

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

UM PROTÓTIPO PARA DIAGNÓSTICO DE DEFEITOS EM
SISTEMAS DE AUTOMAÇÃO DE MOTÉIS

TACIANA MARIA GRETTTER

BLUMENAU
2015

2015/2-22

TACIANA MARIA GREYTER

**UM PROTÓTIPO PARA DIAGNÓSTICO DE DEFEITOS EM
SISTEMAS DE AUTOMAÇÃO DE MOTÉIS**

Trabalho de Conclusão de Curso apresentado ao curso de graduação em Sistemas de Informação do Centro de Ciências Exatas e Naturais da Universidade Regional de Blumenau como requisito parcial para a obtenção do grau de Bacharel em Sistemas de Informação.

Prof. Aurélio Faustino Hoppe, Mestre - Orientador

**BLUMENAU
2015**

2015/2-22

UM PROTÓTIPO PARA DIAGNÓSTICO DE DEFEITOS EM SISTEMAS DE AUTOMAÇÃO DE MOTÉIS

Por

TACIANA MARIA GREYTER

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

Presidente: _____
Prof. Aurélio Faustino Hoppe, Mestre – Orientador, FURB

Membro: _____
Prof. Dalton Solano dos Reis, Mestre – FURB

Membro: _____
Prof. Marcel Hugo, Mestre – FURB

Blumenau, 14 de dezembro de 2015

Dedico esse trabalho aos meus pais, que sempre prezaram pelos meus estudos e me apoiaram até o final.

AGRADECIMENTOS

A Deus pelo seu imenso amor e graça, assim como pelas oportunidades a mim colocadas.

À minha família, pelos valores de vida ensinados que me permitem compreender a importância de finalizar essa etapa da minha vida bem como pela estrutura fornecida para chegar até aqui.

Aos meus amigos pela compreensão e apoio e principalmente pelos empurrões em momentos de dificuldade.

Ao meu orientador, Aurélio Faustino Hoppe, por todo o auxílio prestado e pela confiança na conclusão deste trabalho.

Aos professores do Departamento de Sistemas e Computação da Universidade Regional de Blumenau por suas contribuições durante os semestres letivos.

“Just have a little Faith.”

Michael Scofield

RESUMO

Este trabalho apresenta um protótipo de sistema especialista para auxiliar o atendimento e diagnóstico de defeitos em sistemas de automação de motéis, visando agilizar o processo de manutenção. O protótipo permite que técnicos do próprio local, com conhecimentos básicos, possam solucionar problemas mais complexos através da tomada de decisões baseada no conhecimento do engenheiro responsável pela instalação e manutenção do sistema de automação. Para o desenvolvimento foram utilizadas as linguagens PHP, Javascript e HTML. Os resultados obtidos demonstram que o protótipo desenvolvido soluciona as principais dúvidas e os principais problemas atendidos pelo engenheiro.

Palavras-chave: Sistema especialista. Sistema de automação. Diagnóstico de defeitos.

ABSTRACT

This work presents an expert system prototype to assist the services and defects diagnosis in motels automation systems in order to speed up the process of maintenance. The prototype allow technicians at the place, with basic knowledge, to solve more complex problems by making decisions based on knowledge of the engineer responsible for the installation and maintenance of the automation system. For development, it was used PHP, JavaScript and HTML languages. The results show that the prototype addresses the key questions and the main problems met by the engineer.

Key-words: Expert system. Automation system. Defect diagnosis.

LISTA DE FIGURAS

Figura 1 – Árvore de decisão fictícia	17
Figura 2 - Tela <i>Front End</i> para controle do funcionário	20
Figura 3 - Painel de controle do sistema de automação	20
Figura 4 - Diagrama de Atividade do processo de Entrada.....	22
Figura 5 - Diagrama de Atividade do processo de Saída	23
Figura 6 - Tela de Consulta de Casos	27
Figura 7 - Tela de Cadastro de Novos Casos	27
Figura 8 - Tela principal do sistema com as 6 áreas de expertise	29
Figura 9 - Diagrama de casos de uso	33
Figura 10 - Modelo Entidade-Relacionamento	34
Figura 11 - Processos básicos do protótipo	35
Figura 12 - Menu de módulos em cascata	35
Figura 13 - Tela de cadastro de regra após escolher um módulo	36
Figura 14 - Tela de cadastro com respostas e contingência cadastradas	37
Figura 15 - Árvore da regra cadastrada	37
Figura 16 - Tela de consulta após escolher um módulo	39
Figura 17 - Solução exibida na tela de consulta	42
Figura 18 - Contingência exibida na tela de consulta.....	43

LISTA DE QUADROS

Quadro 1 - Requisitos funcionais	32
Quadro 2 - Requisitos não funcionais	32
Quadro 3 - Trecho do código para receber informações cadastradas.....	38
Quadro 4 - Trecho do código para cadastro da regra no banco.....	38
Quadro 5 - Trecho de código que faz a consulta de regra no banco.....	39
Quadro 6 - Trecho de código com informações da tela de consulta.....	40
Quadro 7- Função <code>proximaPergunta()</code> em consulta.....	41
Quadro 8 - Continuação da função <code>proximaPergunta()</code> em consulta.....	42
Quadro 9 - Respostas quanto à lista de tarefas	44
Quadro 10 - Respostas do engenheiro em relação ao questionário de usabilidade.....	45
Quadro 11 - Respostas quanto à lista de tarefas	46
Quadro 12 - Respostas do usuário em relação ao questionário de usabilidade.....	47
Quadro 13 – Comparação de trabalhos correlatos.....	47
Quadro 14 - Descrição dos casos de uso	52
Quadro 15 - Tabela de área.....	54
Quadro 16 - Tabela de módulo.....	54
Quadro 17 - Tabela de usuário	54
Quadro 18 - Tabela de regra	55
Quadro 19 - Tabela de variável.....	55
Quadro 20 - Lista de tarefas para administrador	56
Quadro 21 - Questionário de usabilidade para administrador.....	58
Quadro 22 - Lista de tarefas para usuário cliente.....	60
Quadro 23 - Questionário de usabilidade para usuário cliente.....	61

LISTA DE ABREVIATURAS E SIGLAS

CASE – *Computer-Aided Software Engineering*

CSS – *Cascading Style Sheets*

EA – Enterprise Architect

HTML – HyperText Markup Language

IA – Inteligência Artificial

LED – *Light-emitting diode*

MER – Modelo Entidade-Relacionamento

PHP – Personal Home Page (Hypertext Preprocessor)

RF – Requisito Funcional

RNF – Requisito Não-Funcional

SE – Sistema Especialista

SQL – Structured Query Language

UC – *Use Case*

UML – *Unified Modeling Language*

URL – *Uniform Resource Location*

SUMÁRIO

1 INTRODUÇÃO.....	13
1.1 OBJETIVOS.....	14
1.2 ESTRUTURA.....	15
2 FUNDAMENTAÇÃO TEÓRICA	16
2.1 SISTEMA ESPECIALISTA	16
2.2 ÁRVORE DE DECISÃO	17
2.3 SISTEMA DE AUTOMAÇÃO PARA MOTÉIS	18
2.4 SISTEMAS CRIADOS PELA CONTROTEC	18
2.5 SISTEMA ATUAL	24
2.6 TRABALHOS CORRELATOS	25
2.5.1 Protótipo de um sistema especialista para atendimento a ocorrências em transformadores de potência	25
2.5.2 Sistema de apoio para o diagnóstico de defeitos em equipamentos eletrônicos aplicado a oficinas eletrônicas utilizando raciocínio baseado em casos	26
2.5.3 Sistema Especialista de apoio à decisão para o diagnóstico de falhas em aeronaves	28
2.5.4 Comparação entre as características dos trabalhos relacionados	29
3 DESENVOLVIMENTO DO PROTÓTIPO	31
3.1 LEVANTAMENTO DE INFORMAÇÕES	31
3.2 ESPECIFICAÇÃO	32
3.2.1 Requisitos Funcionais e Não Funcionais	32
3.2.2 Casos de Uso	32
3.2.3 Modelo Entidade-Relacionamento	34
3.3 IMPLEMENTAÇÃO	34
3.3.1 Técnicas e ferramentas utilizadas.....	35
3.3.2 Etapas da implementação	35
3.4 RESULTADOS E DISCUSSÕES.....	43
3.4.1 Usabilidade do protótipo	43
3.4.2 Comparação com trabalhos correlatos e discussões.....	47
4 CONCLUSÕES.....	49
4.1 LIMITAÇÕES	49
4.2 EXTENSÕES	49

REFERÊNCIAS	51
APÊNDICE A – DESCRIÇÃO DOS CASOS DE USO	52
APÊNDICE B – DICIONÁRIO DE DADOS.....	54
APÊNDICE C – ROTEIRO E QUESTIONÁRIO DE AVALIAÇÃO DE USABILIDADE.....	56

1 INTRODUÇÃO

Pode-se dizer que o desejo de controlar os processos industriais acompanha o homem desde a criação das primeiras máquinas. Até a década de 1940, as plantas eram operadas manualmente por um grande número de operadores, os quais valiam-se de alguns poucos instrumentos mecânicos elementares que realizavam controle local (GUTIERREZ; PAN, 2008).

O termo Automação, do inglês *automation*, foi um termo inventado pelo marketing da indústria de equipamentos da década de 1960. O neologismo, sem dúvida, buscava enfatizar a participação de computadores no controle automático industrial (MORAES; CASTRUCCI, 2007).

Automação descreve um conceito amplo, envolvendo um conjunto de técnicas de controle, das quais é criado um sistema ativo, capaz de fornecer a resposta adequada em função das informações que recebe do processo em que está atuando. Dependendo das informações, o sistema calculará a melhor ação corretiva a ser executada (WEG, 2002).

Para Pinheiro (2004),

por automação entende-se a capacidade de se executar comandos, obter medidas, regular parâmetros e controlar funções automaticamente, sem a intervenção humana. Automação também é sinônimo de integração, ou seja, da função mais simples à mais complexa, existem um ou mais sistemas que permitem que um dispositivo seja controlado de modo inteligente, tanto individualmente quanto em conjunto, visando alcançar um maior conforto, informação e segurança.

Atualmente, a automação já está embutida em grande parte das atividades da vida cotidiana, presente nas diversas máquinas capazes de agirem de maneira automática. Assim como muitas outras tecnologias, não se nota a atuação desses sistemas, mas é possível encontrá-los em processos industriais, processos de compra e venda, transporte de bens, no trânsito, nas residências e até mesmo no gerenciamento do fluxo de motéis.

Em meio à maré de problemas e desafios a serem resolvidos pelos especialistas, o mundo dos negócios, bem como outros ramos em que o uso de conhecimento de especialista é decisivo para tomada de decisões, o uso de sistemas especialistas tem se despontado como ótimas saídas para solução de diversos problemas, que até então seriam resolvidos apenas por especialistas (TARSO, 2005, p. 2).

Tem sido buscada cada vez mais a utilização de sistemas especialistas para substituir o atendimento antes feito somente por especialistas humanos. Isso pode ser visto de forma negativa quando se pensa na substituição do especialista, porém traz muitos benefícios quando se pensa que um sistema especialista pode estar sempre disponível para o atendimento, diferente de uma pessoa. Além disso, o conhecimento fica armazenado em uma base de conhecimento, não dependendo mais do ser humano, assim como no caso dos

sistemas de automação, que se busca eliminar a intervenção humana em determinados processos.

No ramo de sistemas de automação há mais de 5 anos, a empresa Controtec tem como principal foco a criação e instalação de sistemas de automação para hotéis e motéis, sempre buscando se responsabilizar também pela manutenção dos mesmos.

No cenário de manutenção dos sistemas instalados em motéis, há momentos em que a correção do problema se faz urgente, devido à indisponibilidade que pode ser causada, gerando prejuízos ao estabelecimento. Muitos dos defeitos que aparecem podem ser considerados básicos e já conhecidos pelo engenheiro que criou o sistema. Nesses casos, o engenheiro procura orientar o técnico do local com o passo a passo para a correção. Em situações mais complexas onde é necessária a visita de um técnico da empresa, é passada uma solução paliativa para diminuir a indisponibilidade até o momento da visita.

Diante disto, o presente trabalho apresenta o desenvolvimento de um sistema especialista para realizar o atendimento de manutenção básica em sistemas de automação de motéis. Este procedimento é atualmente realizado pelo engenheiro especialista e pretende-se transferir esse conhecimento para uma base de conhecimento, a fim de poder ser feito diretamente no sistema especialista, permitindo que o engenheiro possa focar em outros projetos.

1.1 OBJETIVOS

O objetivo geral deste trabalho é o desenvolvimento de um protótipo que auxilie o atendimento e diagnóstico de defeitos em sistemas de automação de motéis instalados pela Controtec.

Os objetivos específicos do trabalho são:

- a) construir uma base de conhecimento com defeitos que podem ocorrer no sistema de automação;
- b) implementar um Sistema Especialista para emitir o diagnóstico de falhas, as possíveis causas e as providências a serem tomadas com relação ao estado operacional do sistema de automação;
- c) disponibilizar um mecanismo para recomendar soluções de contingência até que a visita do técnico seja realizada.

1.2 ESTRUTURA

No primeiro capítulo tem-se a introdução ao tema principal deste trabalho com a apresentação da justificativa e dos objetivos.

No segundo capítulo apresentam-se a fundamentação teórica pesquisada sobre sistema especialista, sistema de automação para motéis, sistemas criados pela Controtec, sistema atual, além de trabalhos correlatos.

O terceiro capítulo apresenta o desenvolvimento do protótipo iniciando-se com o levantamento de informações, tendo na sequência a especificação com os requisitos da aplicação, diagramas de caso de uso e modelo de entidade relacionamento, em seguida a descrição da implementação com técnicas e ferramentas e etapas da implementação, e por fim os resultados e discussão do trabalho.

No quarto capítulo tem-se as conclusões deste trabalho bem como apresentam-se sugestões para trabalhos futuros.

2 FUNDAMENTAÇÃO TEÓRICA

Este capítulo aborda assuntos a serem apresentados nas seções a seguir, tais como sistema especialista, sistema de automação para motéis, sistemas criados pela Controtec, sistema atual, além dos trabalhos correlatos.

2.1 SISTEMA ESPECIALISTA

Segundo Ribeiro (1987), um sistema especialista é aquele que é projetado e desenvolvido para atender a uma aplicação determinada e limitada do conhecimento humano. É capaz de emitir uma decisão, com apoio e conhecimento justificado, a partir de uma base de informações tal qual um especialista de determinada área do conhecimento humano.

Para o sistema especialista, constrói-se uma base de conhecimento a partir das informações providas pelo especialista no assunto. Essa base terá um conhecimento específico do tema, similar ao conhecimento do especialista, para que possa prover as mesmas orientações quando for consultado. A partir dessa base, consegue fazer escolhas e justificá-las apoiando-se no conhecimento armazenado. Todo esse processo consegue substituir grande parte do atendimento que antes seria feito pelo especialista.

De acordo com Tarso (2005, p. 6),

um sistema especialista consiste num conjunto de componentes integrados e relacionados incluído um banco de conhecimentos, um motor de inferência, um recurso de explicação, um recurso de conhecimento e uma interface com o usuário. O usuário interage com a interface com o usuário, que interage com o motor de inferência. O motor de inferência interage com os demais componentes do sistema especialista.

Para implementação de um sistema especialista, uma das ferramentas existentes é o Expert SINTA que, segundo Paula (2013, p.1),

trata-se de uma ferramenta computacional que utiliza técnicas de Inteligência Artificial para geração automática de sistemas especialistas. Esta ferramenta utiliza um modelo de representação do conhecimento baseado em regras de produção e probabilidades, tendo como objetivo principal simplificar o trabalho de implementação de sistemas especialistas através do uso de uma máquina de inferência compartilhada, da construção automática de telas e menus, do tratamento probabilístico das regras de produção e da utilização de explicações sensíveis ao contexto da base de conhecimento modelada.

Como vantagens do sistema especialista, Weiss e Kulikowski (1988) mencionam "a disseminação do conhecimento de especialista, rara e de alto custo, e o uso mais eficaz e eficiente do especialista". Um sistema especialista pode ser muito vantajoso quando se pensa na otimização do tempo do especialista, que passa a ser livre para focar em outras atividades, gerando maior produtividade. Além disso, não há mais essa dependência do especialista,

podendo ser resolvido problemas mesmo em casos de dificuldade de contato com o mesmo, por exemplo. Isso só é possível porque o conhecimento está disseminado e mais acessível.

2.2 ÁRVORE DE DECISÃO

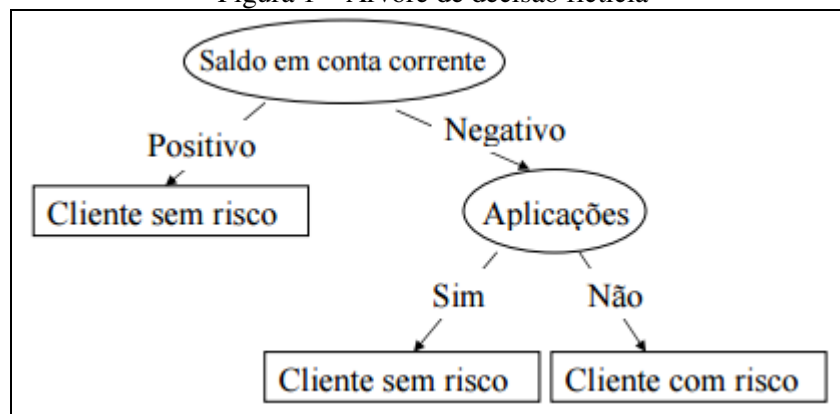
Um dos modelos muito utilizado para implementação de sistemas especialistas é a árvore de decisão, que segundo Pozzer (2006), é similar a regras *if-then*. As árvores de decisão tomam como entrada uma situação descrita por um conjunto de atributos e retornam então uma decisão.

Uma árvore de decisão é uma estrutura que pode ser utilizada para, por meio de uma simples regras de decisão, dividir sucessivamente uma grande coleção de registros em conjuntos menores (HOSOKAWA, 2011).

Conforme explica Pozzer (2006), a árvore de decisão chega a sua decisão pela execução de uma sequência de testes. Cada nó interno da árvore corresponde a um teste do valor de uma das propriedades, e os ramos deste nó são identificados com os possíveis valores do teste. Cada nó folha da árvore especifica o valor de retorno se a folha for atingida.

A Figura 1 mostra um exemplo de árvore de decisão, onde contém nós de decisão, que definem algum teste sobre o valor de um atributo específico, com um ramo para cada um dos valores possíveis do teste. Esses nós se repetem até que se chegue em uma folha, sendo o fim de um ramo, que representa uma resposta para a situação sendo analisada.

Figura 1 – Árvore de decisão fictícia



Fonte: Von Zuben e Attux (2010?).

Para percorrer uma árvore, podem ser utilizados diversos processos de busca, dentre eles a busca em profundidade e a busca por largura. A busca em profundidade escolhe um caminho na árvore de decisão e o percorre até o final, onde irá encontrar o nó folha. Caso esse nó não seja a solução para o problema, irá voltar para o nó pai e percorrer o próximo caminho até o final chegando no próximo nó folha. Já a busca por largura vai testando gradativamente cada possibilidade encontrada, sem se aprofundar demais em cada um dos caminhos. São

avaliados primeiro todos os nós de um determinado nível, antes de prosseguir para a avaliação dos nós do próximo nível.

2.3 SISTEMA DE AUTOMAÇÃO PARA MOTÉIS

Automação é um sistema de equipamentos eletrônicos e/ou mecânicos que controlam seu próprio funcionamento, quase sem a intervenção do homem. O homem vem tentando diminuir as atividades manuais desde a pré-história, quando começou-se a mecanizar certos procedimentos utilizando invenções como a roda (PINTO, 2005). A automação passou a ter seu foco quando os sistemas de produção agrário e artesanal transformaram-se em industrial. Pinto (2005) também aponta que foi a partir daí que cada vez mais se buscam inovações tecnológicas para aumentar a produção e produtividade. Com a consolidação da automação industrial, surgiram outros tipos de automação, como automação comercial e predial, sempre buscando aumentar a eficiência. Pinheiro (2004) explica que "os vários ramos de automação têm em comum os mesmos princípios de controle, utilizando softwares e hardwares, controladores lógicos e linguagens de programação, além dos mais diversos tipos de dispositivos sensores e atuadores." Atualmente, a automação já se expandiu para diversos outros ramos e está presente em diversas atividades do dia a dia do homem.

Um ramo que recebeu a atenção de sistemas de automação foram os motéis, devido a necessidade de controlar partes individuais de forma sincronizada. Com a introdução de sistemas de automação, tornou-se possível gerenciar a chegada e saída de hóspedes de forma mais organizada, mantendo a discricção necessária e valorizando assim o estabelecimento. Além disso, criou uma sincronia para que atividades como abrir portão de entrada, abrir portão da garagem do apartamento e liberar energia do apartamento possam ocorrer de forma simultânea, agilizando o processo de chegada do hóspede e gerando então maior satisfação. Pode-se mencionar, além das melhorias específicas, o avanço da qualidade do atendimento como um todo, o que é possível analisar também em outras áreas que adquirem sistemas de automação.

2.4 SISTEMAS CRIADOS PELA CONTROTEC

Conforme informado em entrevista por Arnildo Gretter¹, a empresa Controtec iniciou seu trabalho com sistemas de automação para motéis quando viu a necessidade de seus

¹ Entrevista concedida pelo engenheiro eletricista Arnildo Gretter, responsável pela empresa Controtec em Blumenau/SC à aluna Taciana Gretter, do curso de Sistemas de Informação da FURB, em 04/04/2015.

clientes, que na época, faziam o controle de acesso através do uso de chaveiros, onde o hóspede recebia um chaveiro na entrada e o utilizava para ter acesso ao apartamento. Esse controle funcionou por um tempo, mas não atendia todas as necessidades, então iniciou-se a busca por melhorias.

Durante o desenvolvimento de melhorias, a empresa Controtec foi procurada por um motel buscando automatização. Para o primeiro sistema de automação criado pelo engenheiro da empresa, foi realizado um estudo do fluxo de atividade e analisado o processo inteiro etapa por etapa. Com o sistema, buscou-se ter tudo em sincronia, onde várias atividades pudessem estar ocorrendo ao mesmo tempo e as atividades que seguissem em cascata ocorressem da forma como deveriam.

O protótipo do sistema foi apresentado e aceito, permitindo que fosse terminado então o projeto de hardware e paralelamente o projeto de software que foi desenvolvido por uma empresa terceirizada. Desde então, o sistema de automação da Controtec tem passado por melhorias, conforme novas técnicas vêm surgindo. Além disso, são feitas alterações para se adaptar às necessidades do cliente, conforme elas evoluem.

Atualmente o sistema de automação é composto por diversas partes físicas:

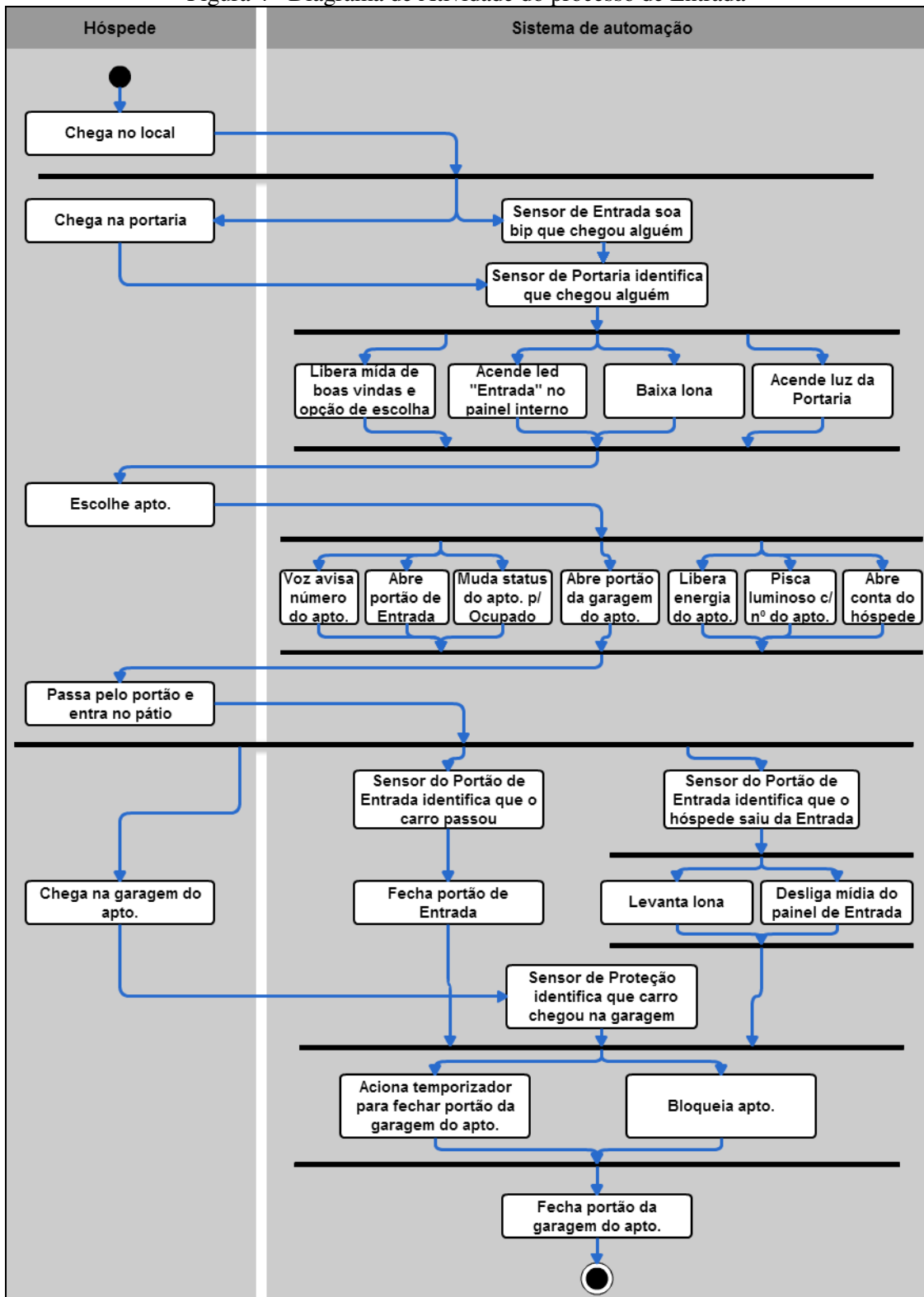
- a) sensores que identificam quando carros ou pessoas passam na frente;
- b) portões que são controlados pelo sistema;
- c) 3 computadores (na entrada, na recepção e na saída);
- d) telas na entrada e na saída que mostram opções de apartamento e apresentam mídias de voz;
- e) outras partes relacionadas ao funcionamento do sistema de automação como porta serial, *leds*, cabos e processador PIC.

Como interfaces principais do sistema tem-se a tela *Front End* (Figura 2), para o funcionário que gerencia as entradas e saídas do motel, e o painel de controle, conforme apresentado na Figura 3, que apresenta basicamente as mesmas funcionalidades que a tela principal do *Front End* porém na forma de hardware.

acesos nas respectivas situações, como porta aberta na suíte 38 ou portão da garagem aberto e infravermelho da garagem ativado na suíte 56. Além disso, os botões encontrados na parte esquerda do painel permitem a alteração de algumas funções de automático para manual, quando alguma dessas funções não está funcionando como deveria. Podem-se mencionar como exemplo os portões, a lona da entrada e o bip das alas do motel.

Quanto ao processo em que o sistema de automação está inserido, pode-se dividi-lo em 2 principais fluxos que são o de entrada e saída do motel. Nas figuras 3 e 4 tem-se os diagramas de atividade do processo de entrada e o de saída respectivamente.

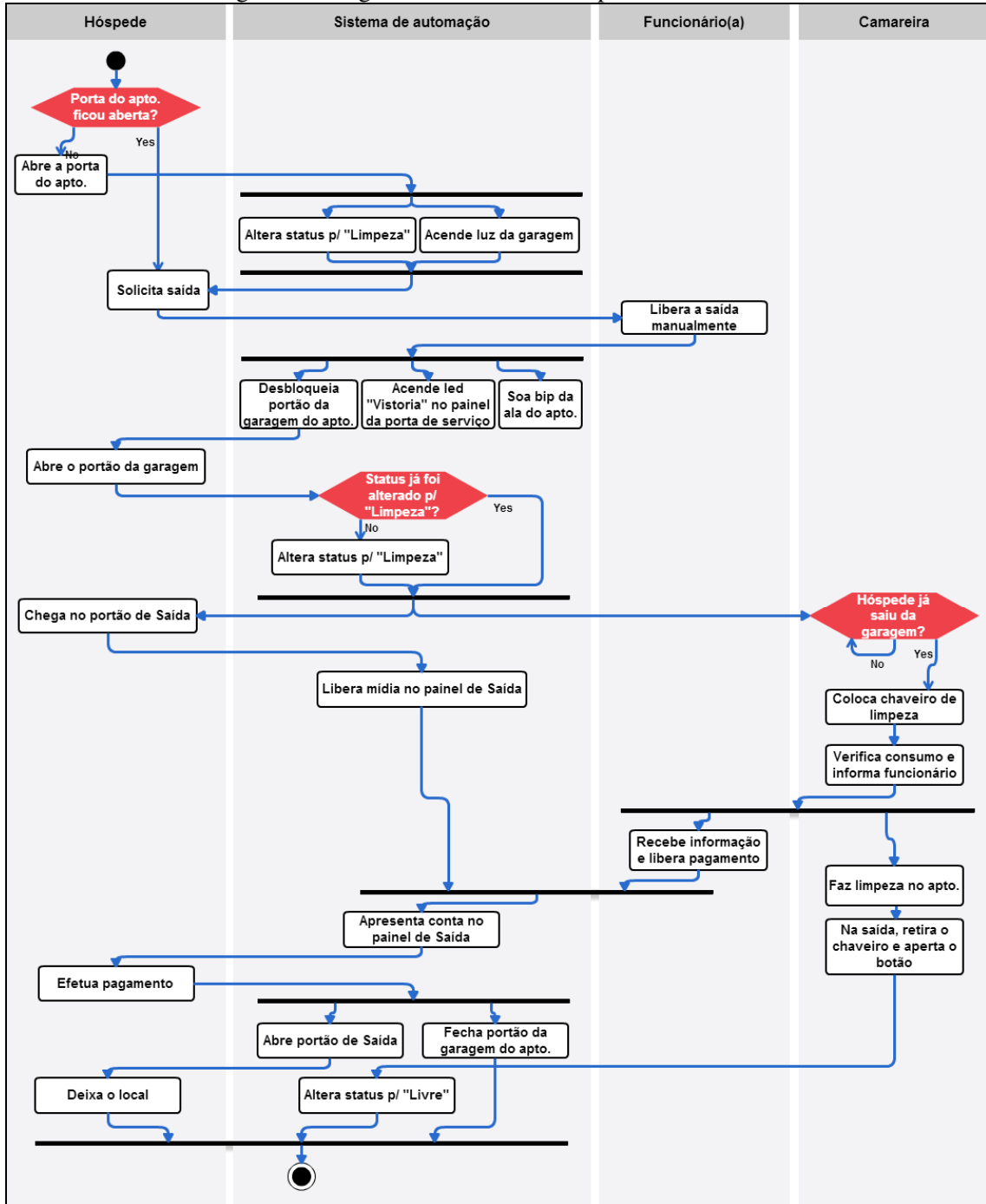
Figura 4 - Diagrama de Atividade do processo de Entrada



O processo de Entrada se inicia quando o hóspede chega no local, dirigindo-se à portaria. O sistema, através de sensores, identifica que chegou alguém e libera a mídia de voz que dá as boas vindas e apresenta as opções de escolha de apartamentos. Ao escolher o apartamento, o sistema abre então uma conta para o hóspede e permite a entrada do mesmo,

abrindo o portão de Entrada e o portão da garagem do apartamento. Nesse momento, a energia do apartamento também é liberada. Para controle do funcionário, o sistema altera o status do apartamento para Ocupado. Tanto no portão da Entrada quanto no portão da garagem, há sensores que identificam quando o carro passa para então o sistema fechar o portão.

Figura 5 - Diagrama de Atividade do processo de Saída



A partir do momento que o carro passa pelo portão da garagem, o apartamento fica então bloqueado, não permitindo a saída do hóspede. Para que ele consiga abrir o portão da garagem, precisa solicitar saída ao funcionário da recepção. O funcionário precisa liberar a

saída manualmente e deve ser feito para um apartamento de cada vez. Quando é liberado, o hóspede tem a possibilidade então de abrir o portão da garagem e se direcionar à saída. Paralelamente, as camareiras percebem que há um hóspede saindo através de um bip que soa nos corredores que é único para cada ala do motel. Além disso, acende-se um led de "Vistoria" no painel da porta de serviço do apartamento, para identificar qual é o apartamento. Assim que o hóspede sai, a camareira entra e verifica o consumo para informar ao funcionário da recepção. Enquanto isso, o hóspede chega na saída, o que libera a mídia de voz no painel de saída. Assim que o funcionário da recepção é informado do consumo, libera o pagamento para o hóspede. Assim que for pago, o portão da saída é aberto e só então o portão da garagem do apartamento é fechado. Por último, o status é alterado para "Livre" quando a camareira termina o serviço.

2.5 SISTEMA ATUAL

A empresa Controtec Indústria e Comércio Ltda. não possui guia ou documentação de solução para problemas básicos que possam ocorrer no sistema de automação de motéis. Hoje, quando alguma parte do sistema apresenta defeito, o técnico do local entra em contato via telefone com o engenheiro da empresa informando o ocorrido. O engenheiro orienta que sejam feitas algumas verificações para entender a causa do problema e em muitos casos já consegue passar uma solução.

O processo de manutenção por telefone ocorre da seguinte forma:

- a) os funcionários do motel identificam um problema que precisa ser corrigido com urgência;
- b) o técnico ou alguém com algum conhecimento básico sobre o funcionamento do sistema presente no local entra em contato com a Controtec buscando falar com o engenheiro;
- c) o engenheiro da Controtec escuta as informações providas pelo técnico e pede as verificações que sejam necessárias;
- d) o técnico do local faz as verificações e informa o engenheiro;
- e) assim que é identificado o problema, o engenheiro passa a orientação para o técnico sobre o que precisa ser feito para a correção. Em casos em que ele não consiga identificar o problema, ou seja, precise de um atendimento presencial da Controtec, o engenheiro passa uma solução paliativa para que possam evitar a indisponibilidade da parte que se encontra com defeito.

Este processo de manutenção pode demorar cerca de 3 minutos a 1 hora, dependendo da complexidade do problema e das verificações que se fazem necessárias. O engenheiro procura ser ágil no atendimento, conseguindo solucionar alguns problemas em poucos minutos, mas há casos em que a investigação é mais complexa e o engenheiro divide a consulta em várias ligações, orientando um teste de cada vez. Nesses procedimentos mais longos, o engenheiro orienta uma análise e continua fazendo sua atividade até que o técnico retorne a ligação com o resultado da análise. Em muitos casos, isso se repete em até 4 ou 5 ligações, até que se ache a razão do defeito.

Durante essa assistência, o engenheiro fica grande parte do tempo impossibilitado de realizar outras atividades, podendo causar atrasos em outros projetos e não tendo nenhum retorno financeiro com isso. Em raras situações também pode ocorrer de o engenheiro estar indisponível para contato, causando insatisfação por parte do cliente quanto à empresa Controtec.

2.6 TRABALHOS CORRELATOS

Podem-se citar como trabalhos correlatos as monografias realizadas pelos alunos Daniel Jonas Heinrich e Paulo Augusto Argenton para conclusão do curso de Ciências da Computação na Universidade Regional de Blumenau e pelo aluno Robson Carneiro Correia da Silva para conclusão do curso de Engenharia da Computação na Escola Politécnica de Pernambuco.

2.5.1 Protótipo de um sistema especialista para atendimento a ocorrências em transformadores de potência

O objetivo do trabalho de Argenton (1996) consistia em desenvolver um protótipo de sistema especialista que auxiliasse no atendimento a ocorrências em transformadores de potência, e dessa forma minimizasse a indisponibilidade do equipamento, gerando agilização e maior fator de certeza na tomada de decisão acerca da ocorrência. Pretendia-se também criar uma padronização no processo de atendimento à ocorrência em transformadores, onde anteriormente existiam documentações complexas e de difícil entendimento e que muitas vezes o operador da subestação necessitava de auxílio (ARGENTON, 1996).

O protótipo criado possibilita a tomada de decisões baseadas no conhecimento do especialista da área de operação e manutenção do sistema de energia elétrica. O resultado esperado foi atingido, sendo que o protótipo inicial satisfazia plenamente as necessidades do usuário.

O conhecimento do especialista foi armazenado em um banco de conhecimentos através de fatos e regras. Para o desenvolvimento do trabalho, ele utilizou duas ferramentas. A primeira foi a *shell* SPIRIT, que tem seus fundamentos baseados na lógica e inferência probabilística. Essa ferramenta não foi utilizada para a construção do produto final, pois não atendeu a todas as necessidades, tendo como principal defeito uma interação confusa com o usuário. Com a exclusão dessa *shell*, a implementação continuou com a linguagem PROLOG.

2.5.2 Sistema de apoio para o diagnóstico de defeitos em equipamentos eletrônicos aplicado a oficinas eletrônicas utilizando raciocínio baseado em casos

O trabalho de Heinrich (2001) tinha o objetivo de desenvolver um sistema de apoio ao técnico de eletrônica que pudesse facilitar e agilizar o serviço prestado, apresentando serviços com maior qualidade.

O sistema criado permite solucionar os problemas dos aparelhos eletrônicos tomando como base as soluções fornecidas através do uso do raciocínio baseado em casos, mais especificamente as regras de similaridade.

Para a consulta de diagnósticos, foi criada uma tela para a consulta de casos, como mostra a Figura 6. Nesta tela o técnico seleciona a marca e o modelo do aparelho e informa as características do defeito, ao clicar no botão RBC o sistema fará uma busca na base de casos procurando os casos mais similares. Ao encontrar casos com características similares estes terão sua solução demonstrada na tela com seu respectivo percentual de similaridade (HEINRICH, 2001).

Figura 6 - Tela de Consulta de Casos

Consulta Casos

Marca: 001-PHILIPS

Modelo: 14PT110A -TELEVISOR 14"

Característica 1: 00001-LINHAS

Característica 2: 00009-DIAGONAIS

Característica 3: 00010-EM SINAL "CABLE"

RBC

%	Solução
100,00%	Utilizar sintonizador com melhor rejeição aos espúrios de adjacentes 4806 21
40,00%	Verificar R3240.
40,00%	Verificar R3407. Utilizar dois resistores conforme informação de serviço núme

Desfazer Sair OK Cadastrar novo caso

Utilizar sintonizador com melhor rejeição aos espúrios de adjacentes 4806 210 47037.

O sistema construído permite que além de visualizar os casos existentes, o usuário também possa cadastrar novos casos. Na Figura 7, tem-se a tela de cadastro de novos casos onde o técnico efetua o cadastro dos mesmos utilizando-se dos dados cadastrados anteriormente. Nesta mesma tela também são atribuídos os pesos das características que compõem este caso. O valor dos pesos é definido pelo técnico conforme o grau de importância de cada característica (HEINRICH, 2001).

Figura 7 - Tela de Cadastro de Novos Casos

Cadastro de Casos

Caso: 000001

Marca: PHILIPS

Modelo: 14PT110A

1ª Característica: NÃO FUNCIONA Peso: 04

2ª Característica: LED FICA ACESO DIRETO Peso: 03

3ª Característica: (+10V, +5V)TENSÕES BAIXAS Peso: 03

Solução: Verificar IC7600 (Micro) pino 42 em curto com o terra.

Entre os benefícios percebidos por Heinrich (2001) na conclusão do trabalho, está a troca de informações entre oficinas eletrônicas, que possibilita o aumento do conhecimento armazenado na base de dados de cada uma das oficinas. Além disso, viu-se a oportunidade de utilizar o sistema para treinar novos técnicos utilizando como referência os casos armazenados pelo sistema.

As ferramentas utilizadas para a realização do trabalho foram a análise essencial, a ferramenta CASE Power Designer 6.1, o banco de dados Paradox e o ambiente de programação Delphi.

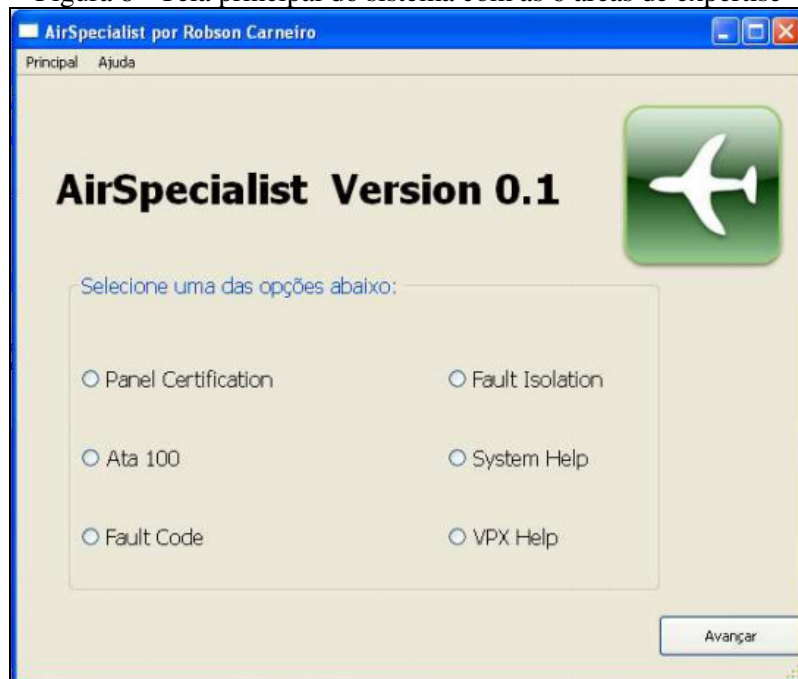
2.5.3 Sistema Especialista de apoio à decisão para o diagnóstico de falhas em aeronaves

O trabalho de Silva (2010) foi concebido para prestar apoio à equipe de manutenção no processo de tomada de decisões no trabalho de manutenção, utilizando um banco de regras contendo conhecimentos especializados. Para exemplificar o conceito, foi implementado um dos subsistemas: o Aviso de Estol. Este módulo evita uma aeronave chegar a uma condição de estol ou perda de sustentação aerodinâmica (SILVA, 2010).

A linguagem de programação utilizada para o desenvolvimento foi Python, e devido à flexibilidade da linguagem escolhida, foi implementada uma das técnicas de mineração de dados, mais especificamente a técnica de Árvores de Decisão.

Na Figura 8 tem-se a tela inicial onde o sistema apresenta 6 opções que devem ser seguidas sequencialmente. Na opção de Panel/Certification, é possível obter informações sobre a aeronave através de número de série, modelo e motor. De posse dessas informações, na opção Ata 100, o usuário deve escolher o capítulo desejado, onde cada capítulo é um dos sistemas que compõe a aeronave. Selecionado o capítulo, é apresentada uma breve descrição e solicitado então a seção, que seria um subsistema.

Figura 8 - Tela principal do sistema com as 6 áreas de expertise



Possuindo os dados da aeronave, do sistema e do subsistema, é iniciada a busca pelo código de falha, que é feita através de uma série de perguntas estruturadas até que o sistema encontre a solução. Esse resultado é apresentado através de um código para reparo relativo ao manual de trabalho da aeronave onde se tem o procedimento necessário ao reparo da mesma.

Para avaliar a eficiência do protótipo criado, foi realizada uma avaliação onde compararam o número de regras utilizados até chegar à conclusão seguindo duas abordagens:

- a) pelos especialistas na área de sistemas de aviso de estol, sem o uso do sistema e apenas com uso de manuais;
- b) pelo sistema especialista guiado pelos usuários que tiveram um breve treinamento do sistema.

Na conclusão dessa avaliação, viu-se que o sistema especialista obteve um melhor resultado, pois foram utilizadas uma quantidade de regras menor para se chegar à solução, em comparação com um especialista humano. Isto mostra que o sistema especialista é eficiente na eliminação de redundância de regras, permitindo ao usuário chegar rapidamente à solução (SILVA, 2010).

2.5.4 Comparação entre as características dos trabalhos relacionados

A partir das informações obtidas com os trabalhos descritos, percebe-se que os sistemas de Argenton (1996) e Heinrich (2001) foram utilizados para auxiliar no diagnóstico de defeitos na área de engenharia elétrica, assim como o protótipo desenvolvido. Em ambos

os sistemas, tem-se o objetivo de agilizar o serviço, aumentar a precisão do atendimento do técnico sem a necessidade de auxílio de um especialista ou de documentações complexas.

No trabalho de Argenton (1996), o conhecimento foi provido pelo especialista da área de operação e manutenção do sistema de energia elétrica, o que é similar ao que foi feito no desenvolvimento desse protótipo, onde o especialista responsável pela criação, instalação e manutenção do sistema de automação compartilhou as informações para construir a base de conhecimento. Nas telas do protótipo desse trabalho foi seguida a mesma lógica do protótipo de Silva (2010), onde se escolhe um capítulo/sistema e dentro deste, é possível selecionar o subsistema que se encontra com defeito. Assim como no trabalho de Heinrich (2001), neste protótipo também há a opção de cadastrar mais casos, permitindo manter o sistema atualizado com as alterações que possam ser feitas no sistema de automação.

3 DESENVOLVIMENTO DO PROTÓTIPO

Neste capítulo estão descritos os aspectos técnicos utilizados no desenvolvimento do protótipo, bem como o levantamento de informações: requisitos funcionais e não funcionais. Além dos diagramas utilizados no desenvolvimento: diagrama de casos de uso e modelo de entidade e relacionamento, estão descritas também as técnicas e ferramentas utilizadas, as etapas da implementação e os resultados obtidos.

3.1 LEVANTAMENTO DE INFORMAÇÕES

Desenvolvido para ser utilizado pelos motéis que possuem sistema de automação instalado pela empresa Controtec, o protótipo de sistema especialista implementado visa aumentar a precisão da identificação de defeitos e tornar mais ágil o processo de correção. De forma indireta, também pretende eliminar o atendimento básico feito pelo engenheiro da empresa, deixando-o livre para realizar outras atividades.

O protótipo possui uma base de conhecimento com alguns dos defeitos já conhecidos pelo engenheiro da empresa que podem ocorrer no sistema de automação. A partir dessa base de conhecimento, o protótipo consegue emitir o diagnóstico de falhas, apresentando as possíveis causas e indicando as providências que devem ser tomadas para a correção do problema.

O processo se inicia de forma que quando identificado um defeito no sistema de automação, o técnico do local ou outra pessoa com conhecimentos básicos sobre o funcionamento do sistema faz a consulta no protótipo de sistema especialista, substituindo o atual processo de ligar para o engenheiro da empresa.

Na consulta no protótipo, o técnico seleciona qual módulo do sistema de automação encontra-se com defeito, para que então o protótipo possa gerar as perguntas relacionadas àquele módulo do sistema de automação. O técnico então realiza as verificações solicitadas e responde quanto ao resultado, permitindo que o protótipo faça mais perguntas ou encontre a causa raiz do defeito. Quando for encontrada, é informado ao usuário sobre as possíveis causas e apresentados os passos a serem seguidos para corrigir o problema.

Em alguns casos de exceção em que a solução não for eficiente ou que o protótipo de sistema especialista não encontre a causa do defeito e a respectiva solução, é disponibilizado um mecanismo que possa recomendar soluções de contingência e é orientado o contato com a empresa Controtec para agendar uma visita técnica ao local. O objetivo deste mecanismo é diminuir a indisponibilidade da parte do sistema de automação até que a visita seja feita.

3.2 ESPECIFICAÇÃO

A seguir é apresentada a especificação do protótipo, contendo os requisitos funcionais e requisitos não funcionais, além dos diagramas de casos de uso e o Modelo de Entidade Relacionamento (MER). Para criar o diagrama de caso de uso foi utilizada a ferramenta Enterprise Architect (EA), já para a criação do MER utilizou-se a ferramenta DBDesigner.

3.2.1 Requisitos Funcionais e Não Funcionais

O Quadro 1 apresenta os requisitos funcionais previstos para o protótipo e sua rastreabilidade, ou seja, vinculação com o caso de uso associado.

Quadro 1 - Requisitos funcionais

Requisitos Funcionais	Caso de Uso
RF01: O sistema deverá permitir ao administrador manter usuários.	UC01
RF02: O sistema deverá permitir ao usuário efetuar login.	UC02
RF03: O sistema deverá permitir ao usuário fazer uma consulta	UC03
RF04: O sistema deverá permitir ao usuário visualizar medidas de contingência para algumas situações.	UC04
RF05: O sistema deverá permitir ao administrador manter as perguntas do sistema.	UC05
RF06: O sistema deverá permitir ao administrador manter soluções.	UC06
RF07: O sistema deverá permitir ao administrador manter contingências.	UC07

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

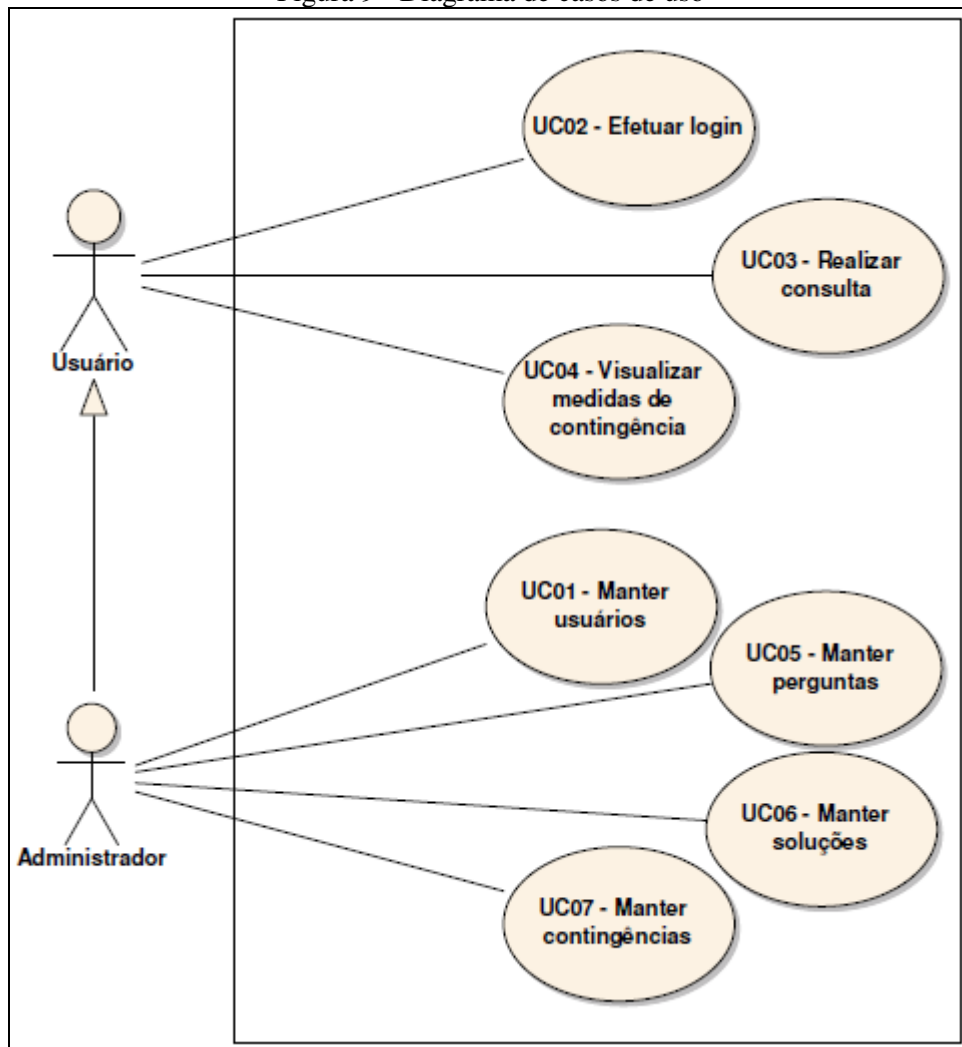
Quadro 2 - Requisitos não funcionais

Requisitos Não Funcionais
RNF01: O sistema deverá ser desenvolvido em PHP para WEB
RNF02: O sistema deverá utilizar banco de dados MySQL.
RNF03: O sistema deverá ser acessível via navegador Microsoft Internet Explorer versão 9 ou superior, Mozilla Firefox versão 30 ou superior e Google Chrome versão 35 ou superior

3.2.2 Casos de Uso

Esta seção apresenta, na Figura 9, o Diagrama de Casos de Uso do protótipo desenvolvido, onde foram identificados dois atores: usuário e administrador. A descrição expandida dos principais casos de uso é apresentada no Apêndice A.

Figura 9 - Diagrama de casos de uso



O caso de uso UC01 Manter usuários permite ao administrador criar contas para novos usuários e excluí-las quando necessário. No caso de uso UC02 Efetuar login, o usuário consegue conectar-se ao sistema através da identificação por nome de usuário e senha criados pelo administrador.

O caso de uso UC03 Realizar consulta permite ao usuário escolher qual parte do sistema de automação está com defeito para que o sistema apresente verificações que devem ser feitas. O usuário precisa responder as verificações para que o sistema possa então identificar a razão do problema e apresentar soluções para a correção do defeito. Quando não houver solução imediata e existir alguma contingência cadastrada para o problema, o sistema deve apresentar a contingência cadastrada para o problema que é apresentada no UC04 Visualizar medidas de contingência.

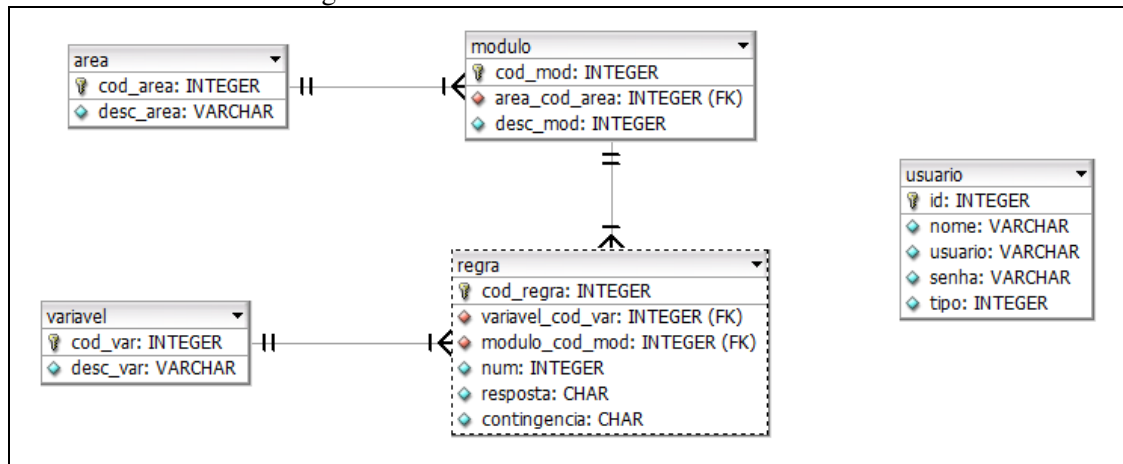
Os casos de uso UC05 Manter perguntas, UC06 Manter soluções e UC07 Manter contingências permitem ao administrador cadastrar novos registros de perguntas, soluções

ou contingências. Além disso, terá possibilidade de editar ou excluir quando houver necessidade.

3.2.3 Modelo Entidade-Relacionamento

A Figura 10 mostra o Modelo Entidade-Relacionamento (MER) do sistema. O Dicionário de Dados está apresentado no Apêndice B.

Figura 10 - Modelo Entidade-Relacionamento



A seguir é apresentada uma breve descrição das entidades criadas para o desenvolvimento do sistema:

- a) *area*: tabela que armazena informações da área do sistema de automação;
- b) *modulo*: tabela que armazena os módulos do sistema de automação. Possui referência de uma das áreas;
- c) *regra*: armazena as informações da regra. Possui referência de um dos módulos e de uma ou mais variáveis;
- d) *usuario*: tabela que armazena informações dos usuários que fazem uso do sistema. Essa tabela é referenciada para diferenciar administrador de usuário e também para login no sistema;
- e) *variavel*: armazena as informações das perguntas cadastradas para a regra.

3.3 IMPLEMENTAÇÃO

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

3.3.1 Técnicas e ferramentas utilizadas

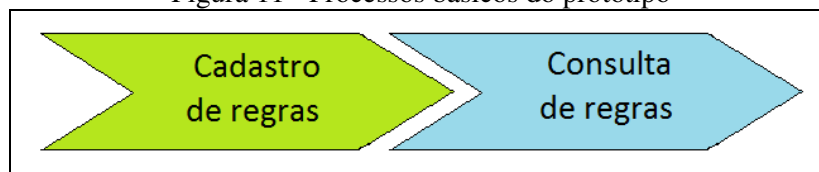
A modelagem de dados e criação dos objetos no banco de dados MySQL foi realizado utilizando o software phpMyAdmin.

O protótipo foi desenvolvido utilizando o software Notepad ++ que é uma ferramenta gratuita para edição de texto. A ferramenta Notepad ++ interpreta os padrões da linguagem PHP, HTML, JavaScript e CSS, o que facilita a edição do código.

3.3.2 Etapas da implementação

O processo para a consulta de regras sobre o sistema de automação compreende, conforme mostra a Figura 11, a aquisição dos dados de entrada, através da tela de cadastro de regras e consulta dessas regras, realizada através de um vetor buscado na base de dados de acordo com o módulo escolhido. Nas seções 3.3.2.1 e 3.3.2.2 são descritas as etapas e cadastro de regras e consulta de regras respectivamente.

Figura 11 - Processos básicos do protótipo



3.3.2.1 Cadastro de regras

Para cadastrar uma nova regra, o administrador irá primeiramente escolher o módulo do sistema de automação, que está disponível em um menu em cascata na tela de cadastro de regra, conforme exhibe a Figura 12.

Figura 12 - Menu de módulos em cascata

A imagem mostra uma interface de usuário para o cadastro de regras. No topo, há um menu em cascata com as opções: 'Entrada / Saída', 'Garagem' e 'Recepção/Interno'. O menu 'Entrada / Saída' está aberto, mostrando sub-opções: 'Luz', 'Bip', 'Painel' e 'Portão'. Abaixo do menu, há um campo de texto com o texto 'no menu acima.' e um campo 'Num' com o valor 'Esc'. À direita, há campos para 'Filhos?' e 'É resposta?' com botões 'SIM', 'NÃO' e 'Excluir'. No rodapé, há um botão verde 'Cadastrar'.

Conforme mostra a Figura 13, ao escolher o módulo, o sistema irá preencher o campo Num da tela, que será correspondente ao número da nova regra associada ao módulo escolhido.

O número preenchido é baseado nas regras já existentes para o módulo, passando então o próximo número sequencial de regra.

Figura 13 - Tela de cadastro de regra após escolher um módulo

CADASTRO DE REGRA

Entrada / Saída Garagem Recepção/Interno

Área: Recepção/Interno Módulo: visor do corredor

Num	Variável	Filhos?	É solução?
1		<input type="checkbox"/> SIM <input type="checkbox"/> NÃO	<input type="checkbox"/>

Tem contingência?

O campo *Num* será sempre preenchido automaticamente, então o único campo obrigatório é a *Variável*, onde o administrador irá preencher as perguntas, respostas e medidas de contingência pertencentes à regra. Para incluir uma sequência na regra, o administrador pode clicar nos botões *SIM* e *NÃO*, escolhendo de acordo com a resposta para a pergunta atual.

Quando for definida uma solução para a regra criada, é necessário marcar o *checkbox* "É resposta?" para indicar o fim de uma sequência. Uma regra pode ter mais de uma sequência e, dessa forma, mais de uma solução.

Para cada regra criada, o sistema também permite cadastrar uma contingência, caso a solução apresentada não tenha o efeito esperado ou seja necessário aguardar uma visita do engenheiro. A Figura 14 mostra um exemplo com mais de uma solução e uma contingência cadastrada para o problema.

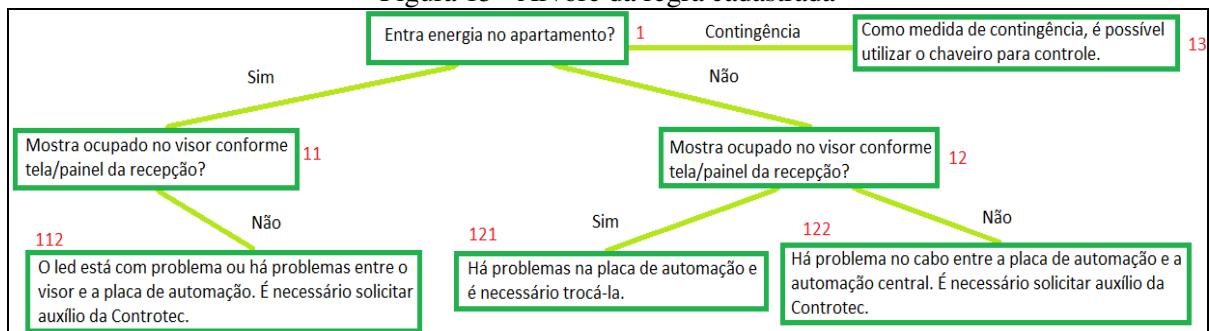
Figura 14 - Tela de cadastro com respostas e contingência cadastradas

Área: Recepção/Interno		Módulo: visor do corredor			
Num	Variável	Filhos?	É solução?		
1	Entra energia no apartamento?	SIM NÃO	<input type="checkbox"/>	<input type="checkbox"/>	Excluir
11	Mostra ocupado no visor conforme tela/;	SIM NÃO	<input type="checkbox"/>	<input type="checkbox"/>	Excluir
112	O led está com problema ou há problem.		<input checked="" type="checkbox"/>	<input type="checkbox"/>	Excluir
12	Mostra ocupado no visor conforme tela/;	SIM NÃO	<input type="checkbox"/>	<input type="checkbox"/>	Excluir
121	Há problemas na placa de automação e		<input checked="" type="checkbox"/>	<input type="checkbox"/>	Excluir
122	Há problema no cabo entre a placa de a		<input checked="" type="checkbox"/>	<input type="checkbox"/>	Excluir
<input checked="" type="checkbox"/> Tem contingência?					
13	Como medida de contingência, é possív		<input type="checkbox"/>	<input type="checkbox"/>	Excluir

Cadastrar

Na Figura 15 é exibida a regra preenchida na tela conforme Figura 14 em formato de árvore, onde cada variável da árvore tem seu Num, sendo que as variáveis que são sequência sim possuem “1” no final e as que são sequência não possuem “2”. No caso de contingência, para ser identificado de forma diferente das demais variáveis, o Num possui “3” no final do Num da primeira variável.

Figura 15 - Árvore da regra cadastrada



Caso o administrador deseje excluir uma linha, todas as variáveis que dependem dela, ou seja, são filhos SIM ou NÃO dessa variável, também serão excluídas. Para realizar o cadastro da regra no banco, o sistema utiliza o trecho de código apresentado nos Quadro 3 e Quadro 4.

Quadro 3 - Trecho do código para receber informações cadastradas

```

01 if (isset($_POST["cadastrar"])) {
02     $vetor = array(array());
03     $temp = '';
04     foreach ($_POST as $key => $value) {
05         if ($key != 'cadastrar' && $key != 'mod2'){
06             if (strpos($key,'num') !== false){
07                 $temp = $value;
08                 $vetor[$temp][0] = null;
09                 $vetor[$temp][1] = 'N';
10                 if (substr($value, -1) == 3 && strlen($value) != 1){
11                     $vetor[$temp][2] = 'S';
12                 }else{
13                     $vetor[$temp][2] = 'N';
14                 }
15             }else if (strpos($key,'var') !== false){
16                 $sql2 = mysql_query("select cod_var from variavel where
17                                     desc_var like '". $value ."");
18                 if (mysql_num_rows($sql2)){
19                     $var = mysql_fetch_array($sql2);
20                     $vetor[$temp][0] = $var['cod_var'];
21                 }else{
22                     $conn->query("INSERT INTO variavel VALUES ('',
23                                     '". $value ."");
24                     $cod = mysqli_insert_id($conn);
25                     $vetor[$temp][0] = $cod;
26                 }
27             }else if (strpos($key,'resposta') !== false){
28                 $vetor[$temp][1] = 'S';
29             }
30         }
31     }
32 }

```

No Quadro 3, das linhas 4 a 28, é mostrado o trecho em que o sistema lê as variáveis preenchidas pelo administrador com seus devidos Nums e as insere num vetor, contendo o Num como índice, a variável, indicador de resposta e o de contingência como valores. Nos casos de resposta e contingência, o sistema insere 'N' indicando que não é resposta ou não é contingência como padrão e altera para 'S' quando encontrar alguma variável com o *checkbox* marcado.

Quadro 4 - Trecho do código para cadastro da regra no banco

```

01 $mod = $_POST['mod2'];
02 reset($vetor);
03 next($vetor);
04 $first_key = key($vetor);
05 foreach ($vetor as $key => $value){
06     if ($key != 0){
07         $conn->query("INSERT INTO regra VALUES ('', '". $first_key ."',
08                                     '". $key ."', '". $value[0] ."', '". $mod ."', '". $value[1] ."',
09                                     '". $value[2] ."");
10     }
11 }

```

O Quadro 4 mostra nas linhas 5 a 9 o trecho de código que lê o vetor que foi criado e cria um registro na tabela regra para cada variável criada.

3.3.2.2 Consulta de regras

Ao encontrar um defeito no sistema de automação, o usuário irá escolher o módulo que apresenta defeito na tela de consulta e o sistema irá apresentar perguntas pertencentes às regras associadas ao módulo escolhido. Na

Figura 16, é exibida a primeira pergunta da primeira regra cadastrada para o módulo consultado.

Figura 16 - Tela de consulta após escolher um módulo

Para exibir a regra de acordo com a escolha do módulo, o sistema utiliza o trecho de código no Quadro 5, onde na linha 5 é passado como parâmetro o código do módulo escolhido para uma consulta no banco que traz as regras cadastradas.

Quadro 5 - Trecho de código que faz a consulta de regra no banco

```

01 <?php
02     if($_SERVER['REQUEST_METHOD'] == 'POST'){
03         if($_POST['mod']){
04             $mod = $_POST['mod'];
05             $sql = mysql_query("select * from regra where cod_mod = $mod
                                order by cod_regra, num");
06         }
07     }else{
08         $sql = '';
09     }
10
11     $vetor = array(array());
12
13     if ($sql != ''){
14         while ($resultado = mysql_fetch_array($sql)) {
15             //pega descrição da variavel
16             $cod_var = $resultado ['cod_var'];
17             $var = mysql_fetch_array(mysql_query("select desc_var from
                                                    variavel where cod_var = $cod_var"));
18             $vetor[$resultado ['num']][0] = $var ['desc_var'];
19             $vetor[$resultado ['num']][1] = $resultado ['resposta'];
20             $vetor[$resultado ['num']][2] = $resultado ['contingencia'];
21         }
22     }
23 ?>

```

Conforme linhas 14 a 21 do Quadro 5, o resultado da busca no banco é passado então para um vetor, armazenado da mesma forma que o vetor do cadastro de regras, onde Num é o índice, a variável e os indicadores de resposta e contingência são os valores.

Para transformar os valores do vetor em perguntas, respostas ou contingências a serem apresentadas na tela de consulta, o sistema lê o vetor e cria uma <div> para cada variável, conforme trecho de código das linhas 2 a 20 apresentado no Quadro 6.

Quadro 6 - Trecho de código com informações da tela de consulta

```

01 <?php
02     foreach ($vetor as $key => $value) {
03         if ($key != 0){
04             if ($value[1] == 'S'){
05                 ?>
06                 <div id="resposta<?php echo $key; ?>"><?php echo
07                     $value[0] ?></div>
08                 <?php
09             }
10             if($value[2] == 'S'){
11                 ?>
12                 <div id="contingencia<?php echo $key; ?>"><?php echo
13                     $value[0] ?></div>
14                 <?php
15             }
16             if ($value[1] == 'N' && $value[2] == 'N'){
17                 ?>
18                 <div id="pergunta<?php echo $key; ?>"><?php echo
19                     $value[0] ?></div>
20                 <?php
21             }
22         }
23     }
24     ?>
25     <div id="naotemsequencia">Não temos registro para o prolema que você
26         está procurando na nossa base.
27     Favor entrar em contato com a Controtec para solicitar auxílio.</div>
28     <input type="hidden" id="perguntaAtual" value=""><br>

```

Conforme o usuário responde sim ou não para as perguntas, o sistema exibe a próxima pergunta ou resposta de acordo com a sequência cadastrada para a regra. Para essa funcionalidade, é utilizada a função `proximaPergunta()` exibida nos Quadro 7 e Quadro 8.

Quadro 7- Função `proximaPergunta()` em consulta

```

01 function proximaPergunta(){
02     atual = document.getElementById('perguntaAtual').value;
03
04     if (document.getElementById("sim").checked){
05         document.getElementById('voltar').style.display = "block";
06         escondeTudo();
07         proximo = atual+"1";
08         document.getElementById('perguntaAtual').value = proximo;
09         if(document.getElementById('pergunta'+proximo) !== null){
10
11             document.getElementById('pergunta'+document.
12                 getElementById('perguntaAtual').value).style.display =
13                 "block";
14         }else if(document.getElementById('resposta'+proximo)
15                 !== null){
16             document.getElementById('resposta'+document.
17                 getElementById('perguntaAtual').value).style.display
18                 = "block";
19
20             ehUltimo();
21             contg = proximo+"3";
22             if(document.getElementById('contingencia'+contg)
23                 !== null){
24
25                 document.getElementById('exibircontingencia').
26                     name = contg;
27                 document.getElementById('exibircontingencia').
28                     style.display = "block";
29             }
30         }else{
31             document.getElementById('naotemsequencia').
32                 style.display = "block";
33         }
34     }
35     ehUltimo();

```

O *input* com *id* “perguntaAtual” mostrado na linha 23 do Quadro 6 sempre recebe o valor da pergunta atual que está sendo apresentada na tela de consulta. Nas linhas 7 do Quadro 7 e 4 do Quadro 8, a variável `proximo` recebe o valor que será passado para o *input* “perguntaAtual” para trazer a próxima variável.

Quadro 8 - Continuação da função proximaPergunta() em consulta

```

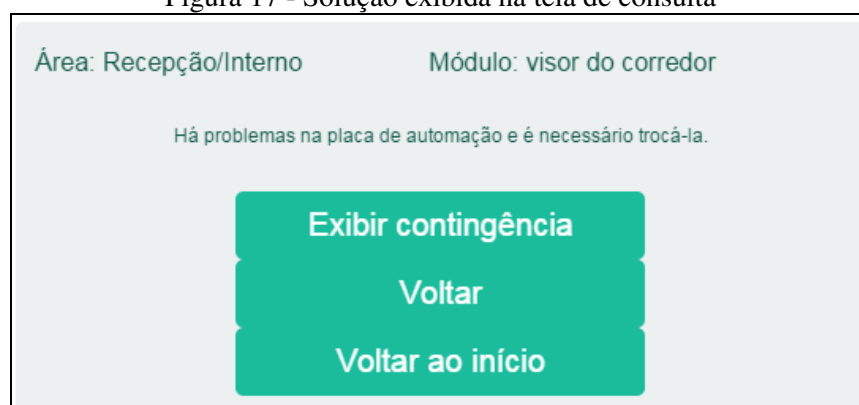
01 }else if(document.getElementById("nao").checked){
02     document.getElementById('voltar').style.display = "block";
03     escondeTudo();
04     proximo = atual+"2";
05     document.getElementById('perguntaAtual').value = proximo;
06     if(document.getElementById('pergunta'+proximo) != null){
07         document.getElementById('pergunta'+document.
08             getElementById('perguntaAtual').value).style.display
09             = "block";
10     }else if(document.getElementById('resposta'+proximo) != null){
11         document.getElementById('resposta'+document.
12             getElementById('perguntaAtual').value).style.display =
13             "block";
14     }
15     ehUltimo();
16     contg = proximo+"3";
17     if(document.getElementById('contingencia'+contg) != null){
18         document.getElementById('exibircontingencia').name = contg;
19         document.getElementById('exibircontingencia').style.display
20             = "block";
21     }
22 }else{
23     document.getElementById('naotemsequencia').style.display
24         = "block";
25     ehUltimo();
26 }
27 }else{
28     alert("Selecione uma das opções: Sim / Não");
29 }
30 document.getElementById("sim").checked = false;
31 document.getElementById("nao").checked = false;

```

As variáveis que são sequência sim possuem “1” no final do Num e as que são sequência não possuem “2”, então o sistema acrescenta um desses valores para o Num da pergunta atual para buscar pela variável que corresponda à “perguntaAtual” nas <div>s criadas no trecho de código do Quadro 6.

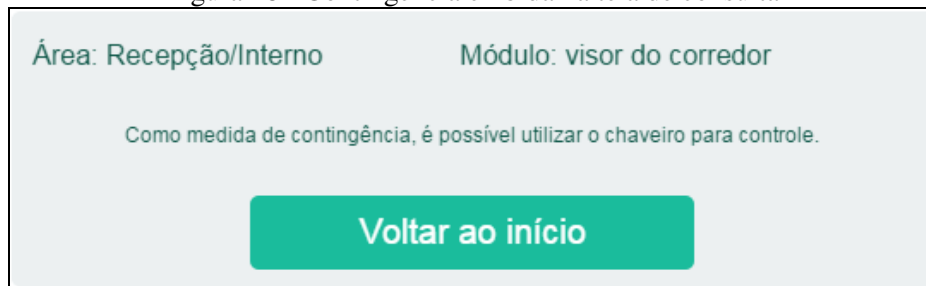
Ao atingir uma solução para o problema, identificada como resposta na linha 7 do Quadro 8, o sistema a apresenta, conforme mostra a Figura 17.

Figura 17 - Solução exibida na tela de consulta



Quando houver contingência cadastrada para a regra, o sistema apresenta o botão “Exibir contingência” conforme mostrado na Figura 17. Ao clicar no botão, a contingência é exibida conforme na Figura 18.

Figura 18 - Contingência exibida na tela de consulta



3.4 RESULTADOS E DISCUSSÕES

Nesta seção são apresentados os resultados de usabilidade do protótipo feitos com o administrador e com um usuário cliente.

3.4.1 Usabilidade do protótipo

Os testes de usabilidade foram realizados com o engenheiro responsável pela instalação e manutenção do sistema de automação e com um usuário cliente que irá utilizar o protótipo no local para avaliar a eficiência e facilidade de uso da aplicação.

3.4.1.1 Metodologia

Os testes foram realizados no mês de dezembro por meio de testes individuais. Para o engenheiro da empresa, foi disponibilizada uma lista de tarefas de administrador a serem executadas e um questionário de usabilidade com questões referente ao papel administrador da aplicação. Para o usuário cliente, foi disponibilizado uma lista de tarefas de usuário cliente e um questionário de usabilidade referente à usabilidade do papel cliente da aplicação. Os questionários disponibilizados aos usuários estão disponíveis no Apêndice C.

3.4.1.2 Teste realizado com o engenheiro / administrador

Primeiramente o engenheiro foi orientado sobre o objetivo do teste, da aplicação e de suas funcionalidades. A lista de tarefas foi composta de oito tarefas que teve por objetivo contemplar todos objetivos e funcionalidades da aplicação. Ao finalizar cada tarefa, foi solicitado que ele informasse se esta foi executada, caso tenha sido identificado algum problema ou se havia alguma observação a ser feita.

O questionário de usabilidade do administrador foi composto por dez perguntas, sendo nove fechadas e uma aberta. Para sete das questões fechadas foi disponibilizada a opção de informar uma observação. As perguntas tiveram por objetivo obter as impressões do engenheiro sobre a usabilidade em geral do protótipo e as funcionalidades disponibilizadas ao papel administrador. Além disso, buscou-se avaliar a opinião dele quanto à utilidade do protótipo no atendimento de problemas. Os resultados referentes à lista de tarefas desempenhadas e a usabilidade do engenheiro são respectivamente apresentados nas seções 3.4.1.2.1 e 3.4.1.2.2.

3.4.1.2.1 Análise dos resultados da lista de tarefas do administrador

No Quadro 9 são apresentados os resultados quanto às questões envolvendo os processos desempenhados pelo engenheiro na aplicação.

Quadro 9 - Respostas quanto à lista de tarefas

Tarefas	Executou?
Criar nova regra	Sim
Visualizar regra criada em regras cadastradas	Sim
Alterar regra existente	Sim
Excluir regra	Sim
Criar novo usuário	Sim
Consultar regra criada em consulta	Sim
Seguir sequência da regra e validar que está correto	Sim
Consultar outra regra e validar que solução é útil	Sim

Considerando os resultados das questões da lista de tarefas exibidas no Quadro 9, percebeu-se que todas as tarefas foram desempenhadas, o que indica o bom funcionamento no uso da aplicação. Uma observação foi feita para a tarefa de criar regras, onde o engenheiro mencionou que seria ideal aumentar o campo da variável. Nas tarefas de validar a sequência da regra e validar que a solução é útil também foi verificado a possibilidade de aprofundar a base de dados, acrescentando mais problemas conhecidos e mais verificações a serem feitas, assim como outras possíveis soluções. Durante a utilização do protótipo, o engenheiro considerou ele de fácil utilização e intuitiva na tela de consulta, porém teve dificuldade de utilização na tela de cadastro de regras, devido à adaptação para cadastrar as perguntas na sequência necessária. Após o cadastro de algumas perguntas e respostas, começou a familiarizar-se com a interface e não apresentou dificuldades para realizar a alteração ou exclusão de regras. No cadastro de usuário, não teve nenhuma dificuldade.

3.4.1.2.2 Análise dos resultados de usabilidade do administrador

Após análise dos resultados da lista de tarefas do administrador, foram analisados os resultados obtidos do questionário de usabilidade. Os resultados constam no Quadro 10.

Quadro 10 - Respostas do engenheiro em relação ao questionário de usabilidade

Perguntas	Respostas
Das tarefas solicitadas, quantas você conseguiu executar?	Todas
De um modo geral, você achou o protótipo intuitivo e fácil de usar?	Sim
Você achou fácil cadastrar / alterar / excluir uma regra?	Concordo parcialmente
Você achou fácil consultar regras referente a um módulo com defeito?	Concordo totalmente
Você considera útil a utilização de um sistema especialista para buscar a solução de defeitos no sistema de automação?	Concordo totalmente
Você considera aceitável o tempo gasto para encontrar a solução utilizando o sistema especialista quando houver defeitos no sistema de automação?	Concordo totalmente
Você acha que o protótipo desenvolvido diminuirá a quantidade de ligações com pedidos de auxílio na correção de falhas do sistema de automação?	Concordo totalmente
Você acha que seus clientes irão gostar da aplicação?	Concordo parcialmente
Qual é a sua avaliação do protótipo?	Muito bom
Qual foi a sua maior dificuldade utilizando o protótipo?	Falta de familiarização com informática

O engenheiro considerou a aplicação muito útil e fácil de usar na busca de soluções através da consulta. Também achou intuitivo e fácil de usar, embora tenha concordado parcialmente quanto à facilidade de cadastrar / alterar / excluir regra, devido a ter tido dificuldade nas primeiras tentativas de cadastro. Referente a considerar aceitável o tempo gasto para encontrar uma solução, respondeu concordar totalmente assim como concordou totalmente que o protótipo desenvolvido diminuirá a quantidade de ligações recebidas por ele pedindo auxílio na correção de falhas no sistema de automação. Avaliou o protótipo como muito bom, porém concordou parcialmente que os clientes irão gostar da aplicação, pois afirmou que nem todos os usuários são familiarizados com informática, e isso pode ser considerado um empecilho na utilização do sistema. Mencionou como maior dificuldade de utilização a falta de familiarização com a informática, o que tornou lento o entendimento do funcionamento da aplicação.

3.4.1.3 Teste realizado com o usuário cliente

Primeiramente o usuário foi orientado sobre o objetivo do teste, da aplicação e de suas funcionalidades. A lista de tarefas foi composta de três tarefas que teve por objetivo contemplar todos objetivos e funcionalidades da aplicação por parte de usuário cliente. Ao finalizar cada tarefa, foi solicitado que ele informasse se esta foi executada, caso tenha sido identificado algum problema ou se havia alguma observação a ser feita.

O questionário de usabilidade do usuário foi composto por oito perguntas, sendo sete fechadas e uma aberta. Para cinco das questões fechadas foi disponibilizada a opção de informar uma observação. As perguntas tiveram por objetivo obter as impressões do usuário sobre a usabilidade e entendimento em geral da aplicação. Além disso, buscou-se avaliar se o usuário pretende utilizar o sistema especialista antes de entrar em contato com o engenheiro solicitando auxílio. Os resultados referentes à lista de tarefas desempenhadas e a usabilidade do usuário são respectivamente apresentados nas seções 3.4.1.3.1 e 3.4.1.3.2.

3.4.1.3.1 Análise dos resultados da lista de tarefas do usuário

No Quadro 11 são apresentados os resultados quanto às questões envolvendo os processos desempenhados pelo usuário na aplicação.

Quadro 11 - Respostas quanto à lista de tarefas

Tarefas	Executou?
Consultar uma regra em consulta	Sim
Fazer as verificações solicitadas pelo sistema	Sim
Validar se a sequência de perguntas faz sentido e solução mostrada é útil para o problema	Não

Considerando os resultados das questões da lista de tarefas exibidas no Quadro 11, percebeu-se que as tarefas relacionadas ao funcionamento do protótipo que incluem consultar a regra e seguir a sequência foram executadas com sucesso, o que indica o bom funcionamento e a facilidade no uso da aplicação. A dificuldade aparente foi encontrada na confiabilidade da sequência e solução apresentadas, pois a observação do usuário foi a seguinte “A base de dados parou de fazer novas perguntas e me deu uma causa do problema que não confere com o que aconteceu. Poderia ter mais um passo de perguntas.”.

3.4.1.3.2 Análise dos resultados de usabilidade do usuário

Após análise dos resultados da lista de tarefas do usuário, foram analisados os resultados obtidos do questionário de usabilidade. Os resultados constam no Quadro 12.

Quadro 12 - Respostas do usuário em relação ao questionário de usabilidade

Perguntas	Respostas
Das tarefas solicitadas, quantas você conseguiu executar?	A maior parte delas
De um modo geral, você achou o protótipo intuitivo e fácil de usar?	Sim
Você achou fácil consultar regras referente a um módulo com defeito?	Concordo totalmente
Você considera útil a utilização de um sistema especialista para buscar a solução de defeitos no sistema de automação?	Concordo totalmente
Você considera aceitável o tempo gasto para encontrar a solução utilizando o sistema especialista quando houver defeitos no sistema de automação?	Concordo totalmente
Você utilizaria/consultaria o protótipo desenvolvido para tentar solucionar o problema antes de entrar em contato com a Controtec solicitando ajuda?	Concordo totalmente
Qual é a sua avaliação do protótipo?	Muito bom
Qual foi a sua maior dificuldade utilizando o protótipo?	Nenhuma

O usuário cliente considerou a aplicação intuitiva e de fácil utilização, não tendo nenhuma dificuldade na utilização, visto que não precisa aprender muitas coisas para operá-la. Concordou totalmente quanto ao tempo gasto para encontrar uma solução ser aceitável, assim como concordou totalmente que pretende utilizar o sistema antes de entrar em contato com a empresa solicitando ajuda. A avaliação do protótipo foi considerada muito boa e acrescentou em observação que com algumas melhorias relacionadas ao banco de dados, pode ficar muito útil.

3.4.2 Comparação com trabalhos correlatos e discussões

O Quadro 13 apresenta a análise comparativa entre as características do sistema desenvolvido com as características existentes nos trabalhos relacionados.

Quadro 13 – Comparação de trabalhos correlatos

Características/trabalhos	Argenton(1996)	Heinrich(2001)	Silva(2010)	Gretter(2015)
Plataforma	Desktop	Desktop	Desktop	Web
Método utilizado	Fatos e regras	Raciocínio baseado em casos	Árvore de decisão	Árvore de decisão
Linguagem de programação	Shell SPIRIT / PROLOG.	Delphi	Python	PHP
Banco de dados	N/A	Paradox	N/A	MySQL
Permite cadastrar novos casos	Sim	Sim	Sim	Sim

A partir do Quadro 13, pode-se observar que o protótipo desenvolvido diferencia-se dos demais trabalhos correlatos principalmente por ter sido desenvolvido para a Web, permitindo assim o acesso através de plataformas de acesso variadas. O método escolhido é o mesmo de Silva, porém foi desenvolvido de forma diferente, utilizando um banco de dados MySQL para armazenar as regras. Todos os trabalhos permitem cadastrar novos casos, porém

somente o protótipo desenvolvido e o trabalho de Heinrich possuem tela específica para essa funcionalidade, tornando-a mais amigável para o usuário.

4 CONCLUSÕES

O objetivo deste trabalho foi desenvolver um protótipo que auxilie o atendimento e diagnóstico de defeitos em sistemas de automação de motéis instalados pela Controtec. Os casos de uso propostos para o protótipo foram atendidos e os requisitos funcionais previstos para o desenvolvimento do sistema foram testados e validados.

O objetivo específico que tinha por necessidade a construção de uma base de conhecimento com defeitos que podem ocorrer no sistema de automação foi atingido, embora tenha sido observado que poderia ser aperfeiçoado, contendo mais verificações a serem realizadas e mais soluções possíveis. O cadastro desses novos registros assim como o cadastro para futuros defeitos é possível, pois foi criada a funcionalidade de cadastrar novas regras, permitindo ao engenheiro da empresa incluir as perguntas e soluções aplicadas por ele durante o atendimento.

O segundo objetivo específico que previa a implementação de um Sistema Especialista para emitir diagnóstico de falhas, possíveis causas e providências a serem tomadas foi atingido, pois a partir das regras cadastradas pelo engenheiro, é possível consultá-las e encontrar soluções, realizando as verificações solicitadas pelo sistema.

O mecanismo de contingência previsto no último objetivo específico também foi atingido, pois o sistema permite que no momento do cadastro de regras, o engenheiro registre uma contingência para a situação reportada.

Visto que este protótipo foi desenvolvido na plataforma web, tem-se com isso todas as vantagens e recursos já oferecidos pela plataforma, como a diversidade de plataformas de acesso, centralização da manutenção da aplicação e assim por diante.

4.1 LIMITAÇÕES

A principal limitação do protótipo é quanto às respostas binárias no momento da consulta. As perguntas são cadastradas de modo que as únicas respostas podem ser "sim" ou "não", ou seja, não há a possibilidade de cadastrar perguntas abertas. Além disso, não é possível identificar a porcentagem de certeza na resposta, indicando níveis diferentes de probabilidade, permitindo que o protótipo considere outras soluções.

4.2 EXTENSÕES

Algumas das extensões possíveis para este trabalho são:

- a) Aperfeiçoar a base de dados, criando mais verificações em forma de perguntas e

mais soluções possíveis;

- b) criar mecanismo para disponibilizar manuais de instrução para determinadas tarefas, assim como imagens de placa de automação, identificando os dispositivos que podem precisar de verificação e vídeos instrutivos para procedimentos mais complexos;
- c) criar a funcionalidade de gerar relatório da sequência seguida, podendo ser enviado ao engenheiro da empresa para análise das verificações realizadas;
- d) possibilitar a visualização das regras cadastradas em formato de árvore, para facilitar a visão da sequência de verificações da regra;
- e) aperfeiçoar a tela de cadastro, tornando-a mais auto explicativa e permitindo uma melhor visualização da regra que está sendo criada;
- f) Utilizar uma ferramenta de sistema especialista para aprimorar a edição da base, atuando como um motor de inferência.

REFERÊNCIAS

- ARGENTON, Paulo A. **Protótipo de um sistema especialista para atendimento à ocorrências em transformadores de potência.** 1996. 62 f. Trabalho de Conclusão de Curso (Bacharel em Ciências da Computação) - centro de Ciências Exatas e Naturais, Universidade Regional de Blumenau, Blumenau.
- GUTIERREZ, R.M.V.; PAN, S.S.K. **Complexo Eletrônico: Automação do Controle Industrial** - Disponível em:<<http://www.bndes.gov.br/conhecimento/bnset/set2807.pdf>>. Acesso em 26 nov. 2015.
- HEINRICH, Daniel J. **Sistema de apoio para o diagnóstico de defeitos em equipamentos eletrônicos aplicado a oficinas eletrônicas utilizando raciocínio baseado em casos.** 2001. 47 f. Trabalho de Conclusão de Curso (Bacharel em Ciências da Computação) - Centro de Ciências Exatas e Naturais, Universidade Regional de Blumenau, Blumenau.
- HOSOKAWA, Eric O. **Técnica de Árvore de Decisão em Mineração de Dados.** 2011. 40 f. Trabalho de Conclusão de Curso (Tecnologia em Processamento de Dados) – Faculdade de Tecnologia de São Paulo, São Paulo.
- MORAES, C. C. de; CASTRUCCI, P. L. **Engenharia de Automação Industrial.** 2.ed. LTC, 2007.
- PAULA, Ariel L. de. **Manual Expert SINTA.** 2013. Disponível em: <<http://www.ebah.com.br/content/ABAAAfFXUAB/manual-expert-sinta>>. Acesso em 12 abr. 2015.
- PINHEIRO, José M. S. **Sistemas de Automação.** 2004. Disponível em: <http://www.projetoderedes.com.br/artigos/artigo_sistemas_automacao.php>. Acesso em: 24 mar. 2015.
- PINTO, Fábio da C. **Sistemas de Automação e Controle.** 2005. Disponível em: <<http://www.ebah.com.br/content/ABAAAAPMUAJ/sistema-automacao-controle>>. Acesso em: 03 abr. 2015.
- POZZER, Cesar T. **Aprendizado por Árvore de Decisão.** 2006. Disponível em: <http://www-usr.inf.ufsm.br/~pozzer/disciplinas/pj3d_decisionTrees.pdf>. Acesso em: 15 dez. 2015.
- RIBEIRO, Horácio da C. e S. **Introdução aos Sistemas Especialistas.** Rio de Janeiro; São Paulo: LTC – Livros Técnicos e Científicos S.A, 1987.
- SILVA, Robson C. C. da. **Sistema Especialista de apoio à decisão para o diagnóstico de falhas em aeronaves.** 2010. 84 f. Trabalho de Conclusão de Curso (Bacharel em Engenharia da Computação) – Escola Politécnica de Pernambuco, Universidade de Pernambuco, Recife.
- TARSO, Paulo. **Sistemas Especialistas.** 2005. Disponível em: <<http://paulotarso.com/Files/FSI/SistemasEspecialistas.pdf>>. Acesso em 06 abr. 2015.
- WEG S.A. **Automação de Processos Industriais - PC12 Design Center.** Apostila para treinamento interno, Jaraguá do Sul, [2002]. Jaraguá do Sul: Weg S.A., 2002
- WEISS, Sholom M. e Kulikowski, Casimir A. **Guia prático para projetar sistemas especialistas.** Rio de Janeiro: LTC – Livros Técnicos e Científicos S.A, 1988.

APÊNDICE A – Descrição dos Casos de Uso

Esta seção apresenta a descrição dos casos de uso conforme previstos no diagrama apresentado na seção 3.3.1, conforme mostra o Quadro 14.

Quadro 14 - Descrição dos casos de uso

UC01 Manter usuários

Permite ao administrador criar contas para novos usuários e excluí-las quando necessário.

UC02 Efetuar login

Através da identificação por nome de usuário e senha permite ao usuário conectar-se ao sistema.

UC03 Realizar consulta

Para realizar uma consulta no sistema, o usuário deverá escolher qual módulo do sistema de automação está com defeito para que o sistema apresente verificações que devem ser feitas. O usuário precisa responder as verificações para que o sistema possa então identificar a razão do problema e apresentar soluções para a correção do defeito.

Constraints

Pré-condição: O usuário deve estar logado no sistema.

Pré-condição: O usuário deve saber o que está apresentando defeito no sistema de automação.

Pré-condição: Ao menos um módulo do sistema de automação deve estar cadastrado.

Pré-condição: Ao menos uma regra deve cadastrada para o módulo escolhido.

Pós-condição: O sistema especialista fará perguntas de verificação relacionadas ao módulo selecionado.

Pós-condição: Com base nas respostas do usuário, o sistema especialista apresentará uma solução cadastrada para o problema.

Cenários

Escolher cenário de defeito {Principal}

1. Sistema apresenta os módulos do sistema de automação cadastrados.
2. Usuário seleciona módulo do sistema de automação que esteja com problema dentre as opções apresentadas.
3. Sistema apresenta perguntas com verificações a serem feitas no sistema de automação.
4. Usuário realiza as verificações e responde o sistema.
5. Sistema interpreta a resposta das perguntas e encontra a solução cadastrada.
6. Sistema apresenta solução para a razão do defeito encontrada.
7. Usuário segue as instruções da solução apresentada.

Não há regra cadastrada para o problema {Exceção}

No passo 5, caso o usuário não encontre uma verificação para o problema dele, o sistema orienta que o usuário entre em contato com a Controtec solicitando auxílio.

UC04 Visualizar medidas de contingência

Quando não houver solução imediata e existir alguma contingência cadastrada para o problema, o sistema deve apresentar a contingência cadastrada para o problema.

Constraints

Pré-condição: O sistema deve ter apresentado uma solução para a razão do defeito identificada.

Pré-condição: Ao menos uma contingência deve estar cadastrada.

Pós-condição: O sistema deve apresentar uma contingência para a razão do defeito identificada.

Cenários**Visualizar contingência {Principal}**

1. Sistema apresenta contingência para a razão do defeito encontrada.
2. Usuário segue as instruções da contingência apresentada.

UC05 Manter perguntas

Quando for identificada a necessidade de um novo teste a ser realizado, o administrador tem a possibilidade de cadastrar uma nova regra, com uma ou mais perguntas. Além disso, terá possibilidade de editar ou excluir quando houver necessidade.

UC06 Manter soluções

Quando for encontrada nova solução para os defeitos do sistema de automação, o administrador tem a possibilidade de cadastrar uma nova solução. Além disso, terá possibilidade de editá-las excluí-las quando as mesmas não forem mais eficientes.

UC07 Manter contingências

Quando for encontrada nova contingência para os defeitos do sistema de automação, o administrador tem a possibilidade de cadastrar uma nova contingência. Além disso, terá possibilidade de editá-las excluí-las quando as mesmas não forem mais eficientes.

APÊNDICE B – Dicionário de Dados

Este Apêndice apresenta a descrição das tabelas do banco de dados apresentadas na seção de especificação deste trabalho. Nos quadros abaixo estão o dicionário de dados das tabelas da aplicação. Os tipos de dados utilizados nos atributos são:

- a) *integer*: armazena números inteiros;
- b) *varchar*: armazena caracteres alfanuméricos;
- c) *char*: corresponde a caracteres.

No Quadro 15, apresenta-se a tabela de área.

Quadro 15 - Tabela de área

Área				
Armazena as informações de área do sistema de informação.				
Campo	Descrição	Tipo	Chave Primária	Chave Estrangeira
cod_area	Código único de identificação da área.	<i>INTEGER</i>	Sim	Não
desc_area	Nome da área	<i>VARCHAR</i>	Não	Não

No Quadro 16, apresenta-se a tabela de módulo.

Quadro 16 - Tabela de módulo

Módulo				
Armazena as informações de módulo do sistema de informação.				
Campo	Descrição	Tipo	Chave Primária	Chave Estrangeira
cod_mod	Código único de identificação do módulo.	<i>INTEGER</i>	Sim	Não
desc_mod	Nome do módulo	<i>VARCHAR</i>	Não	Não
cod_area	Código único de identificação da área.	<i>INTEGER</i>	Não	Sim

No Quadro 17, apresenta-se a tabela de usuário.

Quadro 17 - Tabela de usuário

Usuario				
Armazena as informações de usuário.				
Campo	Descrição	Tipo	Chave Primária	Chave Estrangeira
id	Código único de identificação do usuário.	<i>INTEGER</i>	Sim	Não
nome	Nome do usuário.	<i>VARCHAR</i>	Não	Não
usuario	Usuário utilizado para login no sistema.	<i>VARCHAR</i>	Não	Não
senha	Senha do usuário para login no sistema.	<i>VARCHAR</i>	Não	Não
tipo	Código para identificar se o usuário é administrador ou usuário comum.	<i>INTEGER</i>	Não	Não

No Quadro 18, apresenta-se a tabela de *regra*.

Quadro 18 - Tabela de *regra*

Regra				
Armazena as informações de regras.				
Campo	Descrição	Tipo	Chave Primária	Chave Estrangeira
cod_regra	Código único de identificação da regra.	<i>INTEGER</i>	Sim	Não
num	Número que identifica a variável na regra.	<i>VARCHAR</i>	Não	Não
cod_var	Código único de identificação da variável.	<i>INTEGER</i>	Não	Sim
resposta	Indetifica se a variável é resposta ou não.	<i>CHAR</i>	Não	Não
contingencia	Indetifica se a variável é contingência ou não.	<i>CHAR</i>	Não	Não
cod_mod	Código único de identificação do módulo.	<i>INTEGER</i>	Sim	Sim

No Quadro 19, apresenta-se a tabela de *variável*.

Quadro 19 - Tabela de *variável*

Variável				
Armazena as informações de variáveis das regras cadastradas.				
Campo	Descrição	Tipo	Chave Primária	Chave Estrangeira
cod_var	Código único de identificação da variável.	<i>INTEGER</i>	Sim	Não
desc_var	Nome da variável.	<i>VARCHAR</i>	Não	Não
cod_area	Código único de identificação do módulo.	<i>INTEGER</i>	Não	Sim

APÊNDICE C – Roteiro e questionário de avaliação de usabilidade

Neste apêndice constam o questionário e o roteiro de testes que os usuários seguiram. O Quadro 20 contém a lista de tarefas que conduz o administrador, testando as funcionalidades do protótipo. No Quadro 21 consta o questionário de usabilidade feito para o administrador. O Quadro 22 contém a lista de tarefas que conduz o usuário cliente, testando as funcionalidades do protótipo. No Quadro 23 consta o questionário de usabilidade feito para o usuário cliente.

Quadro 20 - Lista de tarefas para administrador

INSTRUÇÕES

Com este questionário buscamos avaliar a utilização do protótipo de sistema especialista para auxílio no diagnóstico de defeitos em sistemas de automação de motéis.

Um dos objetivos desse trabalho é a construção de uma base de conhecimento com defeitos que podem ocorrer no sistema de automação. Outro objetivo é implementar um sistema especialista que possa emitir o diagnóstico de falhas, as possíveis causas e as providências a serem tomadas para a manutenção do sistema de automação. Sempre que possível, também pretende-se a disponibilização de soluções de contingência até que a visita do técnico seja realizada.

Lista de tarefas a serem executadas pelo administrador:

1) Crie uma nova regra na tela de cadastro, contendo mais de uma resposta e uma contingência.

A tarefa foi executada? () Sim () Não

Observação: _____

2) Consulte a regra criada e as demais regras existentes na tela de regras cadastradas.

A tarefa foi executada? () Sim () Não

Observação: _____

3) Ainda na tela de regras cadastradas, escolha uma regra pra realizar uma alteração e clique em Alterar. Altere alguma informação e salve. Volte na tela de regras cadastradas para validar se a alteração foi salva.

A tarefa foi executada? () Sim () Não

Observação: _____

4) Na lista de regras cadastradas, escolha uma para excluir e valide se ela não é mais exibida nas regras cadastradas.

A tarefa foi executada? () Sim () Não

Observação: _____

5) Crie um novo usuário do tipo cliente que irá fazer a consulta de regras.

A tarefa foi executada? () Sim () Não

Observação: _____

6) Consulte a regra que você criou na tela de consulta.

A tarefa foi executada? () Sim () Não

Observação: _____

7) Siga a sequência da regra criada até chegar a uma resposta ou contingência. A sequência apresentada está correta de acordo com o que você cadastrou?

A tarefa foi executada? () Sim () Não

Observação: _____

8) Consulte mais uma regra já existente associada a outro módulo na tela de consulta. A sequência da regra está sendo seguida corretamente e a solução apresentada é útil para o problema?

A tarefa foi executada? () Sim () Não

Observação: _____

Quadro 21 - Questionário de usabilidade para administrador

QUESTIONÁRIO DE USABILIDADE

1. Das tarefas solicitadas, quantas você conseguiu executar?

- Todas
- A maior parte delas
- Metade das tarefas
- Menos da metade das tarefas
- Nenhuma tarefa

2. De um modo geral, você achou o protótipo intuitivo e fácil de usar?

- Sim Não

3. Você achou fácil cadastrar / alterar / excluir uma regra?

- Concordo totalmente
- Concordo parcialmente
- Indiferente
- Discordo parcialmente
- Discordo totalmente

Observação: _____

4. Você achou fácil consultar regras referente a um módulo com defeito?

- Concordo totalmente
- Concordo parcialmente
- Indiferente
- Discordo parcialmente
- Discordo totalmente

Observação: _____

5. Você considera útil a utilização de um sistema especialista para buscar a solução de defeitos no sistema de automação?

- Concordo totalmente
- Concordo parcialmente
- Indiferente
- Discordo parcialmente
- Discordo totalmente

Observação: _____

6. Você considera aceitável o tempo gasto para encontrar a solução utilizando o sistema especialista quando houver defeitos no sistema de automação?

- Concordo totalmente
- Concordo parcialmente
- Indiferente
- Discordo parcialmente
- Discordo totalmente

Observação: _____

7. Você acha que o protótipo desenvolvido diminuirá a quantidade de ligações com pedidos de auxílio na correção de falhas do sistema de automação?

- Concordo totalmente
- Concordo parcialmente
- Indiferente
- Discordo parcialmente
- Discordo totalmente

Observação: _____

8. Você acha que seus clientes irão gostar da aplicação?

- Concordo totalmente
- Concordo parcialmente
- Indiferente
- Discordo parcialmente
- Discordo totalmente

Observação: _____

9. Qual é a sua avaliação do protótipo?

- Muito bom
- Bom
- Regular
- Insatisfatório

Observação: _____

10. Qual foi a sua maior dificuldade utilizando o protótipo?

Muito obrigada pela participação!

Quadro 22 - Lista de tarefas para usuário cliente

INSTRUÇÕES

Com este questionário buscamos avaliar a utilização do protótipo de sistema especialista para auxílio no diagnóstico de defeitos em sistemas de automação de motéis.

Um dos objetivos desse trabalho é a construção de uma base de conhecimento com defeitos que podem ocorrer no sistema de automação. Outro objetivo é implementar um sistema especialista que possa emitir o diagnóstico de falhas, as possíveis causas e as providências a serem tomadas para a manutenção do sistema de automação. Sempre que possível, também pretende-se a disponibilização de soluções de contingência até que a visita do técnico seja realizada.

Lista de tarefas a serem executadas pelo cliente:

1) Consulte uma regra de um módulo de sua escolha na tela de consulta.

A tarefa foi executada? () Sim () Não

Observação: _____

2) Siga a sequência da regra apresentada até chegar a uma resposta ou contingência. Foi possível fazer as verificações solicitadas pelo sistema?

A tarefa foi executada? () Sim () Não

Observação: _____

3) Consulte mais uma regra associada a outro módulo na tela de consulta. A sequência da regra faz sentido e a solução mostrada apresenta ser útil para o problema?

A tarefa foi executada? () Sim () Não

Observação: _____

Quadro 23 - Questionário de usabilidade para usuário cliente

QUESTIONÁRIO DE USABILIDADE

1. Das tarefas solicitadas, quantas você conseguiu executar?

- Todas
- A maior parte delas
- Metade das tarefas
- Menos da metade das tarefas
- Nenhuma tarefa

2. De um modo geral, você achou o protótipo intuitivo e fácil de usar?

- Sim Não

3. Você achou fácil consultar regras referente a um módulo com defeito?

- Concordo totalmente
- Concordo parcialmente
- Indiferente
- Discordo parcialmente
- Discordo totalmente

Observação: _____

4. Você considera útil a utilização de um sistema especialista para buscar a solução de defeitos no sistema de automação?

- Concordo totalmente
- Concordo parcialmente
- Indiferente
- Discordo parcialmente
- Discordo totalmente

Observação: _____

5. Você considera aceitável o tempo gasto para receber a solução utilizando o sistema especialista quando houver defeitos no sistema de automação?

- Concordo totalmente
- Concordo parcialmente
- Indiferente
- Discordo parcialmente
- Discordo totalmente

Observação: _____

6. Você utilizaria/consultaria o protótipo desenvolvido para tentar solucionar o problema antes de entrar em contato com a Controtec solicitando ajuda?

- Concordo totalmente
- Concordo parcialmente
- Indiferente
- Discordo parcialmente
- Discordo totalmente

Observação: _____

7. Qual é a sua avaliação do protótipo?

- Muito bom
- Bom
- Regular
- Insatisfatório

Observação: _____

8. Qual foi a sua maior dificuldade utilizando o protótipo?

Muito obrigada pela participação!