharia de Produção, Universidade Federal de Santa Catarina, Florianópolis.

LIRA, Alan Augusto, **Sistema de Informação para formação de preços e controle de estoque aplicado a empresas de esquadrias de alumínio, utilizando raciocínio baseado em casos**. 2002. 71 f. Trabalho de Conclusão de Curso (Bacharel em Ciências da Computação) – Centro de Ciências Exatas e Naturais, Universidade Regional de Blumenau, Blumenau.

MENDES, J.V.; ESCRIVÃO FILHO, E. **Sistemas integrados de gestão ERP em pequenas empresas: um confronto entre o referencial teórico e a prática empresarial**. São Paulo. vol.9, nº 3, p.277-296. Disponível em:
<http://www.scielo.br/scielo.php?script=sci_arttext&pid=S0104-530X2002000300006>.
Acesso em: 30 out. 2010.

OLIVEIRA, Djalma. **Sistemas de informações gerenciais: estratégicas, táticas, operacionais**. São Paulo: Atlas, 1996.

STAIR, Ralph M. **Princípios de sistemas de informação: uma abordagem gerencial**. Rio de Janeiro: LTC, 1998.

WANGENHEIM, Christiane Gresse von; WANGENHEIM, Aldo von. **Raciocínio baseado em casos**. Barueri: Manoele, 2003.

APÊNDICE A – Detalhamento dos casos de uso.

É apresentado um detalhamento dos casos de uso (UC01, UC02, UC03) previstos no diagrama apresentado na seção 3.3.1.

No Quadro 19 apresenta-se o caso de uso "UC01 - Indexar Atendimento".

Nome do Caso de Uso	Indexar Atendimento
Descrição	Usuário seleciona um atendimento e acessa a aba Palavras-Chave ou finaliza um atendimento e escolhe opção para salvar caso na base.
Ator	Usuário
Pré-condição	Usuário deve fazer <i>login</i> no sistema e selecionar um atendimento; Sistema deve ter pelo menos uma palavra-chave cadastrada; Sistema deve ter pelo menos um atendimento em aberto;
Fluxo principal	<ol style="list-style-type: none"> 1. Sistema exibe as palavras-chave para o atendimento selecionado; 2. Usuário seleciona palavras-chave para indexação; 3. Usuário Confirma clicando em Salvar.
Cenário – Visualização	Sistema exibe os atendimentos em aberto para o usuário logado
Cenário – Campo Faltante	1. No passo 3 caso o usuário deixe em branco um campo, o sistema valida esse campo e exibe uma mensagem adequada informando que o preenchimento é obrigatório e não salva as alterações.
Cenário – Campo Duplicado	1. No passo 3 caso o usuário selecione a mesma palavra mais de uma vez, o sistema valida esse campo e exibe uma mensagem adequada informando que o não é permitido informar campos duplicados e não salva as alterações.
Cenário – Cancelar Indexação	1. Após o passo 1, a qualquer momento o usuário pode clicar em cancelar para cancelar a indexação, assim o sistema não registra as alterações e volta para tela principal.
Pós-condição	Usuário indexou um caso.

Quadro 19 – Descrição do caso de uso Manter Caso

No Quadro 20 apresenta-se o caso de uso "UC02 - Manter Palavras-Chave".

Nome do Caso de Uso	Manter Palavras-Chave
Descrição	Usuário acessa o botão Cadastrar Palavras-Chave para manter dados de Palavras-Chave. Serão mantidos os dados: Descrição Palavra.
Ator	Usuário
Pré-condição	Usuário deve fazer <i>login</i> no sistema.
Fluxo principal	<ol style="list-style-type: none"> 1. Sistema informa as palavras cadastradas; 2. Usuário opta por editar, apagar ou cadastrar uma palavra.

Cenário – Visualização	Sistema mostra para o usuário os registros palavras-chave cadastrados.
Cenário – Edição	<ol style="list-style-type: none"> 1. Sistema mostra registros cadastrados; 2. Usuário seleciona um registro para edição; 3. Sistema mostra a descrição palavra para edição; 4. Usuário altera registro e seleciona opção para atualizar descrição da palavra; 5. Sistema mostra os registros cadastrados com o registro alterado.
Cenário – Inclusão	<ol style="list-style-type: none"> 1. Sistema mostra registros cadastrados; 2. Usuário inclui um novo registro; 3. Sistema mostra os registros cadastrados.
Cenário – Exclusão	<ol style="list-style-type: none"> 1. Sistema mostra registros cadastrados; 2. Usuário seleciona um registro para exclusão; 3. Sistema exclui o registro e mostra os registros restantes.
Pós-condição	Usuário visualizou, editou, apagou ou cadastrou uma palavra-chave.

Quadro 20 – Descrição do caso de uso Manter Palavras-Chave

No Quadro 21 apresenta-se o caso de uso "UC03 - Efetuar Busca por Similaridade".

Nome do Caso de Uso	Efetuar Busca Por Similaridade
Descrição	Usuário acessa o botão Buscar Casos na tela inicial para buscar casos na base.
Ator	Usuário
Pré-condição	Usuário deve fazer <i>login</i> no sistema.
Fluxo principal	<ol style="list-style-type: none"> 1. O sistema mostra tela para escolha das palavras-chave; 2. Usuário seleciona palavras-chave para efetuar a busca; 3. Sistema exibe resultado da busca realizada.
Cenário – Busca com Sucesso	<ol style="list-style-type: none"> 1. Sistema mostra registros de casos ordenados por percentual de similaridade com as palavras-chave selecionadas na busca; 2. Usuário seleciona um caso; 3. Sistema Exibe as pendências do caso selecionado.
Cenário – Busca Sem Retorno	<ol style="list-style-type: none"> 1. Sistema Exibe mensagem informando que não encontrou caso cadastrado que tenha similaridades com as palavras-chave selecionadas; 2. Sistema volta à tela para escolha de palavras-chave.
Pós-condição	Usuário efetuou busca por similaridade.

Quadro 21 – Descrição do caso de uso Efetuar Busca por Similaridade