

DAS Departamento de Automação e Sistemas
CTC **Centro Tecnológico**
UFSC Universidade Federal de Santa Catarina

Desenvolvimento de Ferramenta para Especificação dos Intertravamentos de Segurança para CLP na Indústria de Petróleo e Gás

*Relatório submetido à Universidade Federal de Santa Catarina
como requisito para a aprovação na disciplina*

DAS 5511: Projeto de Fim de Curso

Guilhermo Keiji Saito

Florianópolis, Agosto de 2016

Desenvolvimento de Ferramenta para Especificação dos Intertravamentos de Segurança para CLP na Indústria de Petróleo e Gás

Guilhermo Keiji Saito

Esta monografia foi julgada no contexto da disciplina
DAS5511: Projeto de Fim de Curso
e aprovada na sua forma final pelo
Curso de Engenharia de Controle e Automação

Prof. Max Hering de Queiroz

Assinatura do Orientador

Banca Examinadora:

Prof. Max Hering de Queiroz
Orientador da Empresa e no Curso

Ricardo José Rabelo
Presidente da Banca

Prof. Rodrigo Scpak
Avaliador

Eduardo Luiz Santos da Silva
Gustavo Sena Mafra
Debatedores

Agradecimentos

Agradeço primeiramente à minha família pelo eterno apoio e constante motivação aos estudos desde a infância.

Aos meus amigos, pelos momentos sempre agradáveis de descontração que passamos juntos.

Ao meu orientador, Prof. Max, por me dar a oportunidade de participar do projeto e pela excelente orientação oferecida.

Aos outros membros do projeto pelas ideias e discussões para o desenvolvimento deste trabalho.

Resumo

Os programas de automação na indústria de petróleo e gás, seja o programa voltado para sistemas de segurança ou de uso geral, são projetados com base em especificações de segurança, definidas em projeto. Falhas ocorridas durante estas especificações podem gerar sérios danos a equipamentos, prejuízos ambientais, financeiros ou até mesmo a perda de vidas humanas. Este trabalho consiste no desenvolvimento de uma ferramenta computacional que tem como objetivo padronizar o formato das especificações de intertravamentos de segurança encontradas na Matriz Causa e Efeito que, atualmente, nos sistemas instrumentados de segurança da plataforma P-51 – estudo de caso deste trabalho – estão ocorrendo divergências no padrão de preenchimento da matriz e que fazem com que reduza significativamente a confiabilidade nestes sistemas, já que as especificações contidas na Matriz Causa e Efeito são utilizadas para a implementação e validação de programas de Controladores Lógicos Programáveis voltados para sistema de segurança. Para o desenvolvimento desta ferramenta utilizou-se conceitos de Engenharia de Software com o emprego o modelo cascata de processo de software, a fim de se organizar o trabalho e produzir documentos que auxiliem na implementação e também na manutenção da ferramenta. O modelo cascata prevê 5 atividades fundamentais e que foram utilizadas no desenvolvimento: Análise e especificação dos requisitos – na qual foram levantados e especificados os requerimentos da ferramenta de acordo com as necessidades do projeto; Projeto – na qual foi descrita parte da estrutura da ferramenta; Implementação – em que realizou-se a codificação de acordo com os requisitos e o projeto de estrutura; Testes – para a verificação de erros ou anomalias da ferramenta ; Implantação – em que realizou-se a apresentação da ferramenta para o cliente. Assim, desenvolveu-se uma ferramenta computacional capaz de realizar a edição de Matrizes Causa e Efeito em um formato padrão através de uma interface gráfica simples e intuitiva. Para a avaliação da ferramenta efetuou-se a edição de exemplos de Matrizes da plataforma P-51 e a prova de conceito com engenheiros da UO-RIO e do CENPES.

Abstract

The automation programs in the oil and gas industry, is the program focused on safety or general purpose systems are designed based on safety requirements defined in the project. Failures occurred during these specifications may cause serious equipment damage, environmental damage, financial or even the loss of human lives. This work is the development of a computational tool that aims to standardize the format of safety interlocks specifications found in the Cause Mother and effect that currently in safety instrumented systems for the P-51 platform - case study of this work - are occurring differences in the matrix fill pattern and make significantly reduces the reliability of these systems, since the specifications contained in the Cause and Effect matrix are used for the implementation and validation of Programmable Logic Controllers programs for security system. For the development of this tool was used concepts of Software Engineering by employing the cascade model software process in order to organize the work and produce documents that assist in deployment and also maintaining the tool. The waterfall model provides 5 core activities and that were used in development: analysis and specification of requirements - in which they were raised and specified the requirements of the tool according to the project requirements; Design - in which was described the structure of the tool; Implementation - in which there was the encoding according to the requirements and structure of the project; Tests - to check for errors or anomalies tool; Implementation - where was held the presentation tool for the customer. Therefore, were developed a computational tool able to perform Matrix editing Cause and Effect in a standard format through a simple and intuitive graphical interface. For the evaluation tool made to edit Matrix examples of P-51 platform and the proof of concept with engineers UO-RIO and the CENPES.

Sumário

Capítulo 1: Introdução	1
1.1: Motivação e Justificativa	1
1.2: Objetivos	3
1.2.1: Objetivo geral	3
1.2.2: Objetivos específicos	3
1.3: Estrutura do documento	3
Capítulo 2: Caracterização do problema	5
2.1: Metodologia atual de desenvolvimento de Programas de CLP na Petrobras	5
2.2: Metodologia proposta pelo projeto	7
2.3: Definição do Problema	9
2.4: Proposta de ferramenta para edição de matriz C&E	11
2.5: Comentários Finais	12
Capítulo 3: Padrão para especificação de Matriz Causa e Efeito	13
3.1: Fundamentos da Matriz Causa e Efeito	13
3.2: Padrão de elaboração de Matriz de Causa e Efeito na Petrobras	14
3.3: Comentários Finais	18
Capítulo 4: Projeto da Ferramenta	19
4.1: Engenharia de Software/Metodologia de desenvolvimento de sistemas	19
4.2: Modelo de processo de software adotado	21
4.3: Análise e definição dos requisitos	22
4.3.1: Requisitos Funcionais	23

4.3.2: Requisitos Não Funcionais.....	25
4.4: Projeto/modelagem.....	25
4.4.1: Diagrama de classes.....	25
4.4.2: Projeto interface gráfica	28
4.4.3: Estruturação do XML	29
4.5: Comentários Finais.....	32
Capítulo 5: Desenvolvimento da Ferramenta	33
5.1: Implementação	33
5.2: Testes	34
5.3: Implantação e manutenção.....	35
5.4: Comentários Finais.....	35
Capítulo 6: Avaliação dos resultados	37
6.1: Interface gráfica.....	37
6.2: Geração de arquivo XML.....	41
6.3: Geração de arquivo PDF	42
6.4: Prova de conceito.....	43
6.5: Comentários Finais.....	44
Capítulo 7: Conclusões e Perspectivas.....	45
7.1: Diretrizes para trabalhos futuros	46
Bibliografia:	47
Apêndice A – Estrutura tipo árvore do arquivo XML projetado	49

Simbologia

CLP – Controladores Lógicos Proporcionais

SIS – Sistemas Instrumentados de Segurança

P&ID – Process and Instrumentation Diagram

XML – eXtensible Markup Language

PDF – Portable Document Format

C&E – Causa e Efeito

ET – Especificação Técnica

TAF – Teste de Aceitação de Fábrica

DOM – Document Object Model

Lista de Figuras

- Figura 1: Causas de acidentes resultados de falhas de segurança [2]
- Figura 2: Metodologia utilizada na Petrobras.[6]
- Figura 3: Metodologia proposta pelo projeto [6]
- Figura 4: Etapas da metodologia [6]
- Figura 5: Interface da matriz nas metodologias
- Figura 6: Esquema da interface atual da matriz
- Figura 7: Visão geral da ferramenta
- Figura 8: Esquema da matriz C&E
- Figura 9: Matriz C&E utilizada na Petrobras [3]
- Figura 10: Informações da matriz divididas em áreas
- Figura 11: Engenharia de Software em Camadas [8]
- Figura 12: Etapas do modelo cascata
- Figura 13: Diagrama de classes
- Figura 14: Diagrama de navegação
- Figura 15: Estrutura de documentos XML [10]
- Figura 16: Divisão da matriz para a estruturação do XML
- Figura 17: Árvore simplificada da estrutura do XML(Árvore completa no apêndice)
- Figura 18: Ocupação de memória da ferramenta
- Figura 19: Janela principal com a interface da matriz
- Figura 20: Janela para inserção de causas
- Figura 21: Janela para inserção dos efeitos

Figura 22: Janela para inserção da lógica

Figura 23: Janela para inserção de notas

Figura 24: Janela para inserção de documentos de referência

Figura 25: Janela para inserção de dados do rodapé

Figura 26: Exemplo de causa

Figura 27: Exemplo de efeito

Figura 28: Exemplo de documento XML gerado pela ferramenta

Figura 29: Exemplo de documento PDF gerado pela ferramenta

Lista de Tabelas

Tabela 1: Simbologia das lógicas

Tabela 2: Padrão de “tag” a ser utilizado

Tabela 3: Exemplo de grupo de sensores

Tabela 4: Requisitos Funcionais

Tabela 5: Requisitos Não-Funcionais

Capítulo 1: Introdução

1.1: Motivação e Justificativa

Sistemas críticos são sistemas geralmente encarregados de atividades de controle que requerem alto grau de confiabilidade, tendo em vista que falhas em tais sistemas podem levar a danos sérios de equipamentos de custo elevado, danos ambientais e até em perdas de vidas humanas [1]. A necessidade de atender os requisitos rigorosos deste tipo de sistemas exige que os projetos os levem em conta e que tenham as suas especificações claramente definidas e também que as implementações sejam previamente validadas [1]. No caso da indústria de petróleo e gás, sistemas instrumentados de segurança (SIS) são conjuntos de sensores, dispositivos lógicos e atuadores utilizados para garantir a segurança operacional de instalações industriais. Exemplos típicos são: Sistema de parada de emergência (*ESD-Emergency shutdown*); Sistema de parada de segurança (*Safety shutdown*); Sistema de intertravamento de segurança; Sistema de fogo e gás. A complexidade e criticidade de determinados sistemas automatizados como aqueles encontrados na indústria do petróleo gera a necessidade de se garantir que a lógica sendo utilizada nos controladores lógico-programáveis (CLPs) e sua integração com os dispositivos de medição e atuação atende as especificações do sistema instrumentado de segurança.

A atual metodologia para o desenvolvimento de sistemas de automação ligado a segurança na Petrobras consiste em uma sequência de passos começando a partir de especificações básicas da planta a ser automatizada e termina com o desenvolvimento de um software utilizado em um CLP para automação. Esta metodologia utiliza um conjunto de normas gerais e principalmente internas, como visto em (COMISSÃO DE NORMAS TÉCNICAS, 2002) e (COMISSÃO DE NORMAS TÉCNICAS, 2012). De acordo com essas normas, uma série de documentos que contenham informações relevantes ao longo do projeto são criados. A Matriz de Causa e Efeito (matriz C&E), por exemplo, é um documento que define a relação entre sinais de campo que indicam situações críticas e ações de segurança

adequadas. A correta especificação deste documento é essencial para se obter confiabilidade nos sistemas instrumentados de segurança, uma vez que ele possui um papel fundamental tanto na programação quanto na validação dos SIS. Erros no processo de especificação pode levar a sérios prejuízos, visto que maioria dos acidentes resultados de falhas no sistema de segurança são causados por especificações incorretas e incompletas [2], como mostra a Figura 1.

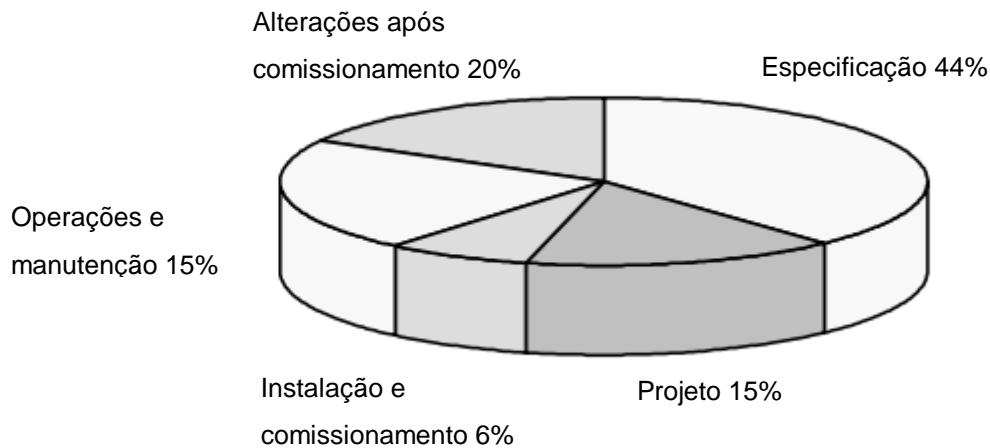


Figura 1: Causas de acidentes resultados de falhas de segurança [2]

O estudo de caso utilizado neste trabalho, o SIS da plataforma P-51 da Petrobras, retrata muito bem a problemática das falhas nas especificações, já que a atual metodologia prevê a elaboração de matriz C&E a partir de regras presentes uma diretriz visto em [3], mas que muitas vezes não são cumpridas, pois são utilizadas planilhas eletrônicas para a criação do documento que torna o documento susceptível a erros de preenchimento, uma vez que o usuário é livre para inserir quaisquer dados na matriz sem nenhuma restrição.

Dada a atual abordagem de especificação de intertravamentos na Petrobras, mostra-se importante a utilização de uma ferramenta que possibilite uma maior facilidade e confiabilidade das especificações para que se evite possíveis falhas na elaboração destes documentos.

1.2: Objetivos

1.2.1: Objetivo geral

Este trabalho se origina no contexto de um projeto de pesquisa e desenvolvimento no Departamento de Automação e Sistemas juntamente com a Petrobras intitulado “Validação de sistemas de controle e automação na indústria do petróleo e gás utilizando métodos de teste, verificação e síntese de programas”, que visa a operação segura e em conformidade com as especificações de funcionamento das unidades de produção *off-shore*. Especificamente neste projeto fim de curso, pretende-se, a partir de uma metodologia de desenvolvimento de sistemas, desenvolver um protótipo de ferramenta para a especificação dos intertravamentos de segurança para CLPs na indústria de petróleo e gás, baseado no estudo de caso da Petrobras.

1.2.2: Objetivos específicos

Todas as especificações dos intertravamentos de segurança para CLPs do estudo de caso abordado neste trabalho, são especificadas em um documento denominado matriz C&E, que por sua vez define a relação entre sinais de campo que indicam situações críticas e ações de segurança adequadas. O objetivo deste trabalho consiste no desenvolvimento de uma ferramenta para definição de um padrão de construção desta matriz através da criação de uma interface gráfica simples e clara a fim de facilitar usuário na elaboração da matriz e impedir que ela seja preenchida com informações inconsistentes e fora do padrão definido. Outra finalidade da ferramenta é gerar um documento PDF da matriz e também estabelecer uma interface para ferramentas de teste e verificação formal a partir da geração de arquivo XML. Por fim, objetiva-se realizar a avaliação da ferramenta em um estudo de caso real da Petrobras – o SIS da plataforma P-51.

1.3: Estrutura do documento

Na sequência deste capítulo introdutório, são apresentados no capítulo 2 a metodologia atual de desenvolvimento de CLP na Petrobras juntamente com a

descrição do projeto global e a caracterização do problema seguida da proposta do desenvolvimento desta ferramenta para a criação da matriz de causa e efeito. No capítulo 3 são apresentados os fundamentos da matriz C&E assim como o padrão atual de sua elaboração na Petrobrás. O capítulo 4 trata do projeto da ferramenta, no qual serão apresentados conceitos de metodologias de desenvolvimento de sistemas, a metodologia adotada para o desenvolvimento da ferramenta e a fase de análise e definição de requisitos seguida da fase de projeto do software. O capítulo 5 aborda a implementação da ferramenta, testes realizados e a implantação dela. No capítulo 6 são apresentadas análises dos resultados obtidos com a ferramenta e finalmente, no capítulo 7, são apresentadas as conclusões deste trabalho e sugestões para trabalhos futuros.

Capítulo 2: Caracterização do problema

A metodologia de desenvolvimento de projetos de automação atualmente utilizada em empresas de petróleo e gás, como a Petrobras consiste em uma sequência de etapas que partem de definições básicas da planta a ser automatizada e culmina com o desenvolvimento do programa utilizado para automação de CLP. Durante o projeto, uma gama de documentos, cada qual contendo informações relevantes a uma ou várias etapas do projeto, é criada. Cada etapa dessa metodologia possui entradas e saídas na forma de documentos, sendo que os documentos na entrada de uma determinada etapa são aqueles que são necessários para a geração dos documentos de saída. Neste capítulo, a atual metodologia de desenvolvimento e os documentos serão explicados bem como a nova metodologia de teste automático e os problemas encontrados.

2.1: Metodologia atual de desenvolvimento de Programas de CLP na Petrobras

As normas internas da Petrobras, (N-2595 2012) (N-1883 2002), estabelecem uma metodologia para o desenvolvimento de um projeto de automação e/ou instrumentação. Essencialmente, as diretrizes das normas descrevem e discriminam a documentação necessária que deve ser criada tanto para a especificação como para a validação dos sistemas de controle.

Em relação aos documentos que influenciam diretamente no projeto do sistema de controle, podem ser citados por ordem de elaboração o Diagrama P&ID (Process and Instrumentation Diagram), Matriz C&E, Memorial Descritivo, Diagrama da Logica e Documento do TAF (Teste de Aceitação de Fábrica). A Figura 2 auxilia no entendimento do desenvolvimento de um projeto.

Na fase de definição da especificação do projeto cria-se o Diagrama P&ID que contém todas as informações de processo necessárias a seleção e dimensionamento dos instrumentos. A partir desse documento é então elaborada a Matriz C&E que representa o inter-relacionamento entre os eventos (causas) e as ações (efeitos),

que devem ocorrer de forma automática e controlada pelo sistema (SIS, BPCS, etc.). Esse documento é apresentado em uma forma matricial com as causas nas linhas e os efeitos nas colunas.

Em paralelo a criação da Matriz C&E prepara-se o Memorial Descritivo de Automação que deve conter informações básicas que permitam a completa especificação de equipamentos e instrumentos para os diversos sistemas de instrumentação como mostra a Figura 2.

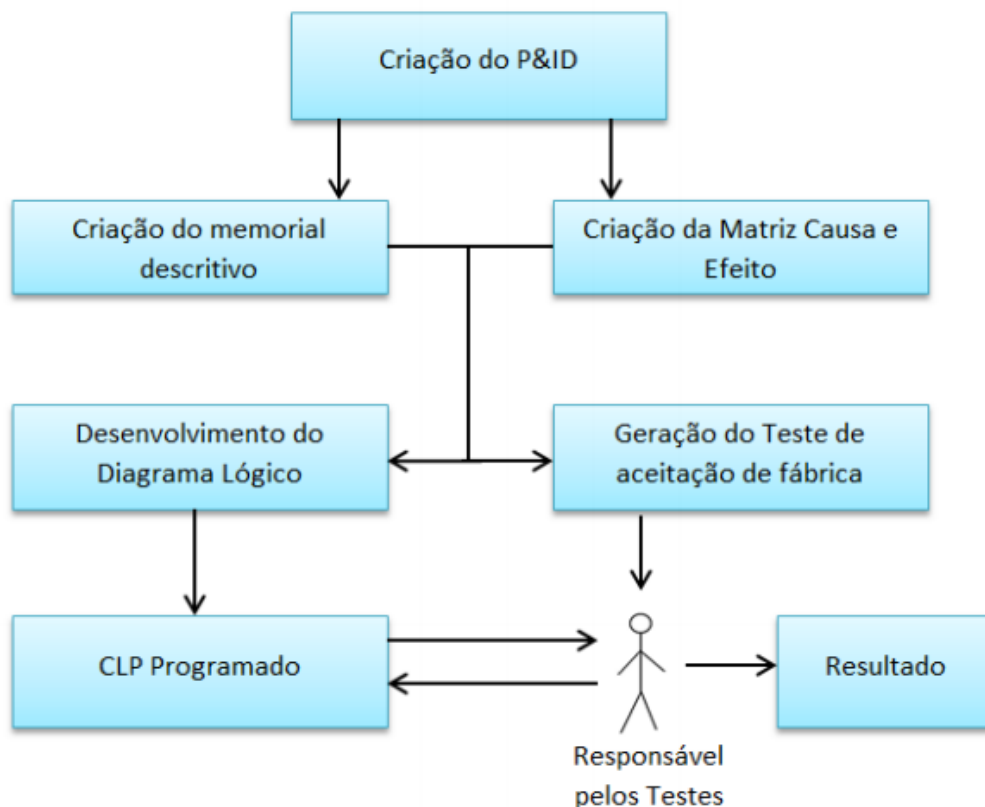


Figura 2: Metodologia utilizada na Petrobras.[6]

Em uma etapa seguinte é criado o Diagrama Lógico que tem por finalidade representar todas as logicas e funções discretas a serem executadas. Esse pode ser pensado como uma prévia do programa final do CLP. Por fim é gerado o Documento do TAF (Teste de Aceitação de Fábrica) que define uma lista sequencial de testes que deverão ser realizados para que então o programa do CLP seja validado.

Após a aprovação da documentação é então executado o FAT (Factory Acceptance Test). Caso seja identificada alguma irregularidade nos testes, faz-se necessária as correções do projeto e/ou do código do CLP, e ainda é agendada uma

nova bateria de testes. Só quando o sistema for devidamente testado e não apresentar mais inconformidades é que poderá ser encaminhado à planta para as fases de montagem e comissionamento.

2.2: Metodologia de teste automático

O projeto “Validação de sistemas de controle e automação na indústria do petróleo e gás utilizando métodos de teste, verificação e síntese de programas”, proposta por (Prati 2014), sugere uma nova metodologia para testes complementar à existente na Petrobras, como mostra a Figura 3 e tem como objetivos: não aumentar o tempo já gasto durante a etapa do TAF e oferecer a possibilidade de validar o sistema por uma ferramenta automática de testes.

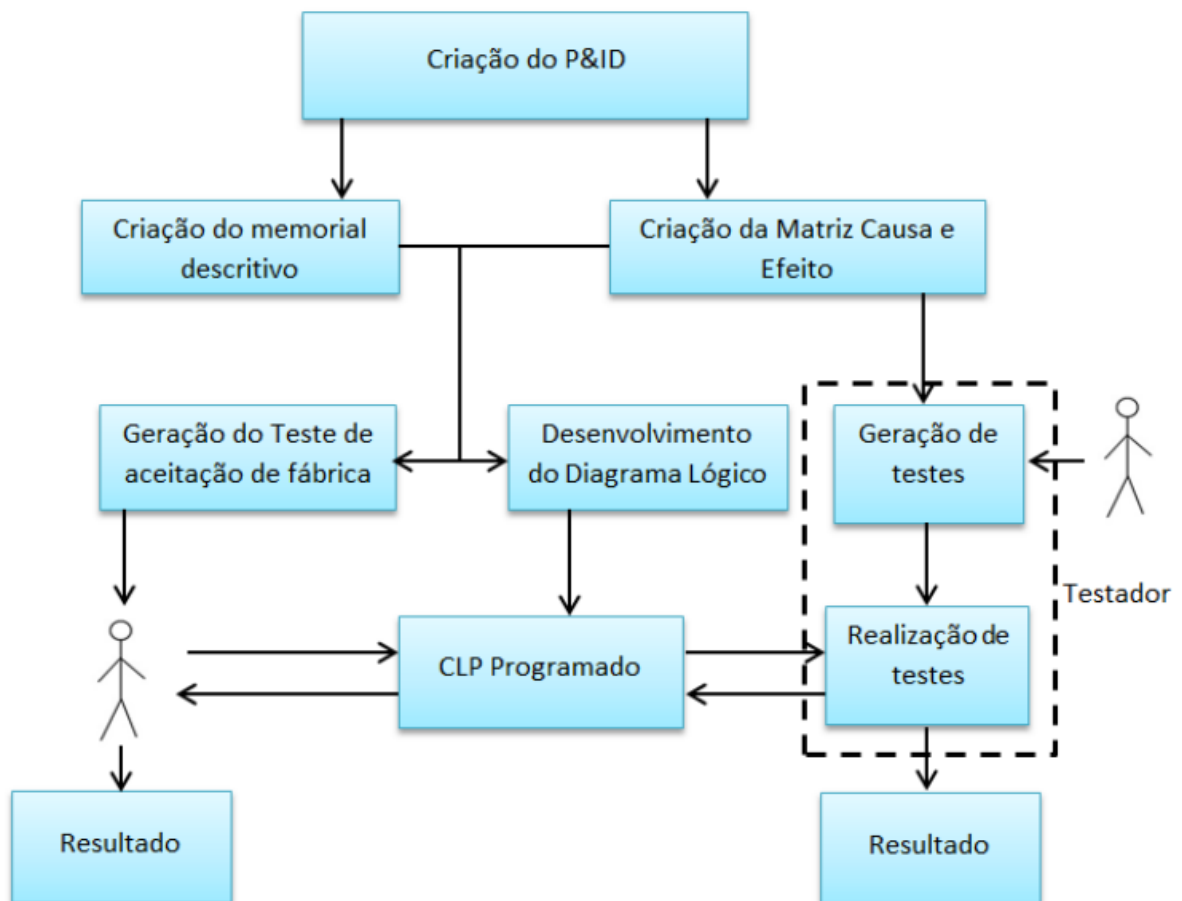


Figura 3: Metodologia proposta pelo projeto [6]

Para tanto foi desenvolvido um protótipo capaz de verificar o comportamento das funções de segurança do CLP e avaliar se o sistema atende as especificações impostas na Matriz C&E. O protótipo faz isso observando o comportamento das

saídas do CLP para determinados valores de sinais de entrada, e ainda, é capaz de dar um veredito se o SUT(System Under Test) possui ou não alguma falha na execução do teste, identificar essa falha e ainda apontar o erro no código do CLP.

O método de testes utilizado no trabalho faz uso de uma abordagem caixa-preta onde o acesso ao código do CLP é restrito. Com base na Figura 4, é descrita a sequência de passos do método.

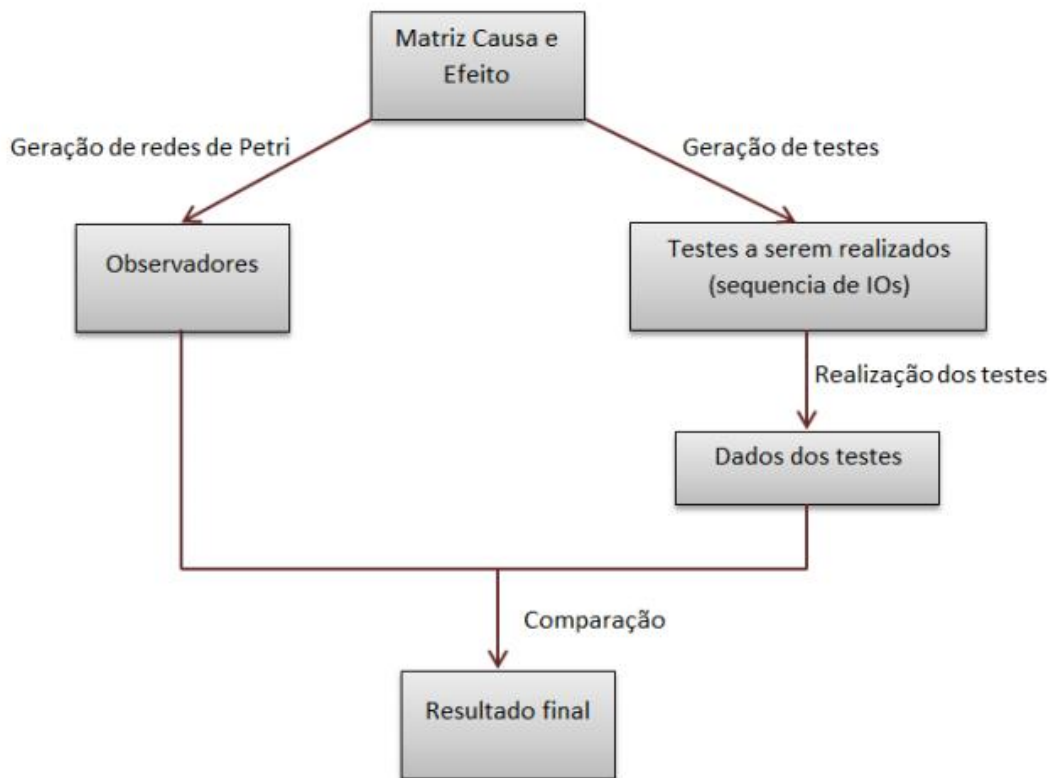


Figura 4: Etapas da metodologia [6]

A primeira etapa do método parte de informações contidas na Matriz C&E. São geradas sequências de testes e ainda criados observadores baseados em redes de Petri capazes de avaliar o comportamento do SUT. Uma justificativa para a escolha da Matriz C&E, como base para a criação dos observadores, é que ela possui informações claras e bem definidas do funcionamento da lógica básica e das funções de segurança do processo.

O processo de geração de testes resulta na definição de uma sequência de sinais de entradas para o CLP. Essa sequência é enviada ao CLP e os valores dos sinais da saída são gravados. Por sua vez, esses valores são enviados aos

observadores para que assim seja realizada uma análise do atendimento da especificação.

Inicialmente é feita a leitura de um arquivo de texto que possui todas as informações da Matriz C&E. A partir desse arquivo são gerados os observadores e a sequência de comandos que será enviada ao CLP. O envio dos comandos ao CLP é realizado através do protocolo OPC (OLE for Process Control).

Os dados dos sinais de entrada e saída do CLP são armazenados em uma tabela. Através de um jogador de redes de Petri, os dados dessa tabela são enviados aos observadores. O jogador de redes de Petri é responsável pela inserção de fichas de acordo com a presença dos sinais de entrada e saída que foram registrados na tabela. Por fim um log de resultados é gerado onde são identificadas as inconformidades encontradas.

2.3: Definição do Problema

Haja vista a importância da matriz C&E no desenvolvimento tanto da metodologia atual de elaboração de programas quanto da metodologia de testes proposta pelo projeto, percebe-se a necessidade deste documento possuir especificações corretas e confiáveis para que não ocorram falhas nos sistemas de segurança.

Dois documentos são utilizados para orientar a elaboração da matriz C&E: a Norma Técnica N-1883 e a diretriz ET-3000.00-1200-800-PGT-006. Conforme a norma técnica N-1883 da Petrobras, a matriz C&E é, entre outros, um dos documentos fundamentais no projeto de instrumentação e automação. A norma prevê, sobretudo, regras de quais informações devem estar contidas na matriz. Já a diretriz estabelece critérios mínimos a serem considerados na elaboração de matrizes de causa e efeito, como formatações e nomenclaturas a serem utilizadas.

Atualmente o principal problema encontrado no processo de desenvolvimento de programas de CLP de SIS na Petrobras reside na etapa de elaboração da matriz C&E, visto que, mesmo havendo a diretriz para a elaboração da matriz, não se segue nenhum padrão específico de formatação das informações inseridas no documento. Com isso, cada responsável pela edição das matrizes preenche os dados de acordo com a sua preferência, sem submeter à utilização das regras da diretriz.

A principal causa deste problema trata-se da ferramenta utilizada para a criação da matriz. Utiliza-se atualmente um editor de planilhas eletrônicas, que, apesar de ser uma ferramenta altamente flexível, não atende totalmente as necessidades presentes, uma vez que não possui praticidade para a inserção de dados e é bastante susceptível a erros de preenchimento devido a sua facilidade de edição.

A falta de padronização da matriz C&E, além de comprometer a confiabilidade na elaboração de programas, prejudica também o processo de validação dos programas dos SIS, uma vez que o responsável pelos testes manuais deve interpretar as diferentes formatações das matrizes, o que torna o processo mais demorado e passível de erros.

Além destes problemas encontrados, há também a necessidade de interface com as metodologias como apontado na Figura 5.

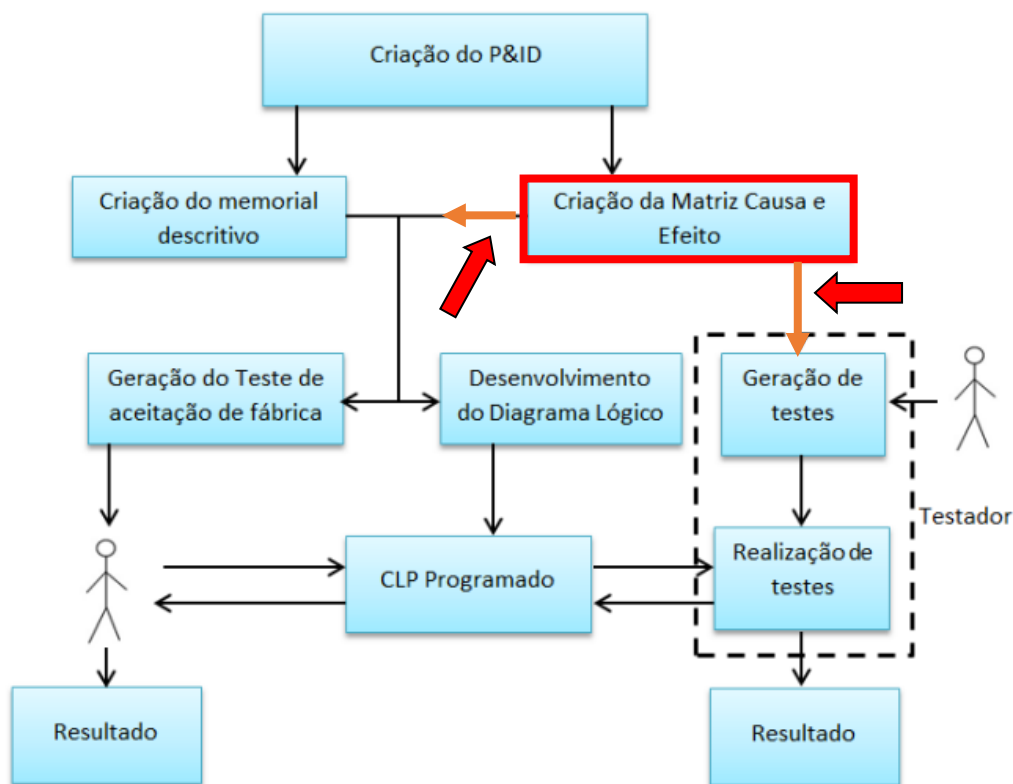


Figura 5: Interface da matriz nas metodologias

Atualmente a interface para a atual metodologia de validação manual da Petrobrás consiste no documento impresso da matriz a partir de sua criação em planilhas eletrônicas. Já a interface para a metodologia de teste automático do

projeto é feita a partir da transcrição das informações contidas na matriz em um documento de texto, que é uma tarefa árdua e penosa. A Figura 6 mostra a visão geral do problema.

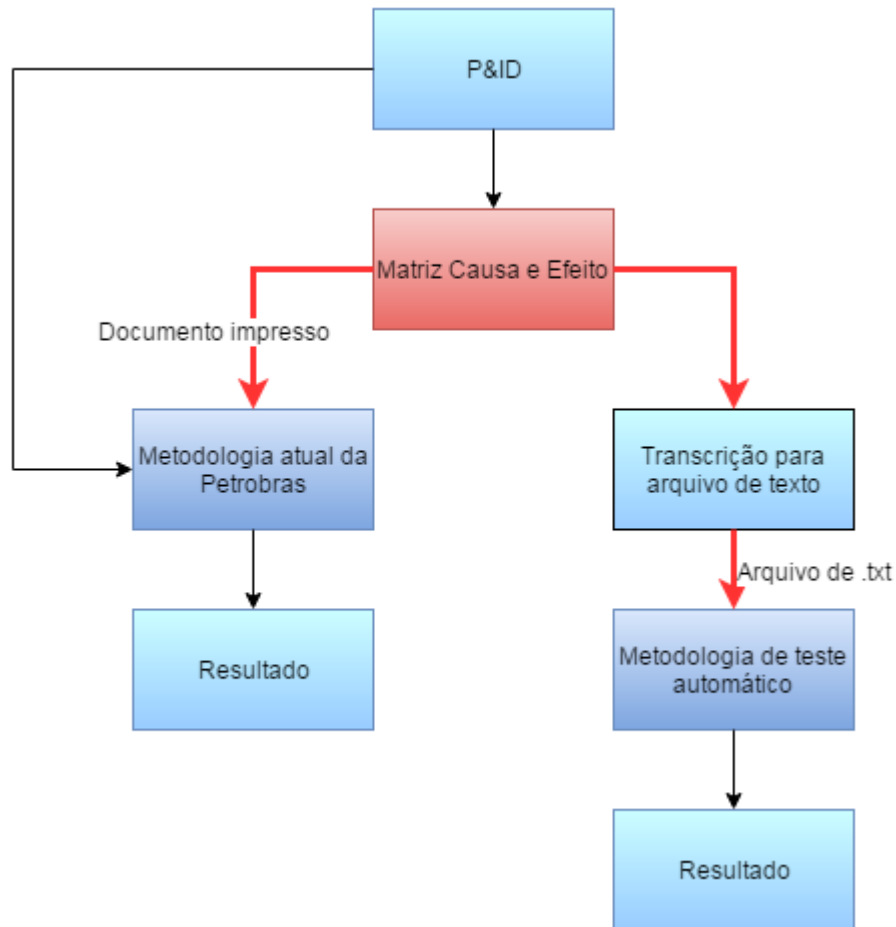


Figura 6: Esquema da interface atual da matriz

Portanto, verifica-se a necessidade de um método ágil e descomplicado de elaborar a matriz C&E em um formato padrão e estabelecer uma interface com as outras etapas das metodologias.

2.4: Proposta de ferramenta para edição de matriz C&E

Tendo em vista a atual sistemática e os problemas encontrados na elaboração da matriz C&E, será apresentado neste trabalho o desenvolvimento de uma ferramenta com uma interface gráfica para a edição de matriz e com a geração documentos como mostrado na Figura 7.

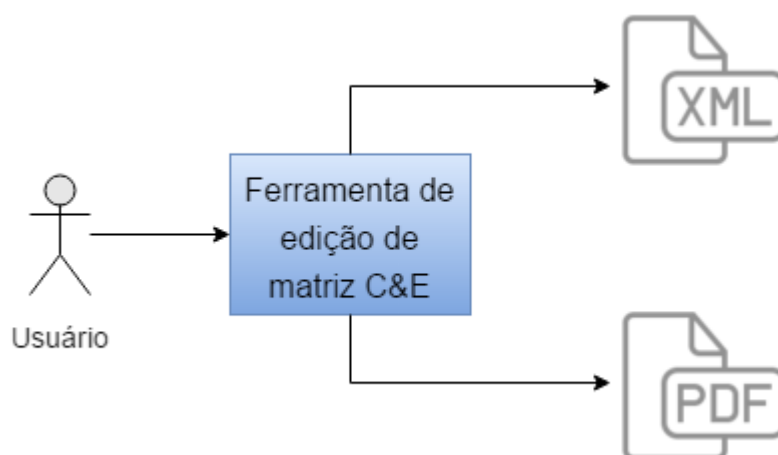


Figura 7: Visão geral da ferramenta

O objetivo desta ferramenta consiste em possibilitar o usuário criar matriz C&E através de uma interface gráfica simples e clara para facilitar a edição para evitar erros e também que faça a inserção das informações de acordo com a diretriz, a fim de se obter uma melhor padronização do documento em relação ao que ocorre atualmente. A ferramenta também tem como propósito gerar um arquivo PDF para que seja feita a documentação e impressão da matriz sem a utilização de planilha, e um arquivo XML para a interface com a ferramenta de teste desenvolvida no projeto dispensando a necessidade de se transcrever manualmente as informações da matriz em um documento de texto.

2.5: Comentários Finais

Neste capítulo foi realizada uma breve descrição da metodologia atual de desenvolvimento de programas de CLP na Petrobras e a apresentação dos documentos utilizados durante o processo. Em seguida foi apresentada a nova metodologia proposta por (Prati 2014) e utilizada atualmente no projeto. Posteriormente foi realizada a definição do problema da matriz C&E encontrado nestas metodologias e, finalmente, apresenta-se a proposta de criação de uma ferramenta para a solução dos problemas encontrados. No próximo capítulo serão apresentados os fundamentos da matriz C&E juntamente com a padrão para a sua elaboração na Petrobras.

Capítulo 3: Padrão para especificação de Matriz Causa e Efeito

Neste capítulo, serão abordados os fundamentos da matriz C&E explanando-se informações presentes nela e como é feita a sua utilização. Em seguida é apresentada o atual padrão aplicado para a criação da matriz mostrando-se as regras definidas na diretriz de elaboração da matriz e que serão utilizadas para o desenvolvimento da ferramenta deste trabalho.

3.1: Fundamentos da Matriz Causa e Efeito

A matriz C&E é um recurso que pode ser utilizado em diversos tipos de aplicação além de especificações de intertravamentos de sistemas de segurança. É usualmente utilizada em análises estatísticas de processos e produtos e podendo ser utilizada para a tomada de decisões estratégicas.

As matrizes C&E possuem geralmente o mesmo arranjo de suas informações, apesar das diferentes aplicações, como mostra a Figura 8.

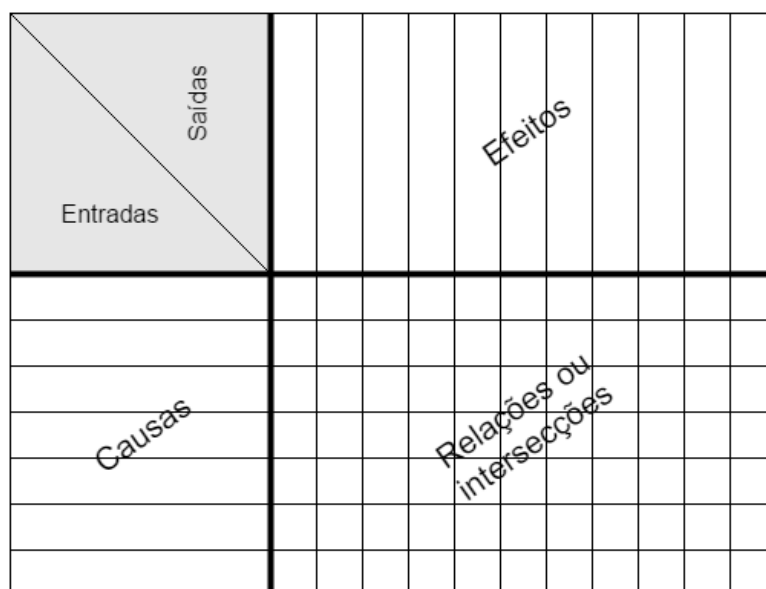


Figura 8: Esquema da matriz C&E

A matriz C&E possui entradas chamadas de causas e saídas, que são chamadas de efeito. O cruzamento dessas entradas e saídas correlacionam as causas aos efeitos e são denominadas relações ou intersecções, podendo conter valores lógicos ou numéricos dependendo da aplicação.

No caso de sistemas de segurança, a matriz C&E é utilizada para definir como e quando as ações são executadas no sistema. Este documento envolve a organização de eventos de processo em categorias de causas – que são os sensores, e efeitos – que são os acionamentos, e, em seguida, ligando-se elas. As ligações entre as causas e efeitos, chamadas de relação ou intersecção, indica os efeitos que resultarão de uma causa ativa. A partir destes dados, a lógica pode ser extraída para criar um programa para a execução de um sistema de respostas para conter e evitar eventos antes de causarem danos a um processo.

3.2: Padrão de elaboração de Matriz de Causa e Efeito na Petrobras

A principal referência para a elaboração da matriz C&E na Petrobras é a diretriz ET-3000.00-1200-800-PGT-006, que apresenta o modelo da matriz a ser utilizada assim como simbologias, nomenclaturas e regras gerais para a determinação de um padrão de formatação da matriz.

A Figura 9 mostra o modelo da matriz utilizado na Petrobrás conforme a diretriz.

SYMBOLS				EFFECT	TAG NUMBER	EFFECT	DESCRIPTION	NOTES
FIELD	EQUIP. OR SERVICE OR ZONE	NOTING	TAG NUMBER					
<input checked="" type="checkbox"/>	"OR" LOGIC			61	62	63	64	65
<input checked="" type="checkbox"/>	"AND" LOGIC (EXPOSED OR)			66	67	68	69	70
<input type="checkbox"/>	"AND" LOGIC			71	72	73	74	75
<input type="checkbox"/>	"OR" LOGIC TIME DELAY			76	77	78	79	80
<input type="checkbox"/>	VERTICAL FUNCTION FROM TRANSACTOR (USE ON "H" FIELD CAUSE COLUMN)			81	82	83	84	85
<input type="checkbox"/>	VERTICAL FUNCTION (VERTICAL CROSSING) (USE ON "H" FIELD EFFECT COLUMN)			86	87	88	89	90
				91	92	93	94	95
				96	97	98	99	100
				101	102	103	104	105
				106	107	108	109	110
				111	112	113	114	115
				116	117	118	119	120
				121	122	123	124	125
				126	127	128	129	130
				131	132	133	134	135
				136	137	138	139	140
				141	142	143	144	145
				146	147	148	149	150
				151	152	153	154	155
				156	157	158	159	160
				161	162	163	164	165
				166	167	168	169	170
				171	172	173	174	175
				176	177	178	179	180
				181	182	183	184	185
				186	187	188	189	190
				191	192	193	194	195
				196	197	198	199	200
				201	202	203	204	205
				206	207	208	209	210
				211	212	213	214	215
				216	217	218	219	220
				221	222	223	224	225
				226	227	228	229	230
				231	232	233	234	235
				236	237	238	239	240
				241	242	243	244	245
				246	247	248	249	250
				251	252	253	254	255
				256	257	258	259	260
				261	262	263	264	265
				266	267	268	269	270
				271	272	273	274	275
				276	277	278	279	280
				281	282	283	284	285
				286	287	288	289	290
				291	292	293	294	295
				296	297	298	299	300

SYMBOLS			
FIELD	EQUIP. OR SERVICE OR ZONE	NOTING	TAG NUMBER

SYMBOLS			
<input checked="" type="checkbox"/>	"OR" LOGIC		
<input checked="" type="checkbox"/>	"AND" LOGIC (EXPOSED OR)		
<input type="checkbox"/>	"AND" LOGIC		
<input type="checkbox"/>	"OR" LOGIC TIME DELAY		
<input type="checkbox"/>	VERTICAL FUNCTION FROM TRANSACTOR (USE ON "H" FIELD CAUSE COLUMN)		
<input type="checkbox"/>	VERTICAL FUNCTION (VERTICAL CROSSING) (USE ON "H" FIELD EFFECT COLUMN)		


REFERENCE DOCUMENTS			
1-REV-3000.00-1200-800-800-004 (EMERGENCY GUIDELINES FOR THE CORRECTION OF CAUSE AND EFFECT MATRICES AND LOGIC DIAGRAMS)			
GENERAL NOTES			
REV.	DESCRIPTION	DATE	BY
THIS DOCUMENT IS PROPERTY OF PETROBRAS AND IS PROTECTED BY APPLICABLE IPR. PREVALING LAW IF SMALL. ONLY BE USED FOR THE PURPOSE IT IS DESIGNED.			
 PETROBRAS			
PROJECT TITLE			
JOB			
DRAWING TITLE			
SUBSYSTEM			
DRAWN BY	CHECKED BY	DESIGNED BY	REVIEWED BY
SCALE			
DATE	DRAWING NO.		
	DE-3010-__1200-800-__		

Figura 9: Matriz C&E utilizada na Petrobras [3]

Para uma melhor ilustração, a matriz pode ser dividida em 8 áreas distintas onde as informações estão contidas conforme mostra a Figura 10. Entre elas estão as áreas de: Legendas, Causas, Efeitos, Relações/Intersecções, Notas, Documentos referenciados, Notas gerais e Rodapé.

Legendas	Efeitos	X	Documentos referenciados
			Notas gerais
Causas	Relações	Notas	Rodapé

Figura 10: Informações da matriz divididas em áreas

Cada área é descrita a seguir:

- **Legendas:** Nesta área estão presentes a legenda das simbologias utilizadas para o preenchimento das relações da matriz.
- **Causas:** Contém os dados e identificadores de cada um dos sensores de campo;
- **Efeitos:** Abrange os dados e identificadores dos acionamentos (atuadores).
- **Relações (intersecções):** Contém a lógica que relaciona cada causa ao seu(s) efeito(s) associados;
- **Notas:** Referencia cada causa ou intersecção a uma nota presente na área “Notas gerais”;
- **Documentos referenciados:** Contém lista de documentos utilizados para a elaboração da matriz;
- **Notas gerais:** Possui a descrição detalhada de cada nota inserida na matriz;
- **Rodapé:** Contém informações que identificam a matriz, como a data em que foi criada, responsável pela elaboração do documento, identificador da matriz, o sub-sistema o qual a matriz pertence, entre outros.

A diretriz define também algumas regras de formatação dos dados a serem inseridos como simbologias das lógicas, nomenclaturas das “tags” e define também como os dados devem estar dispostos na matriz. Entre as principais regras determinadas pela diretriz estão:

- A matriz C&E terá 50 linhas x 50 colunas no formato A2.
- Cada matriz deverá ter no campo “Documentos de Referência” os números e títulos entre parênteses das matrizes dos demais sub-sistemas.
- A simbologias das lógicas a serem utilizadas nas intersecções matrizes serão apenas “OU”, “E” ou “OU temporizado” conforme a Tabela 1.

FUNÇÃO	SÍMBOLO	OBSERVAÇÃO
Lógica OU		
Lógica NOR		OU negado
Lógica E	A Ou Ax	X é numero inteiro a ser usado quando necessário para distinguir conjuntos de lógicas E entre si quando os mesmos estão em uma única coluna.
Lógica Temporizada	Tx	X é o tempo em segundos escolhido para a lógica.

Tabela 1: Simbologia das lógicas

Outras lógicas ou combinações deverão ser descritas por notas laterais, diagramas lógicos e lógicas booleanas. No modelo da matriz utilizada, existe o campo “NOTAS”. Não sendo suficiente, deve-se usar o campo “NOTAS GERAIS”.

- Os sensores de fogo, gás e fumaça deverão ter o seu “tag” de acordo com o seguinte formato:

SSS-ZZZLXX

Onde:

SSS = sensor, baseado no API-14C e ISA-S5.1, conforme Tabela 2;

Sensor	Descrição
AST	Sensor de gás hidrogênio
AST	Sensor de gás hidrocarboneto
OST	Sensor de gás sulfídrico
YST	Sensor de fumaça
HSS	Botoeira de alarme manual de incêndio
UST	Sensor de chama
TST	Sensor de calor

Tabela 2: Padrão de “tag” a ser utilizado

ZZZ = zona onde está localizado o sensor;

L = malha onde está o sensor (0 se não-endereçável);

XX = sequencial dentro da zona.

- Os grupos de sensores (envolvidos em votação) deverão ser separados por linhas de maior peso (peso 1) conforme o exemplo da Tabela 3:

	votação	Fogo det Z-101	Fogo conf Z-101	Gás det Z-101	Gás conf Z-101	ESD-3
Z-101						
GHC-101001	1oo3			X		
GHC-101002	2oo3				X	X
GHC-101003						
PSL-542001			X			X
FLM-101001	1oo2	X				
FLM-101002	2oo2		X			X

Tabela 3: Exemplo de grupo de sensores

3.3: Comentários Finais

Neste capítulo mostrou-se os fundamentos da matriz C&E e a visão geral da diretriz que servirá de base para os requisitos da ferramenta. No próximo capítulo serão apresentadas etapas de projeto da ferramenta proposta utilizando-se conceitos de engenharia de software.

Capítulo 4: Projeto da Ferramenta

Neste capítulo, serão apresentados conceitos de Engenharia de software e metodologias de desenvolvimento de sistemas, a fim de justificar a escolha de uma delas para o desenvolvimento da ferramenta. Será abordado também a etapa de projeto da ferramenta, com a apresentação da análise e definição dos requisitos e do projeto de arquitetura, interface e da estrutura do XML.

4.1: Engenharia de Software/Metodologia de desenvolvimento de sistemas

A fim de se obter melhor organização do desenvolvimento, documentação, manutenibilidade e qualidade da ferramenta, utilizou-se no trabalho fundamentos/conceitos de Engenharia de Software, que, de acordo com a IEEE (Institute of Electrical and Electronics Engineers), é definido da seguinte forma:

A aplicação de uma abordagem sistemática, disciplinada e quantificável para o desenvolvimento, operação e manutenção de software, isto é, a aplicação de engenharia em software. [7]

Como esta definição sugere, o que é produzido não é a única coisa importante, mas também como é produzido. As disciplinas de engenharia empregam um conjunto estabelecido de abordagens sistemáticas, disciplinadas e quantificáveis para o desenvolvimento de artefatos. Ao aplicar cuidadosamente um conjunto análogo de abordagens ao desenvolvimento de software, podemos esperar a produção de software altamente confiável, manutenível e que atende os requisitos especificados. Uma abordagem disciplinada é particularmente importante conforme o tamanho de um projeto de software cresce. Com o aumento de tamanho surge uma complexidade muito maior, e aplicar uma abordagem sistemática e disciplinada é essencial.

A engenharia de software é uma tecnologia em camadas. De acordo com a Figura 11, qualquer abordagem de engenharia, inclusive a de software, deve se apoiar num compromisso organizacional com a qualidade [8].



Figura 11: Engenharia de Software em Camadas [8]

O alicerce da engenharia de software é a camada de processo. O processo de engenharia de software é o adesivo que mantém unidas as camadas de tecnologia e permite o desenvolvimento racional e oportuno de softwares de computador. O processo define um arcabouço que deve ser estabelecido para a efetiva utilização da tecnologia de engenharia de software. Os processos de software formam a base para o controle gerencial de projetos de software e estabelecem o contexto no qual os métodos técnicos são aplicados, os produtos de trabalho (modelos, documentos, dados, relatórios, formulários etc.) são produzidos, os marcos são estabelecidos, a qualidade é assegurada e as modificações são adequadamente geridas.

Existem muitos modelos de processos de softwares diferentes, mas todos devem incluir quatro atividades fundamentais para a engenharia de software: especificação, projeto e implementação, validação e evolução do software [9]. Entre eles, os mais tradicionais são:

1. **Modelo em cascata.** Esse modelo considera as atividades fundamentais do processo de especificação, desenvolvimento, validação e evolução, e representa cada uma delas como fases distintas, como especificação de requisitos, projeto de software, implementação, teste, e assim por diante.
2. **Modelo incremental.** Essa abordagem intercala as atividades de especificação, desenvolvimento e validação. O sistema é desenvolvido com uma série de versões (incrementos), de maneira que cada versão adiciona funcionalidade à anterior.
3. **Engenharia de software orientada a reúso.** Essa abordagem é baseada na existência de um número significativo de componentes reusáveis. O processo de desenvolvimento de sistemas concentra-se na integração

desses componentes em um sistema já existente em vez de desenvolver um sistema a partir do zero.

4.2: Modelo de processo de software adotado

Para o desenvolvimento da ferramenta deste trabalho, adotou-se o modelo cascata de processo de software. O modelo em cascata, algumas vezes chamado de ciclo de vida clássica, sugere uma abordagem sistemática e sequencial para o desenvolvimento de softwares que começa com a especificação dos requisitos pelo cliente e progride ao longo do planejamento, projeto, implementação, testes e culminando na implantação, como visto em [8]. A escolha é justificada em virtude de o modelo cascata ser usado apenas quando os requisitos são bem compreendidos e pouco provavelmente venham a ser radicalmente alterados durante o desenvolvimento do sistema, como dito em [9], e estes são aspectos que se enquadram nas características da ferramenta.

O modelo em cascata é um exemplo de um processo dirigido a planos – em princípio, deve-se planejar e programar todas as atividades do processo antes de começar a trabalhar nelas [9]. Os principais estágios do modelo em cascata refletem diretamente as atividades fundamentais do desenvolvimento:

1. **Análise e definição dos requisitos.** Os serviços, restrições e metas do sistema são estabelecidos por meio de uma consulta aos usuários. Em seguida são definidos em detalhes e funcionam como uma especificação do sistema.
2. **Projeto de sistema e software.** O processo de projeto de sistemas aloca os requisitos tanto para sistemas de hardware como para sistemas de software, por meio da definição de uma arquitetura geral do sistema. O projeto de software envolve a identificação e descrição das abstrações fundamentais do sistema de software e seus relacionamentos.
3. **Implementação.** Durante este estágio, o projeto do software é desenvolvido como um conjunto de programas ou unidades de programa.

4. **Integração e Testes de sistema.** As unidades individuais do programa são integradas e testadas como um sistema completo para assegurar que os requisitos do software tenham sido atendidos.
5. **Implantação e manutenção.** Nesta fase o sistema é instalado e colocado em uso. A manutenção envolve a correção de erros que não foram descobertos em estágios iniciais do ciclo de vida, com melhoria da implementação das unidades do sistema e ampliação de seus serviços em resposta às descobertas de novos requisitos.

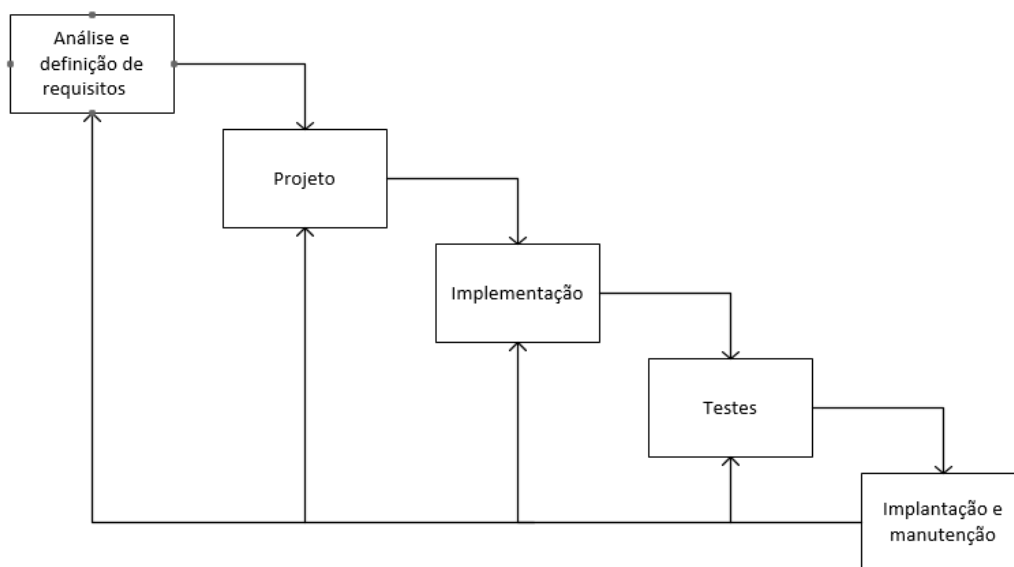


Figura 12: Etapas do modelo cascata

Com isso o desenvolvimento deste trabalho obedece esta sequência de atividades estabelecidos pelo modelo cascata.

4.3: Análise e definição dos requisitos

Os requisitos de um sistema são as descrições do que o sistema deve fazer, os serviços que oferece e as restrições a seu funcionamento. Esses requisitos refletem as necessidades dos clientes para um sistema que serve a uma finalidade determinada, como controlar um dispositivo, colocar um pedido ou encontrar informações [8].

Os requisitos de software são frequentemente classificados com requisitos funcionais e requisitos não funcionais:

1. **Requisitos funcionais:** São declarações de serviços que o sistema deve fornecer, de como o sistema deve reagir a entradas específicas e de como o sistema deve se comportar em determinadas situações. Em alguns casos, os requisitos funcionais também podem explicitar o que sistema não deve fazer [8].
2. **Requisitos não funcionais:** São restrições aos serviços ou funções oferecidos pelo sistema. Incluem restrições de *timing*, restrições no processo de desenvolvimento e restrições impostas pelas normas. Ao contrário das características individuais ou serviços do sistema, os requisitos não funcionais, como a necessidade de incluir recursos de autenticação de usuário no sistema [8].

4.3.1: Requisitos Funcionais

As especificações dos requisitos funcionais da ferramenta foram levantadas a partir das regras da norma técnica, da diretriz e das necessidades do projeto.

ID	Requisito	Descrição	Entrada	Saída
[RF01]	Criar nova matriz causa e efeito	A ferramenta deve ter a função de criar uma nova matriz	-	Uma nova matriz vazia é criada na interface
[RF02]	Editar/Abrir matriz causa e efeito	A ferramenta deve ser capaz de editar uma matriz existente	Escolha de uma matriz já existente em XML	Os dados da matriz salva escolhida são mostrados na interface
[RF03]	Visualização da matriz causa e efeito	A ferramenta deve possuir a visualização da matriz durante a edição	-	Interface que permita a visualização da matriz e atualização dos dados conforme a interação com o usuário
[RF04]	Inserir identificadores da matriz (rodapé)	A ferramenta deve possuir janelas de entrada de dados para as informações do rodapé	Campos de entrada para todos os identificadores da matriz.	Informações do rodapé inseridas na matriz
[RF05]	Inserir dados da causa	A ferramenta deve possuir janelas de entrada de	Campos de entradas para tipo de sinal,	Inserção dos dados causas inseridas na

		dados de causas	P&ID, equipamento, votação e tag.	matriz
[RF06]	Inserir dados dos efeitos	A ferramenta deve possuir janelas de entrada de dados de efeitos	Campos de entrada para descrição, tag, tipo de sinal, P&ID e interface.	Inserção dos dados dos efeitos na matriz
[RF07]	Inserir lógica das intersecções	A ferramenta deve possuir janelas de entrada de dados de intersecções	Escolha de uma das lógicas disponíveis na matriz	Inserção da lógica na matriz
[RF08]	Inserir notas	A ferramenta deve possuir janelas de entrada de dados para notas	Campo de entrada para a inserção da nota	Nota inserida na matriz
[RF09]	Inserir documentos referenciados	A ferramenta deve possuir janelas de entrada de dados para documentos referenciados	Campo de entrada para a inserção do nome do documento	Inserção do nome do documento na matriz
[RF10]	Excluir dados	A ferramenta deve ser capaz de realizara exclusão de dados	Escolha dos dados a serem excluídos	Exclusão dos dados selecionados da matriz
[RF11]	Gerar PDF da matriz	A ferramenta deve ser capaz de gerar um documento PDF da matriz C&E	Matriz de causa e efeito presente na interface	Criação de um documento PDF da matriz e com as informações contidas nela.
[RF12]	Gera XML da matriz	A ferramenta deve ser capaz de gerar um documento XML da matriz C&E	Matriz de causa e efeito presente na interface	Criação de um documento XML da matriz e com as informações contidas nela.
[RF13]	Escolher o local para salvar a matriz	A ferramenta deve possuir a opção de o usuário escolher o local de armazenamento dos documentos gerados	Escolha de uma pasta para armazenar os documentos gerados	Definição de pasta/local para salvar os documentos
[RF14]	Agrupar causas	A ferramenta deve possuir a funcionalidade de agrupar sensores de uma mesma zona	Seleção das causas a serem agrupadas	Agrupamento das causas selecionadas separando-as com uma linha de peso maior

Tabela 4: Requisitos Funcionais

4.3.2: Requisitos Não Funcionais

Já os requisitos não-Funcionais especificados têm como objetivo criar restrições de desempenho e usabilidade como mostra a Tabela 5.

ID	Descrição
[RNF01]	A matriz deve possuir o mesmo Layout definido pela Petrobras
[RNF02]	O software deve ser implementado na linguagem Python
[RNF03]	O software deve possuir uma interface com navegação fácil e intuitiva
[RNF04]	O tempo para a geração de um arquivo XML ou PDF da matriz causa e efeito não deve ultrapassar 5 segundos
[RNF05]	O tempo de transição entre janelas da interface não deve ultrapassar 2 segundos
[RNF06]	A ferramenta deve operar em ambiente Windows XP ou superior
[RNF07]	A ferramenta deve fazer log de falhas caso ocorram erros durante a execução do software

Tabela 5: Requisitos Não-Funcionais

4.4: Projeto/modelagem

Um projeto de software é uma descrição da estrutura do software a ser implementado, dos modelos e estruturas de dados usados pelo sistema, das interfaces entre componentes do sistema e, às vezes, dos algoritmos usados [8].

Para o projeto da ferramenta, por se tratar de um software bastante inteligível, julgou-se suficiente a concepção de um diagrama de classes, um diagrama de navegação e o diagrama de estrutura do documento XML a ser gerado pela ferramenta.

4.4.1: Diagrama de classes

O objetivo do Diagrama consiste em identificar e descrever vários tipos de objetos no sistema e o relacionamento entre eles.

Existem três perspectivas que podem ser utilizadas para projetar diagramas de classes:

- **Conceitual:** Se tomarmos a perspectiva conceitual, projeta-se um diagrama que representa os conceitos do domínio que está sendo estudado. Estes

conceitos serão naturalmente relacionados às classes que irão executá-los. Na verdade, um modelo conceitual deve ser projetado com pouca ou nenhuma preocupação com o software que poderá implementá-lo. Portanto, deve ser considerado independente da linguagem implementada. Esta perspectiva recebe o nome de perspectiva essencial.

- **Especificação:** Agora estamos examinando o software, mas estamos analisando as suas interfaces, não a sua implementação. O desenvolvimento orientado a objetos dispõe muita ênfase na diferença entre interface e implementação, mas isso é frequentemente negligenciado na prática porque a noção que temos de classe em uma linguagem orientada a objetos combina interface com implementação.
- **Implementação:** Nesta visão, realmente temos classes e estamos pondo a implementação às claras. Esta é, provavelmente, a perspectiva usada com mais frequência.

Devido à simplicidade e compreensibilidade da ferramenta, optou-se pela criação do diagrama de classes com perspectiva conceitual para ilustrar somente os objetos do sistema e suas relações com intuito de auxiliar na etapa de implementação/codificação como mostra a Figura 13.

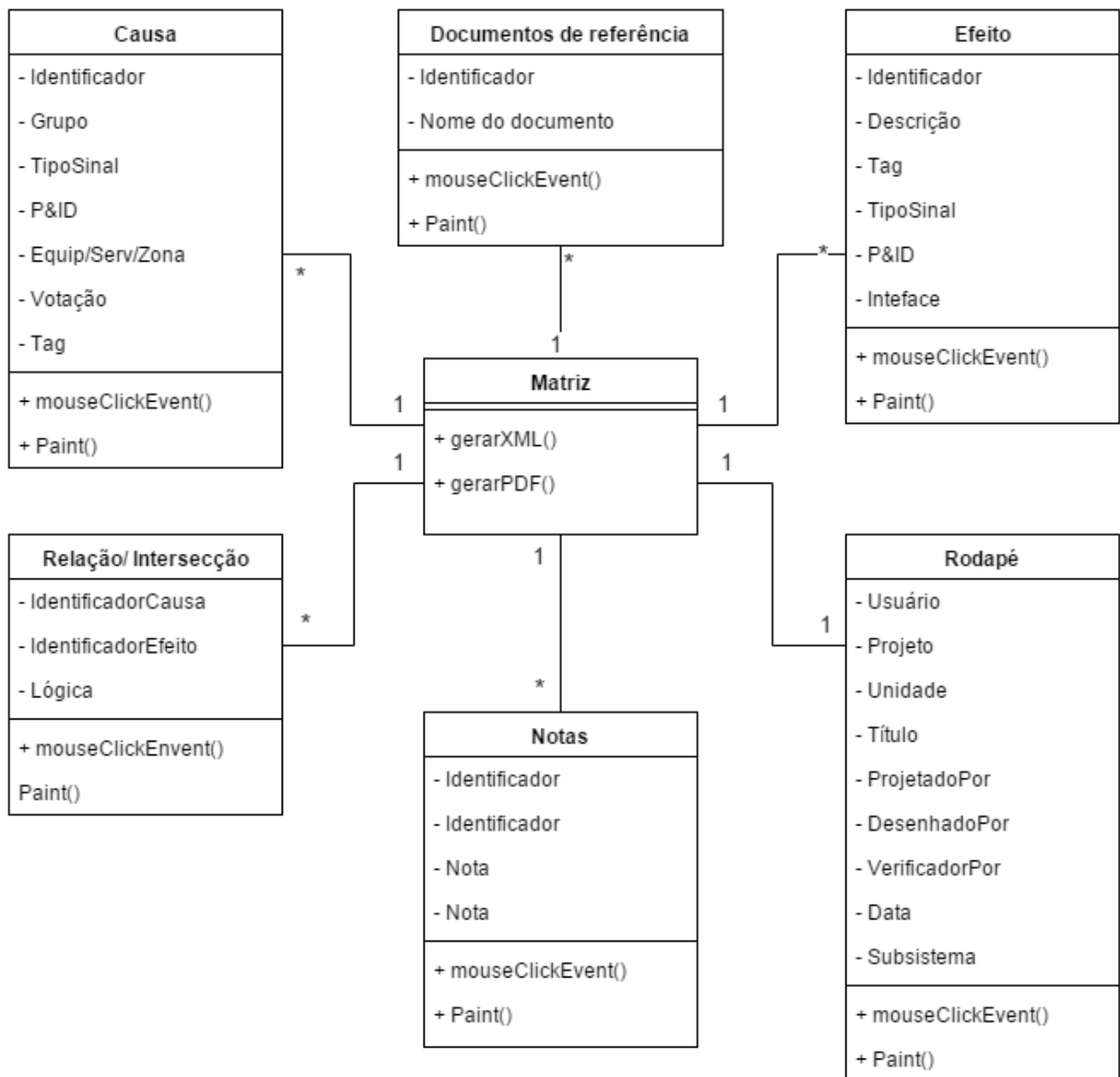


Figura 13: Diagrama de classes

No diagrama de classes, cada bloco representa uma classe, na qual estão contidos os atributos – indicados com o símbolo “ – “ e métodos – indicados com o símbolo “+”. Já a conexão entre esses blocos representa o relacionamento entre as classes.

O diagrama criado é composto por 7 classes, sendo que a classe “matriz” é a principal pois está associado a todas as outras 6 classes. Ela possui 2 métodos

- gerarXML(): Método que realiza a função de geração de arquivo XML
- gerarPDF(): Método que realiza a função de gerar arquivo PDF

As outras 6 classes também possuem 2 métodos que são:

- `mouseClickEvent()`: Esta função ocorre quando o usuário realiza o clique em um dos objetos da interface gráfica e que por sua vez abre-se a respectiva janela de entrada de dados.
- `Paint()`: Método que faz o desenho de cada objeto na interface gráfica.

Os atributos de cada classe estão relacionados a seus respectivos dados.

4.4.2: Projeto interface gráfica

O projeto de interface tem o intuito de determinar quais são as janelas da ferramenta e as possibilidades de navegação entre uma janela e outra. Para tanto, elaborou-se o diagrama de estados de navegação que indica quais são as janelas que compõe o sistema e quais eventos permitem ao usuário navegar pela interface como mostra a Figura 14.

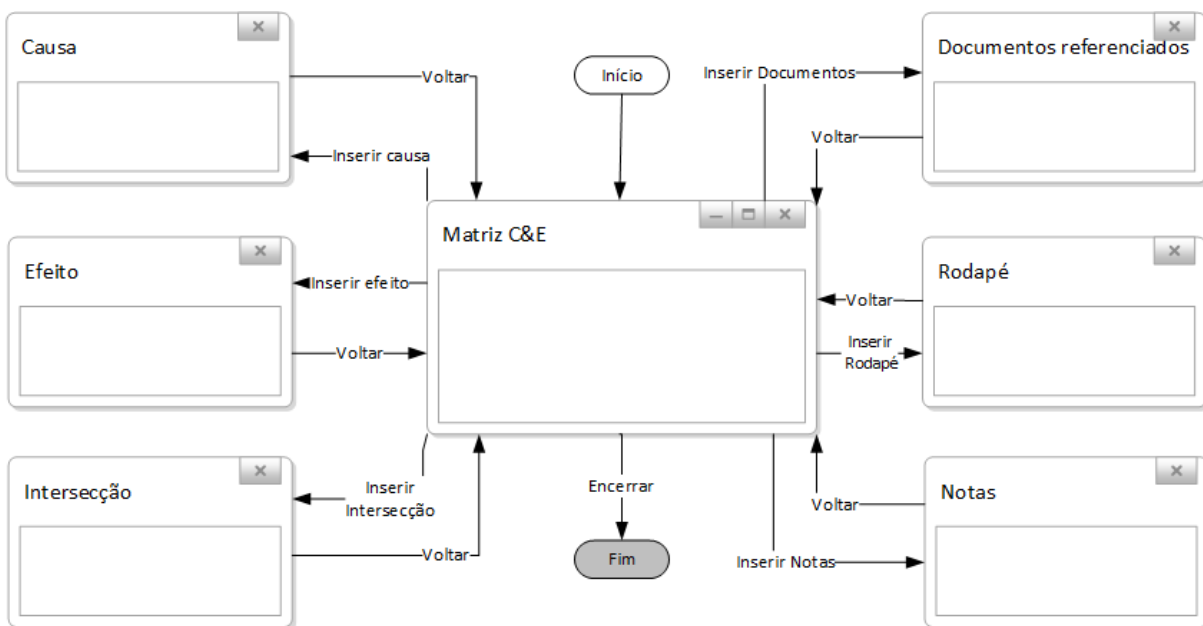


Figura 14: Diagrama de navegação

Conforme o diagrama, o projeto da interface é composto por uma janela principal e seis janelas secundárias. A janela principal contém a matriz C&E e a partir de eventos gerados pela ação do usuário as outras janelas são mostradas ou fechadas.

Ao iniciar a interface, a janela principal com a matriz é mostrada e o usuário poderá optar por realizar uma dessas ações:

- Clicar em fechar: Quando se clica no botão de fechar a interface é encerrada.
- Clicar em uma das causas: Ao clicar em uma das linhas das causas, aparecerá ao usuário a janela para a inserção dos dados da causa que foi pressionada. Com isso o usuário tem a opção de fechar esta janela ou realizar o preenchimento e encerrar. Ambas as ações farão o fechamento da janela e o retorno à janela principal.
- Clicar em um dos efeitos: Com o clique em um dos efeitos, abre-se a janela para a introdução dos dados do atuador/acionamento. Nesta janela o usuário poderá realizar o preenchimento dos dados e finalizar ou fechar a janela diretamente. Qualquer um desses eventos fará o retorno para a janela principal.
- Clicar em uma das intersecções: Ao clicar em uma das intersecções da matriz, abre-se a janela para a escolha da lógica associada à ela. Nela, o usuário poderá fazer a escolha de uma das opções de lógica e encerrar ou fechar a janela sem realizar nenhuma ação. Em qualquer um dos casos haverá o retorno para a janela principal.
- Clicar em documentos referenciados: Clicando-se no campo de documentos referenciados, abre-se a janela de entrada para nome de documentos referenciados na matriz. O usuário poderá fechar a janela sem inserir nenhum dado ou adicionar/remover um documento.
- Clicar no rodapé: Ao clicar no rodapé, aparecerá ao usuário uma janela para a inserção dos dados que identificam a matriz. Assim como nas outras janelas, pode-se fechar a janela diretamente ou fazer o preenchimento dos dados e encerrar para o retorno à janela principal.
- Clicar em notas: Com o clique em notas abre-se a janela para inserção de notas. Pode-se adicionar uma nova nota ou remover nota existente.

4.4.3: Estruturação do XML

XML, do inglês eXtensible Markup Language, é uma linguagem de marcação recomendada pela W3C para a criação de documentos com dados organizados hierarquicamente, tais como textos, banco de dados ou desenhos vetoriais. A

linguagem XML é classificada como extensível porque permite definir os elementos de marcação. Alguns dos propósitos do XML são: auxiliar os sistemas de informação no compartilhamento de dados, codificar documentos e inserir seriais nos dados comparando o texto com o de outras linguagens baseadas em serialização [10]. (W3C, ou World Wide Web Consortium, é um consórcio de empresas de tecnologia que visa padronizar a criação e interpretação de conteúdos para websites. Foi fundada em 1994 por Tim Berners-Lee, o criador da internet, para extrair o máximo que a rede pode oferecer)

Documentos XML é formado por uma estrutura tipo árvore, que começa com um elemento “raiz” e os ramos da raiz para elementos “filho” como mostra o exemplo da Figura 15. Todos os elementos podem ter sub-elementos (elemento filho) ou atributos ou pode possuir “Text”, que é o dado do tipo texto associado a ele.

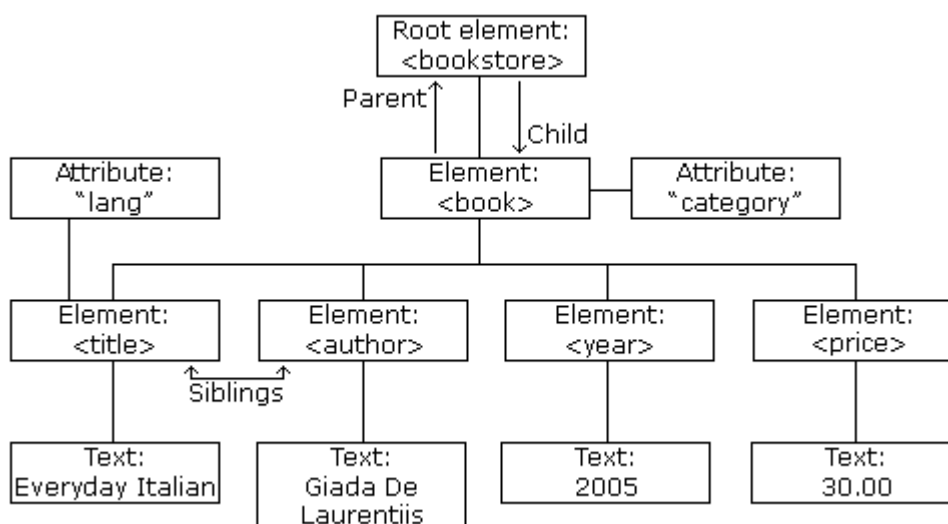


Figura 15: Estrutura de documentos XML [10]

Desta forma, para a estruturação das informações da matriz C&E no arquivo XML, dividiu-se o documento da seguinte em 6 áreas distintas como mostra a Figura 15:

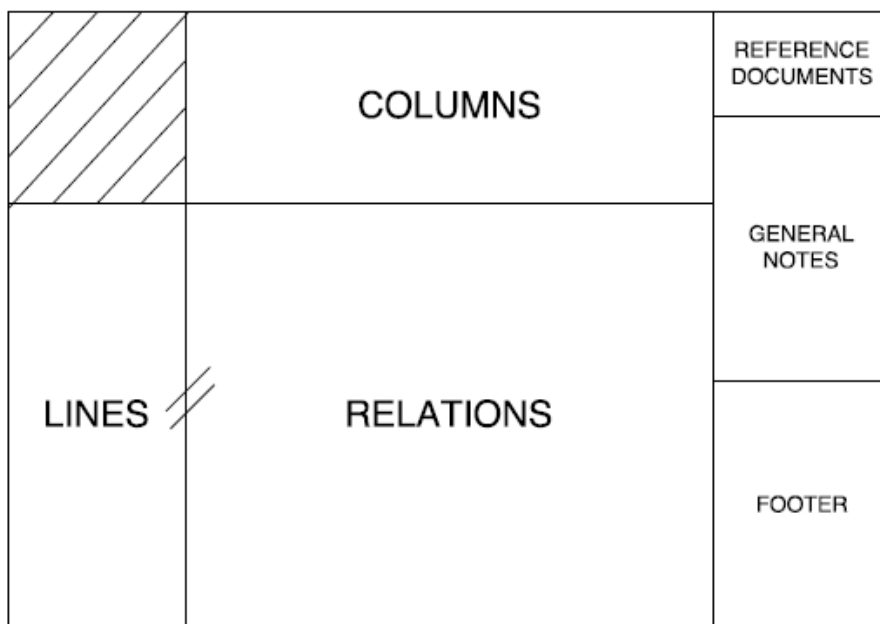


Figura 16: Divisão da matriz para a estruturação do XML

Com isso, estruturou-se a árvore conforme mostrado na Figura 17:

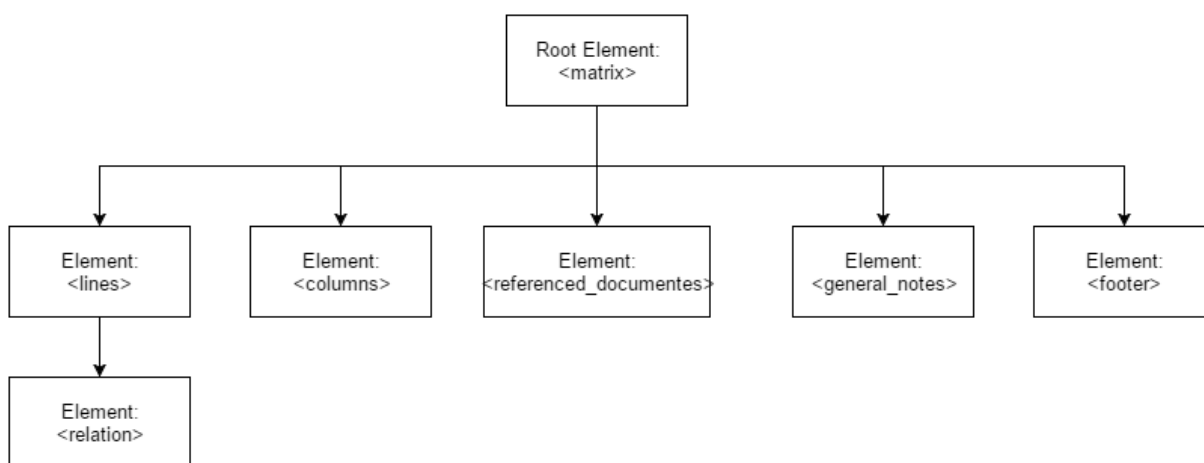


Figura 17: Árvore simplificada da estrutura do XML (Árvore completa no Apêndice A)

O elemento <matrix> trata-se da matriz como um todo e que possui ramos com os seguintes elementos filho:

- <lines>: Possui os dados de cada causa da matriz e possui o elemento filho <relation>, que por sua vez compreende a lógica que relaciona cada causa com um efeito;

- <columns>: Detêm informações dos efeitos;
- <referenced_documents>: Contém os nomes dos documentos de referência;
- <general_notes>: Possui as informações de cada nota inserida na matriz;
- <footer>: Compreende os dados do rodapé da matriz, com os dados que identificam o documento.

4.5: Comentários Finais

Neste capítulo foi tratado o projeto da ferramenta proposta, com a introdução de conceitos de engenharia de software e apresentação do modelo de processo de software utilizado. Posteriormente, fez-se o levantamento e especificação dos requisitos funcionais e não funcionais da ferramenta seguido da etapa do projeto de estrutura, com a elaboração do diagrama de classes, diagrama de navegação e o projeto de estrutura do arquivo XML a ser gerado. O próximo capítulo tratará do estágio de desenvolvimento da ferramenta apresentando-se as etapas de implementação, testes e implantação.

Capítulo 5: Desenvolvimento da Ferramenta

No capítulo anterior, foi mostrada a etapa de projeto da ferramenta, na qual define-se os requisitos e realiza-se o projeto com a elaboração de diagramas e estrutura de dados. Neste capítulo, será abordado a implementação da ferramenta a partir dos requisitos e do projeto realizado, no qual serão apresentados alguns recursos de programação utilizados assim com a alguns detalhes da codificação. Em seguida será apresentada a etapa de testes para a verificação do correto funcionamento da ferramenta e finalmente será apresentada a etapa de implantação.

5.1: Implementação

O estágio de implementação do desenvolvimento de software é o processo de conversão de uma especificação do sistema executável e sempre envolve processos de projeto e programação de software [8].

Para a codificação da ferramenta utilizou-se a linguagem de programação Python. Python é uma linguagem de programação de alto nível, interpretada, de script, imperativa, orientada a objetos, funcional, de tipagem dinâmica e forte. Python suporta vários paradigmas de programação, incluindo orientada a objeto, imperativo e programação funcional ou processuais estilos. Possui um tipo dinâmico do sistema e automático gerenciamento de memória e tem um grande e abrangente biblioteca padrão [11].

Para a implementação da GUI, foi utilizada a biblioteca PyQt. O PyQt é uma biblioteca que cria uma interface entre o Python e a biblioteca Qt, que por sua vez é um framework multiplataforma para desenvolvimento de interfaces gráficas em C++ criado pela empresa norueguesa Trolltech. PyQt implementa cerca de 440 classes e mais de 6.000 funções e métodos [12].

Já para a geração de arquivos PDF utilizou-se a biblioteca ReportLab, que é uma coleção de módulos de linguagem Python open source organizados e apoiados por ReportLab. Os módulos se preocupam principalmente com a automatização de métodos para gerar arquivos Portable Document Format (PDF), mas há também um

multi-plataforma biblioteca de gráficos vetoriais capaz de gerar bitmap e vetoriais gráficos e desenhos [13].

Para acessar e manipular arquivos XML, utilizou-se a biblioteca Minidom que é a implementação mínima da interface DOM – Document Object Model – e define uma maneira padrão para acessar e manipular documentos XML. DOM é uma representação de um documento XML em uma estrutura em árvore carregada na memória. Portanto nas definições das interfaces DOM descrevem os meios de acessarmos está estrutura em árvore [14].

Com a utilização destas bibliotecas foi possível realizar a implementação das funcionalidades requeridas para a ferramenta.

5.2: Testes

O teste é destinado a mostrar que um programa faz o que é proposto a fazer e para descobrir os defeitos do programa antes do uso. Quando se testa o software, o programa é executado usando dados fictícios. Os resultados do teste são verificados à procura de erros, anomalias ou informações sobre os atributos não funcionais do programa [9].

Existem diversas técnicas e estratégias que podem ser usadas para teste de software, conforme [8]. Porém, neste trabalho, realizou-se somente simples testes de interface com o usuário, uma vez que haverá futuras modificações e acréscimo de funcionalidades na ferramenta e também pelo fato de não ser um produto comercial e, portanto, não observou-se necessidade de realizar testes mais sofisticados. Portanto, examinaram somente aspectos fundamentais do sistema, pouco se preocupando com a lógica com a estrutura lógica do software.

Com isso, foram realizados diversos testes seguidamente, com o preenchimento de diversos dados na matriz C&E utilizando todas as possibilidades da ferramenta e seguido da geração de arquivo XML e PDF. Contudo, nenhum problema ou falha de execução foi encontrada.

Outro teste realizado foi a verificação de ocupação de memória do sistema. A ocupação de memória da ferramenta manteve-se constante em torno de 24 megabytes, não exigindo muito além desse valor conforme a Figura 18.

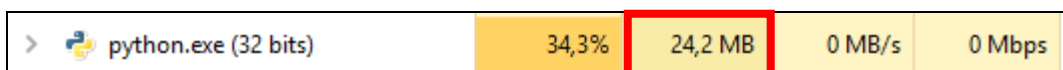


Figura 18: Ocupação de memória da ferramenta

5.3: Implantação e manutenção

A etapa de implantação e manutenção consiste no processo de entrega e liberação do software para o cliente para a sua utilização e também a correção de erros que não foram identificados nos estágios anteriores.

Entretanto este processo de entrega e liberação não foi cumprido visto que trata-se de uma etapa mais apropriada para produtos comerciais, o que não é o caso. Outro motivo consiste no fato de que o projeto global ainda está no início e continuará por mais tempo, e com isso a ferramenta deve ser aperfeiçoada conforme as necessidades do projeto. Porém, para suceder esta etapa, realizou-se uma prova de conceito com a apresentação e a demonstração da ferramenta na Unidade de Operação da Petrobras no Rio de Janeiro e também no CENPES – Centro de Pesquisa e Desenvolvimento da empresa. A ferramenta foi bem aceita pelos técnicos e engenheiros da Petrobras e foi recebida algumas sugestões de melhorias e modificações para a próxima versão.

5.4: Comentários Finais

Neste capítulo abordou-se as etapas de desenvolvimento da ferramenta, apresentando-se as bibliotecas utilizadas para a implementação, os testes realizados e as atividades realizadas na etapa de implantação e manutenção. O próximo capítulos tratará dos resultados obtidos com a ferramenta desenvolvida.

Capítulo 6: Avaliação dos resultados

Neste capítulo são apresentados os resultados obtidos com a ferramenta juntamente com exemplos de testes de suas principais funcionalidades para a verificação do cumprimento dos requisitos estabelecidos. Os resultados podem ser divididos em 3 partes: interface gráfica resultante, geração de arquivo XML, geração de arquivo PDF.

6.1: Interface gráfica

Conforme requisitos e projeto de interface apresentado anteriormente, a interface gráfica ficou constituída com o total de 7 janelas, sendo que 6 delas são para entrada de dados inseridos pelo usuário e que são reveladas a partir da janela principal onde está a matriz C&E. Cada uma das janelas é mostrada a seguir:

- Visão geral da ferramenta – Janela principal:

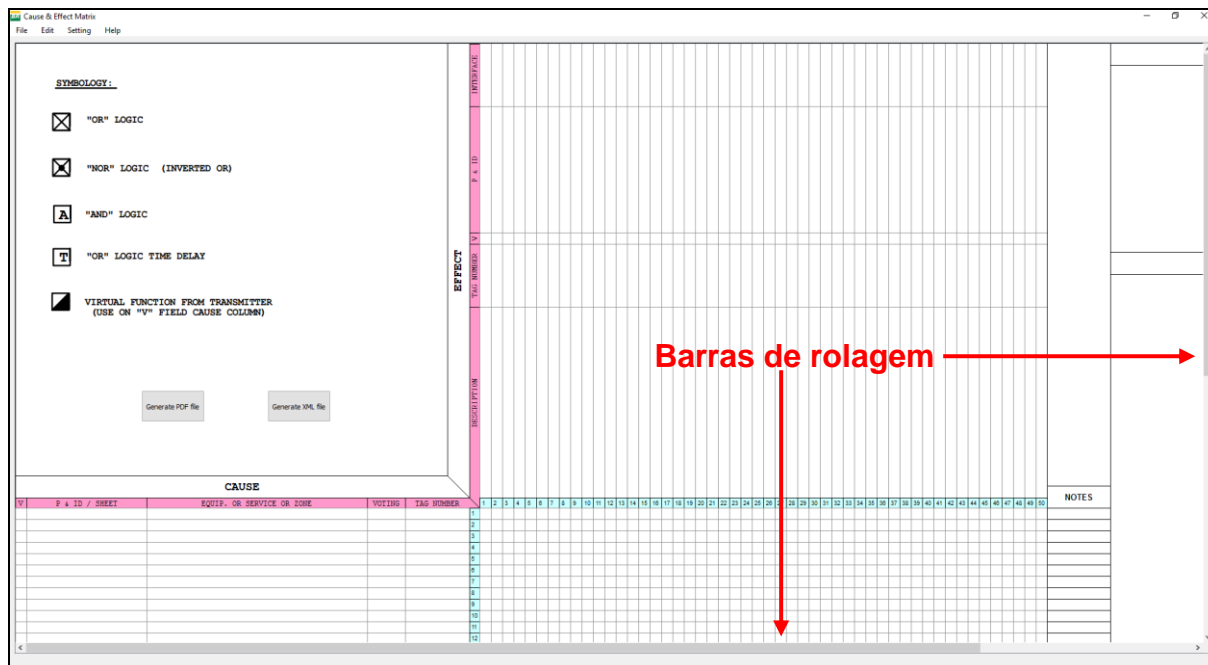


Figura 19: Janela principal com a interface da matriz

A janela na qual a matriz se encontra possui barras de rolagem vertical e horizontal para a navegação como visto na Figura 19 e possui o mesmo *layout* definido pela Petrobras.

- Janela para inserção das informações da causa:

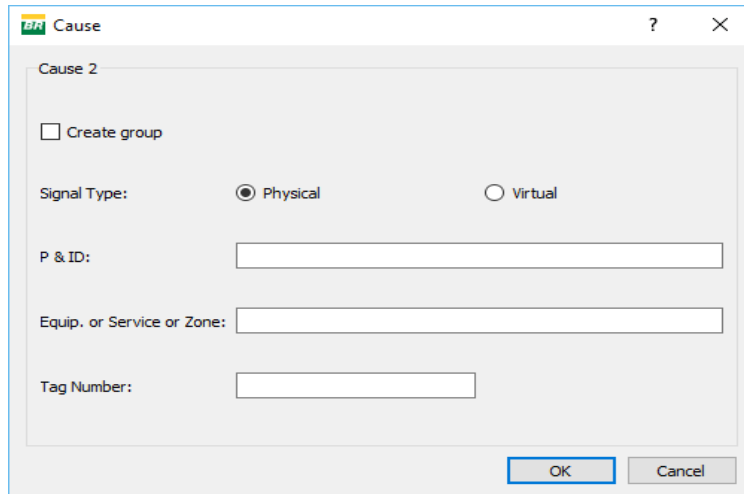


Figura 20: Janela para inserção de causas

Ao clicar em uma das linhas de causas, abrirá-se uma janela para a introdução dos dados da causa referente a linha clicada. A janela possui a opção para escolher o tipo de sinal e 3 campos de entrada de dados: P&ID, Equipamento/Serviço/Zona e a Tag do sensor. Há também a opção de se criar grupos de sensores marcando-se a opção “Create group”.

- Janela para inserção de dados do efeito:

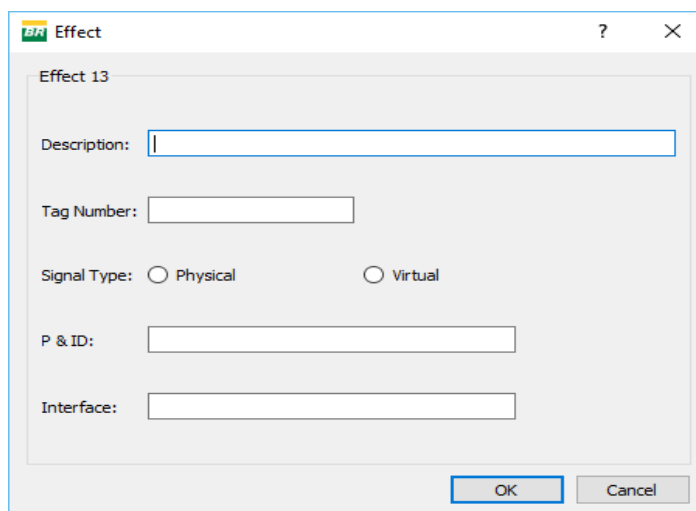


Figura 21: Janela para inserção dos efeitos

A janela da Figura é mostrada ao clicar em uma das colunas do efeito e nela é possível inserir todos os dados referentes ao instrumento de acionamento.

- Janela para escolha da lógica da intersecção:

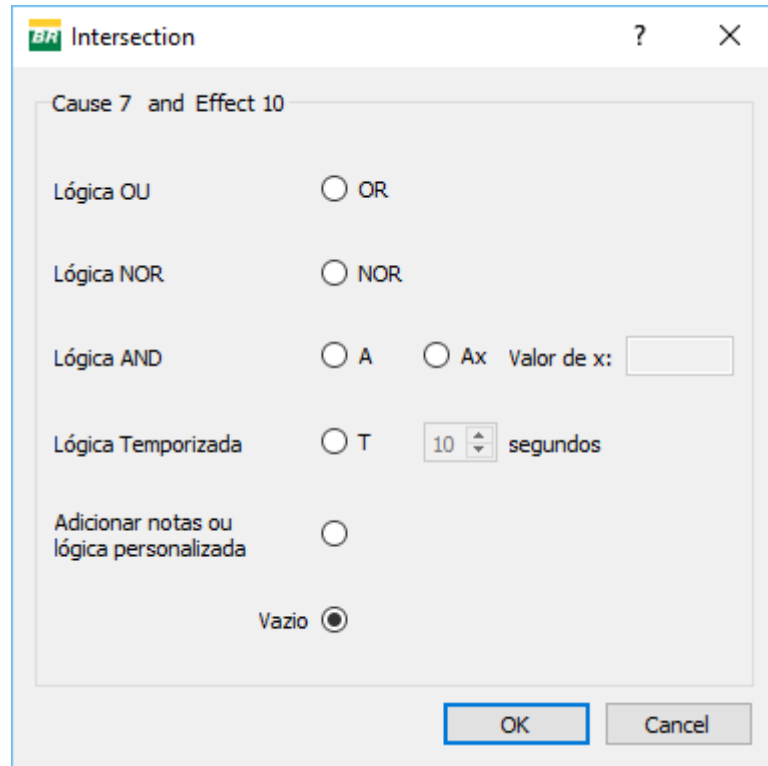


Figura 22: Janela para inserção da lógica

Ao clicar em uma das intersecções da matriz, aparecerá a janela mostrada na Figura acima para a escolha da lógica referente à intersecção clicada. Nela o usuário poderá escolher uma das lógicas dentre as opções definidas na diretriz.

- Janela de inserção de notas ou escolha de notas existentes:

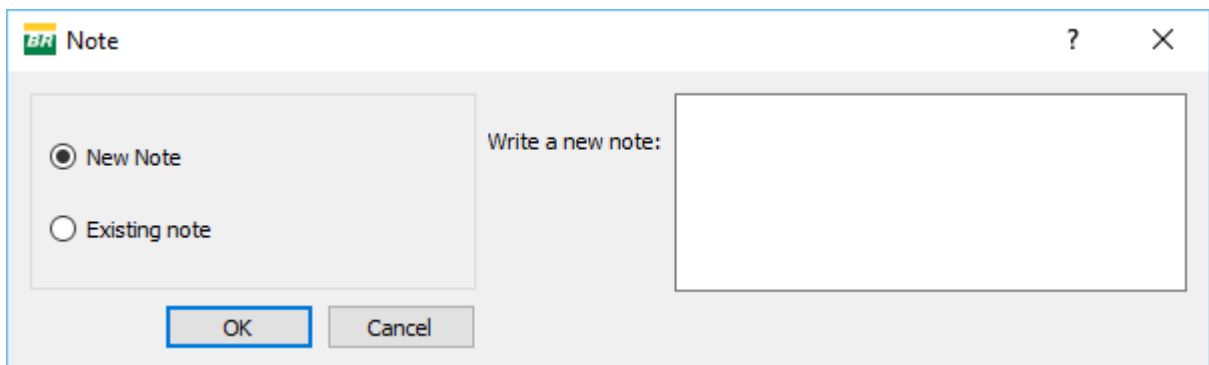


Figura 23: Janela para inserção de notas

Esta janela irá aparecer para a adição de notas e o usuário poderá escrever uma nova nota ou referenciar a uma nota já existente.

- Janela para inserir nome dos documentos de referência:

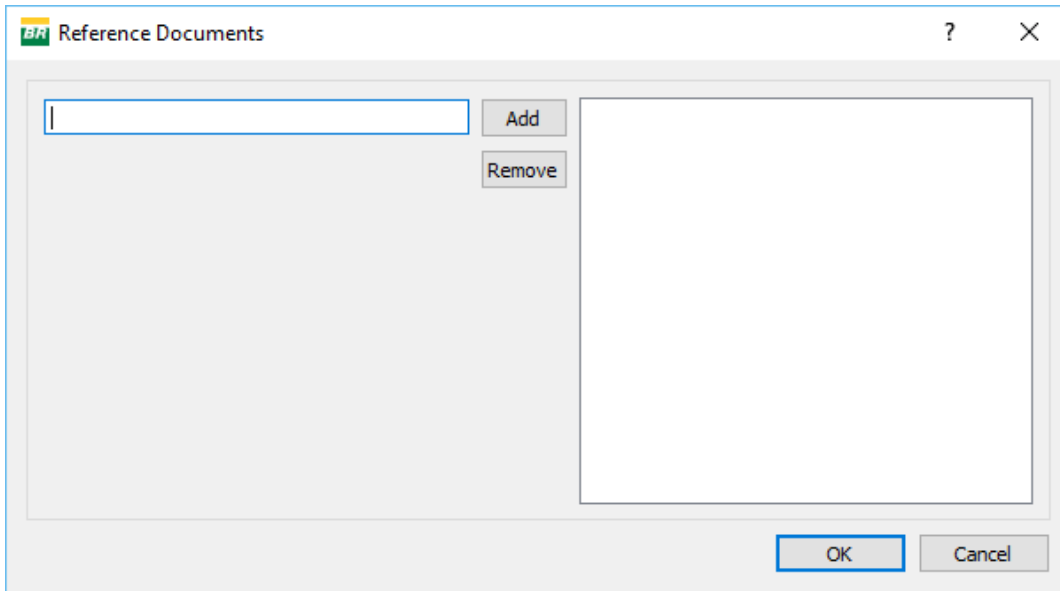


Figura 24: Janela para inserção de documentos de referência

Esta janela é mostrada quando o usuário clicar em “Referenced Documents”. Com isso pode-se adicionar novos nomes de documentos referenciado ou excluir um documento existente.

- Janela de inserção de dados do rodapé:

The image shows a software dialog box titled "Matrix ID". It features a standard Windows-style title bar with a question mark and a close button. The main area contains several text input fields for data entry. The fields are arranged vertically: "User:", "Project Title:", "Unit:", "Drawing Title:", "Designed by:", "Draw by:", "Verified by:", "Draw No.:", "Scale:", "Sheet: [] of []", "Date:" (with a date picker showing "01/07/2016"), and another "Draw No.:" field. At the bottom right, there are "OK" and "Cancel" buttons.

Figura 25: Janela para inserção de dados do rodapé

Ao clicar em alguma área do rodapé, aparecerá a janela da Figura 25 que por sua vez possui campos de entrada de texto para a inserção de todos os identificadores da matriz, como usuário, unidade, data, título do projeto etc.

Tendo em vista os requerimentos da interface gráfica pode-se dizer que a interface implementada cumpriu os objetivos estabelecidos, posto que todas as janelas de entrada de dados e a janela com a matriz foram implementadas com sucesso.

6.2: Geração de arquivo XML

A ferramenta desenvolvida é capaz de produzir um arquivo XML com todas as informações inseridas na matriz C&E conforme a estrutura apresentada na etapa de projeto. Para ilustrar o resultado, segue um simples exemplo de uma matriz com uma causa, um efeito e uma relação entre eles:

- Causa e Relação “OR”:

V	P & ID / SHEET	EQUIP. OR SERVICE OR ZONE	VOTING	TAG NUMBER	1
P	I-DE-3010.63-5425-944-TKP-001	HS CO2 ROOM SYSTEM ACTION		HS-571C007	1 X

Figura 26: Exemplo de causa

- Efeito:

	DESCRIPTION	TAG NUMBER	V	P & ID	INTERFACE
1	CO2 SYSTEM ACTUATION	XY-5425013		I-DE-3010.63-5425-944-TKP-001	

Figura 27: Exemplo de efeito

Arquivo XML resultante:

```

<matrix>
  <lines>
    <cause group="0" row="1">
      <v>p</v>
      <peid>I-DE-3010.63-5425-944-TKP-001</peid>
      <ecz>HS CO2 ROOM SYSTEM ACTION</ecz>
      <voting></voting>
      <tag_number>HS-571C007</tag_number>
      <notes>teste notes</notes>
      <relations>
        |
        <relation col="1">OR</relation>
      </relations>
    </cause>
  </lines>
  <columns>
    <effect col="0">
      <description>CO2 SYSTEM ACTUATION</description>
      <tag_number>XY-5425013</tag_number>
      <v>v</v>
      <peid>I-DE-3010.63-5425-944-TKP-001</peid>
      <interface></interface>
    </effect>
  </columns>
  <reference_documents/>
  <general_notes/>
  <footer/>
</matrix>

```

Figura 28: Exemplo de documento XML gerado pela ferramenta

Com a geração do arquivo XML foi resolvida a questão da necessidade de uma interface com a ferramenta de teste automático do projeto global, dispensando a necessidade de se transcrever manualmente os dados da matriz em um documento de texto.

6.3: Geração de arquivo PDF

Outro resultado positivo obtido foi a funcionalidade de geração de arquivo PDF da matriz C&E que é obtido através da função *QPixmap* da biblioteca de

interface gráfica e que realiza o mapeamento de pixel da própria interface gráfica e faz o armazenamento da imagem. A partir da imagem obtida, a biblioteca ReportLab faz a sua conversão para um arquivo PDF. A Figura 29 mostra o mesmo exemplo utilizado anteriormente, porém em um documento PDF.

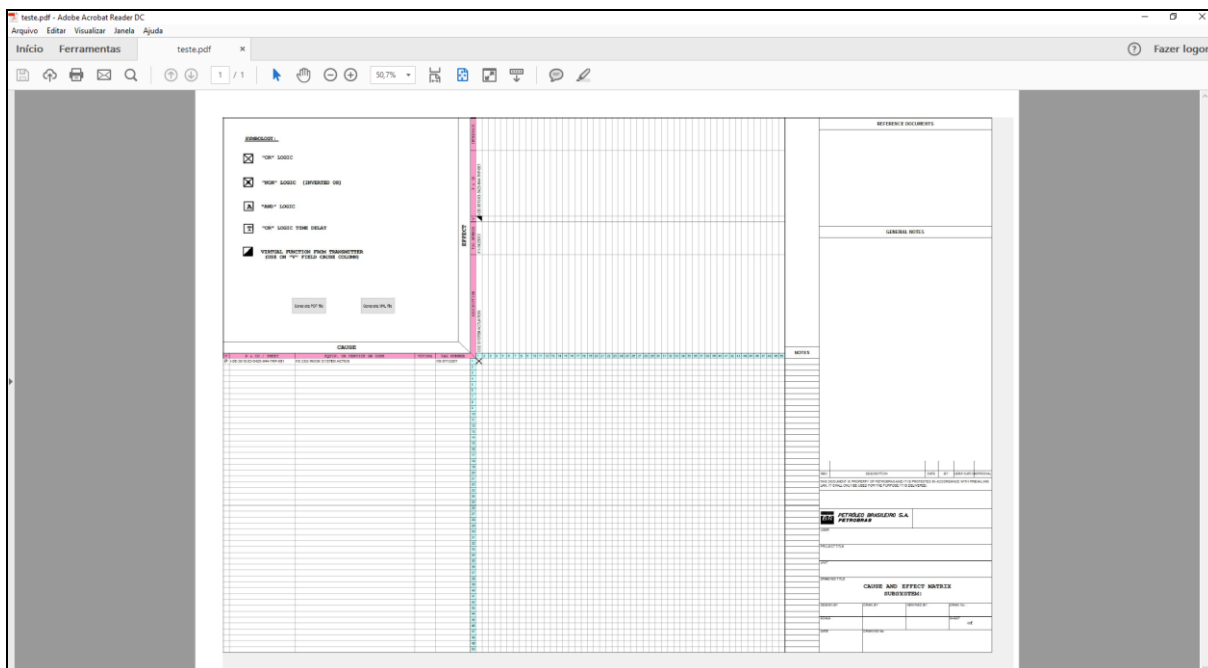


Figura 29: Exemplo de documento PDF gerado pela ferramenta

Dessa forma torna-se possível a fácil documentação da matriz C&E e também a impressão do documento para ser utilizado nos processos de projeto dos sistemas de segurança.

6.4: Prova de conceito

Para a apresentação e avaliação da ferramenta desenvolvida, foi realizada uma viagem para o Rio de Janeiro para efetuar uma prova de conceito com engenheiros da UO-Rio e do CENPES com a demonstração da edição de exemplos de matriz C&E da plataforma P-51 da Petrobras. Esta demonstração revelou a necessidade urgente de atualização da diretriz de elaboração de matriz C&E, visto que, segundo os engenheiros, a diretriz não já não atende as necessidades atuais para especificação da matriz, uma vez que ela foi elaborada há muito tempo.

Em geral, a ferramenta foi muito bem recebida pelos membros da empresa e com uma avaliação positiva, visto que solucionaria o problema da padronização,

substituindo a atual ferramenta de edição de matriz C&E que consiste no software de planilhas eletrônicas.

Outro resultado da prova de conceito foi o recebimento de sugestões de melhorias e adição de novas funcionalidades para a próxima versão da ferramenta. Entre elas estão:

- Funcionalidade de pré-filtragem de dados por zonas: Foi sugerida a adição de uma funcionalidade para que antes de se iniciar o preenchimento da matriz, indica-se de qual zona se trata a matriz para que dados comuns referente a zona escolhida já estejam presentes na matriz.
- Funcionalidade de busca automática: Foi proposta a função de busca de informações como “tags” e nomes em geral com o objetivo de facilitar o preenchimento e diminuir a necessidade de se digitar cada entrada da matriz.
- Mudança na forma de inserção de notas: Atualmente as notas podem ser inseridas juntamente com a lógica nas intersecções. A mudança propõe que a nota seja adicionada somente no campo de notas.
- Adição de campo de entrada para endereço de memória: Adicionar campos de entrada de endereço de memória referente a cada sensor ou atuador para salvá-los junto ao XML com o propósito de se utilizar na ferramenta de teste automático.

6.5: Comentários Finais

Neste capítulo foram apresentados os resultados obtidos com a ferramenta desenvolvida, mostrando-se partes da interface gráfica e o resultado da geração de documentos PDF e XML da matriz C&E. Apresentou-se também resultados da prova de conceito com os engenheiros da Petrobras obtendo-se sugestões de modificações e aperfeiçoamento da ferramenta.

Capítulo 7: Conclusões e Perspectivas

A importância da padronização das especificações dos intertravamentos de segurança é fundamental para o desenvolvimento de programas confiáveis e seguros, visto que especificações erradas ou incompletas podem levar a danos sérios e custosos de equipamentos, danos ambientais e até mesmo a perda de vidas humanas.

Neste trabalho foi abordado o desenvolvimento de uma ferramenta que tem como finalidade definir um padrão para a elaboração da matriz C&E, que por sua vez é um documento de especificações que relaciona os equipamentos destinados à segurança e sinais de sensores que devem ativar ou desativar tais equipamentos.

Para o desenvolvimento do software utilizou-se fundamentos de engenharia de software, a fim de sistematizar o desenvolvimento com o emprego do modelo cascata de processo de software. Com o modelo cascata a concepção da ferramenta foi dividida em cinco estágios: Análise e definição dos requisitos, Projeto/Modelagem, Implementação, Testes e Implantação. Com isso foram elaborados documentos de projeto/modelagem que irão auxiliar na manutenibilidade da ferramenta.

Os principais benefícios obtidos com a ferramenta desenvolvida neste trabalho foram a definição de um padrão das informações a serem inseridas na matriz C&E e a facilidade no processo de criação das matrizes C&E, dado que se obteve muito mais praticidade em relação à utilização de planilhas eletrônicas principalmente devido à interface gráfica mais simples e funcional.

Outra vantagem conquistada foi a geração de arquivo XML com as informações da matriz para a interface com ferramenta de teste automático do projeto assim como a geração de arquivo PDF para a documentação da matriz.

Mesmo com a finalização deste trabalho, o aprimoramento da ferramenta prosseguirá até o fim do projeto. Contudo, pode-se dizer que os objetivos inicialmente estabelecidos foram alcançados com sucesso pois as principais funcionalidades requeridas para a ferramenta foram implementadas com êxito.

7.1: Diretrizes para trabalhos futuros

Conforme sugestões recebidas da prova de conceito e das necessidades do projeto, para o futuro aperfeiçoamento da ferramenta, recomenda-se a adição de novos recursos e melhorias como:

- Adição de entrada de endereço de memória do CLP: Adicionar campos de entrada de endereço de memória referente a cada sensor ou atuador para salvá-los junto ao XML com o propósito de se utilizar na ferramenta de teste e simulação.
- Funcionalidade de busca e filtragem de dados: Para melhorar a usabilidade da ferramenta, torna-se interessante a função de busca e filtragem para que o usuário não tenha a necessidade de digitar todos os dados a serem inseridos na matriz.
- Função para realizar zoom na matriz: Funcionalidade para aumentar ou diminuir o tamanho da matriz conforme a necessidade do usuário.
- Inserir dicas: Apresentar ao usuário algumas regras da diretriz durante o preenchimento da matriz.
- Verificação de erros: Antes de se salvar alguma matriz tanto em PDF como em XML, a ferramenta fará uma verificação completa na matriz visando possíveis erros no preenchimento com dados incompletos ou incorretos.

Bibliografia:

- [1] J. E. Skogdalen, O. Smogeli. Looking Forward-Reliability of Safety Critical Control Systems on Offshore Drilling Vessels. Deepwater Horizon Study Group Working Paper-January 2011.
- [2] Gruhn, P. and Cheddie, H. L. (2006). Safety Instrumented Systems - Design, Analysis, and Justification. ISA, 2nd edition.
- [3] Especificação técnica ET-3000.00-1200-800-PGT-006. Petrobras, 2000.
- [4] COMISSÃO DE NORMAS TÉCNICAS. Norma técnica N-1883. [S.I.]: Petrobras, 2002.
- [5] COMISSÃO DE NORMAS TÉCNICAS. Norma técnica N-2595. [S.I.]: Petrobras, 2012.
- [6] Prati, T. J. (2014). Desenvolvimento de uma metodologia para realização de testes em programas de CLP na indústria de petróleo e gás. In Dissertação de mestrado em Engenharia de Automação e Sistemas - PPGEAS.
- [7] "IEEE Standard Glossary of Software Engineering Terminology", IEEE Std 610 12-1990, December 1990, p. 67
- [8] Pressman, R. Engenharia de Software: Uma abordagem Profissional. 6ª edição. Editora Bookman, 2010.
- [9] Sommerville, Ian. Engenharia de Software, 9ª edição, Pearson Prentice Hall, 2011.
- [10] World Wide Web Consortium (W3C), Extensible Markup Language (XML). Disponível em: < <https://www.w3.org/XML/>>. Acesso em: 16 de maio 2016.
- [11] Python Software Foundation, Python Documentation Overview. Disponível em: < <https://docs.python.org/3/>>. Acesso em: 25 de abril 2016.
- [12] Riverbank Computing, PyQt Whitepaper. Disponível em: < <https://www.riverbankcomputing.com/static/Docs/PyQt4/pyqt-whitepaper-a4.pdf>>. Acesso em: 02 de maio 2016.

- [13] Reportlab Inc, ReportLab Documentation. Disponível em: <
<http://www.reportlab.com/docs/reportlab-userguide.pdf> >. Acesso em: 05 de maio 2016.
- [14] Python Software Foundation, Minimal DOM implementation. Disponível em <
<https://docs.python.org/2/library/xml.dom.minidom.html> >. Acesso em 18 de maio 2016.

Apêndice A – Estrutura tipo árvore do arquivo XML projetado

