

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

Desenvolvimento de sistema de gerenciamento de informações de ferramentas

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

DAS 5511: Projeto de Fim de Curso

Guilherme Ranzolin Piazzetta

Florianópolis, Fevereiro de 2013

Desenvolvimento de sistema de gerenciamento de informações de ferramentas

Guilherme Ranzolin Piazzetta

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. Agostinho Plucenio

Assinatura do Orientador

*“A tarefa não é tanto ver
aquilo que ninguém viu, mas
pensar o que ninguém ainda
pensou sobre aquilo que
todo mundo vê.”*

(Arthur Schopenhauer)

AGRADECIMENTOS

Antes de tudo, agradeço ao Pai Celestial que me acompanhou em todos os momentos e colocou em meu caminho pessoas e oportunidades certas nos momentos certos.

Aos meus queridos e amados pais, Thaís e Ivanor, que nunca mediram esforços para me proporcionar o melhor estudo, pelo amor e por estarem junto comigo em todos os momentos e em todos os sentidos.

Ao meu irmão Gustavo, pela amizade e companheirismo.

À minha namorada Paula, que mesmo de longe esteve sempre ao meu lado e foi muito importante nesta jornada tornando as vitórias mais saborosas e as dificuldades mais brandas.

A todos meus amigos. Aos velhos e novos amigos de Macaé que tornaram os momentos de folga mais descontraídos e o ambiente de trabalho mais agradável. Aos da faculdade, que mesmo cada vez mais longe buscando seu espaço nos mais diversos lugares, acompanho suas vitórias e aguardo ansioso pelo nosso reencontro. E aos amigos de Araranguá, que apesar da distância e do tempo, nunca perdemos o contato e a amizade.

Aos mestres do Departamento de Automação e Sistemas da UFSC pela base sólida e princípios profissionais oferecidos em minha formação. Em especial ao professor Agostinho Plucenio pela orientação, conselhos e motivação durante o desenvolvimento deste projeto.

Por fim, a todos os amigos e funcionários do setor de Testing & Subsea da base Halliburton de Macaé-RJ. Graças à paciência e dedicação dos diretores, coordenadores, supervisores, engenheiros, técnicos, dentre tantos outros, foi possível que todo o conhecimento fosse adquirido e a expectativa de aproveitamento superada.

RESUMO

O presente documento visa relatar e detalhar o projeto de fim de curso (disciplina DAS 5511) desenvolvido na empresa Halliburton, tendo todas as suas etapas desenvolvidas pelo estudante Guilherme Ranzolin Piazzetta durante o período de 13/08/2012 a 05/02/2013, durante a realização do estágio final do curso de Engenharia de Controle e Automação na base operacional Halliburton de Macaé – RJ.

O mundo vive um momento de grande crescimento por parte da indústria do petróleo e o Brasil, em especial, acompanha este desenvolvimento como protagonista alçado pelas recentes descobertas de reservas de hidrocarbonetos. Se o presente é bom para as empresas do ramo, o futuro trás possibilidades de crescimento ainda maior, porém para isto é necessário que as grandes companhias comecem a se preparar desde já.

Com a alta operacional esperada, novos equipamentos são adquiridos diariamente para suprir a demanda. Este crescimento dos setores trás alguns “problemas” que podem refletir na qualidade dos serviços e segurança das operações. Um destes problemas remete a organização do grande volume de informações referentes aos equipamentos que devem ser armazenados e atualizados diariamente.

Neste contexto, o projeto em questão propõe a criação de um sistema que unifique as informações referentes aos equipamentos e ferramentas, de modo que se possa afirmar com agilidade e convicção como se encontra cada equipamento ou em qual etapa da manutenção ele está. Este desenvolvimento é feito fazendo-se uso de algumas ferramentas como o *framework PHP CakePHP* e o sistema de gerenciamento de banco de dados *MySQL*.

Palavras-chave: Petróleo, Halliburton, banco de dados, CakePHP, MySQL.

ABSTRACT

The objective of this document is to report and detail the final project developed at Halliburton Company. All stages were done by the student Guilherme Ranzolin Piazzetta from 13/08/2012 to 05/02/2013, during the final internship of the Control and Automation Engineering at the Halliburton headquarter in Macaé – RJ.

Today the world lives a great growing moment for oil and gas industry, and Brazil leads this growing because of the recent discovers of hydrocarbons reservoirs. If the present is good for these companies, the future brings possibilities to grow even more, but it's necessary to be prepared for it.

Waiting for a great demand, new equipments are bought everyday to supply it. This growing of all departments brings some problems that can reflect in the service quality and operational safety. One of these problems is related to the big volume of information of the equipments that must be filed and updates every and each day.

In this context, the project proposes the development of a system to unify all informations related to equipments and tools allowing supervisors to confirm the tools status and in maintainance stage for example, quickly and certainly. For this project it was used programming tools like a framework called CakePHP and the Database Management System MySQL.

Palavras-chave: Oil and Gas, Halliburton, DataBase, CakePHP, MySQL.

SUMÁRIO

Agradecimentos	iv
Resumo	v
Abstract	vi
Índice de Figuras	x
Índice de Quadros	xii
Lista de siglas.....	xiii
Glossário técnico da indústria do petróleo e gás.....	xiv
Capítulo 1: Introdução	15
1.1. O petróleo e a indústria	15
1.2. O Brasil no mercado do petróleo	16
1.3. Halliburton Energy Services.....	18
1.4. Testing & Subsea	19
1.4.1. <i>Test Tools</i> (TT).....	21
1.4.1.1. Ferramentas principais	22
1.4.2. <i>Subsea Safety Systems</i> (SSS).....	23
1.4.2.1. Ferramentas principais	24
1.4.3. <i>Data Acquisition Service</i> (DAS).....	24
1.4.3.1. Ferramentas principais	24
1.4.4. <i>Fluid Sampling</i> (SAM)	25
1.4.4.1. Ferramentas principais	25
1.4.5. <i>Surface Well Testing</i>	26
1.4.5.1. Ferramentas principais	26
1.5. Programa Trainee 2012.b.....	27
1.6. Estrutura do documento	27
1.7. Relação com Eng. de Controle e Automação	28
Capítulo 2: Apresentação do projeto	30

2.1: Contexto.....	30
2.2: Padrão 510.....	31
2.3: Motivação/Justificativa	32
2.4: Objetivos	35
Capítulo 3: Criação e concepção do sistema	37
3.1: Processo Unificado de desenvolvimento de <i>software</i> (PU).....	37
3.2: <i>Unified Modeling Language</i> (UML).....	39
3.3: Fase 1: Concepção	41
3.3.1. Visão geral e levantamento dos requisitos do sistema.....	42
3.3.2. Casos de uso e organização dos requisitos.....	47
3.4: Fase 2: Elaboração	51
3.4.1. Expansão dos casos de uso.....	51
3.4.2. Operações e consultas do sistema	53
3.4.3. Modelo de sistema em <i>Wireframe</i>	55
3.5: Projeto de Banco de Dados	57
Capítulo 4: Desenvolvimento e Implementação do Sistema	60
4.1: Linguagem de programação	61
4.1.1: PHP (<i>Hypertext PreProcessor</i>)	62
4.2: Sistema de Gerenciamento de Banco de Dados (SGBD).....	64
4.2.1: MySQL.....	64
4.2.1.1: Modelo Lógico.....	65
4.2.1.2: Modelo Físico.....	66
4.3: Servidor.....	68
4.3.1: Apache.....	68
4.3.2: Xampp	68
4.4: Framework	69
4.4.1: CakePHP	70

4.5: MVC (<i>Model-View-Controller</i>)	71
4.5.1: Modelos	72
4.5.2: Controladores	76
4.5.3: Visões	78
4.5.3.1: Elements	79
Capítulo 5: Detalhamento	81
5.1: Permissões	81
5.1.1: Usuário não autenticado	82
5.1.2: Usuário comum	85
5.1.3: Administrador	86
5.2: Casos de Uso	88
5.2.1: Mostrar <i>Status</i> de ferramentas (F1)	88
5.2.2: Mudar <i>Status</i> de ferramentas (F2)	89
5.2.3: Gerenciamento do sistema (F3)	90
5.2.4: Gerenciar Manutenção (F4)	93
5.2.5: Gerar Relatórios (F5)	95
Capítulo 6: Resultados, Conclusões e Futuro do Projeto.	97
6.1: Resultados	97
6.2: Conclusões	99
6.3: Futuro do projeto	101
Capítulo 7: Bibliografia	102

ÍNDICE DE FIGURAS

FIGURA 1 - CRESCIMENTO DO CONSUMO MUNDIAL DE PETRÓLEO AO LONGO DOS ANOS.	16
FIGURA 2 - MAIORES PRODUTORES MUNDIAIS DE PETRÓLEO	16
FIGURA 3 - PERFIL DA PRODUÇÃO DOS PAÍSES DA AMÉRICA DO SUL NOS ÚLTIMOS ANOS.	17
FIGURA 4 - RESERVAS DE PETRÓLEO E GÁS NATURAL NO BRASIL	18
FIGURA 5 - LOGOTIPO HALLIBURTON	18
FIGURA 6 - DIVISÃO SETORIAL POR ÁREAS DE ATUAÇÃO.....	19
FIGURA 7 - DIVISÃO DO SETOR DE TSS HALLIBURTON.....	21
FIGURA 8 - COLUNA BÁSICA DE TESTES.....	22
FIGURA 9 - ESQUEMA DE DESCONEÇÃO DA COLUNA DE TESTE	23
FIGURA 10 - REGISTRADORES ELETRÔNICOS DE MEMÓRIA	25
FIGURA 11 - AMOSTRADORES NO PORTA AMOSTRADOR.....	25
FIGURA 12 - PLANTA DE SUPERFÍCIE.....	26
FIGURA 13 - FLUXOGRAMA DA MANUTENÇÃO DE REGISTRADORES DE MEMÓRIA DE DAS	31
FIGURA 14 – QUADRO DE STATUS DE EQUIPAMENTOS.....	33
FIGURA 16 – TABELA EXCEL DE CONTROLE DE INFORMAÇÕES DE FERRAMENTAS.....	34
FIGURA 15 – FOLHA DE ENGENHARIA DE FERRAMENTA	34
FIGURA 17 – 4 FASES DO PROCESSO UNIFICADO.....	38
FIGURA 18 – CLASSIFICAÇÃO DOS TIPOS DE DIAGRAMA UML.....	40
FIGURA 19 – DIAGRAMA DE CASOS DE USO.....	49
FIGURA 20 – DIAGRAMA DE SEQUENCIA PARA O CASO DE USO “ABRIR MANUTENÇÃO”	55
FIGURA 21 – WIREFRAMES	56
FIGURA 22 – FASES DE PROJETO DE BD	57
FIGURA 23 – MODELO CONCEITUAL DO BANCO DE DADOS	58
FIGURA 24 – MODELO LÓGICO.....	66
FIGURA 25 – INTERFACE PHPMYADMIN	67
FIGURA 26 – ESTRUTURA DO CAKEPHP	70
FIGURA 27 – REQUISICÃO MVC.....	72
FIGURA 28 – MENUS DE NAVEGAÇÃO PARA OS 3 NÍVEIS DE USUÁRIOS.....	80
FIGURA 29 – PÁGINA INICIAL PARA USUÁRIO NÃO AUTENTICADO	82
FIGURA 30 – INFORMAÇÕES DE FERRAMENTA COM DESTAQUE PARA LINK PARA MANUAL DA FERRAMENTA.	83
FIGURA 31 – LISTA DE MANUTENÇÕES ORDENADAS PELO <i>EQUIPMENT NUMBER</i>	83
FIGURA 32 – VISUALIZAÇÃO DE MANUTENÇÃO.	84
FIGURA 33 – FORMULÁRIO DE MANUTENÇÃO EM PDF.....	84
FIGURA 34 – PÁGINA INICIAL DO USUÁRIO COMUM.	85
FIGURA 35 – MENU DE GERENCIAMENTO DE MANUTENÇÃO.	86

FIGURA 36 – PÁGINA INICIAL DO ADMINISTRADOR.	87
FIGURA 37 – PÁGINA DE GERENCIAMENTO DE USUÁRIOS.	87
FIGURA 38 – ALTERAÇÃO DO STATUS DE FERRAMENTA.	89
FIGURA 39 – PARTE SUPERIOR DA PÁGINA PARA ADIÇÃO DE NOVA FERRAMENTA NO BANCO DE DADOS	90
FIGURA 40 – VISUALIZAÇÃO DE FERRAMENTA POR ADMINISTRADOR COM DETALHE PARA MENU DE GERENCIAMENTO.....	91
FIGURA 41 – CONFIRMAÇÃO DE EXCLUSÃO DE FERRAMENTA.	91
FIGURA 42 – MENSAGEM DE EXCLUSÃO COM SUCESSO.....	92
FIGURA 43 – EDIÇÃO DE USUÁRIO.	92
FIGURA 44 – CONFIRMAÇÃO DE EXCLUSÃO DE USUÁRIO.	93
FIGURA 45 – LISTA DE MANUTENÇÕES DA FERRAMENTA SELECIONADA.	94
FIGURA 46 – PÁGINA DE EDIÇÃO DE MANUTENÇÃO.	94
FIGURA 47 – CONFIRMAÇÃO DE EXCLUSÃO DE MANUTENÇÃO.	95
FIGURA 48 – RELATÓRIO DE MANUTENÇÃO DE GAUGES (DAS)	96
FIGURA 49 – RELATÓRIO DE STATUS DE FERRAMENTAS.	96
FIGURA 50 – MODELO CONCEITUAL SEGUNDA VERSÃO DO SISTEMA.....	101

ÍNDICE DE QUADROS

QUADRO 1 – VISÃO GERAL DO SISTEMA	42
QUADRO 2 – REQUISITO FUNCIONAL F1: MOSTRAR INFORMAÇÕES DE FERRAMENTAS	44
QUADRO 3 – REQUISITO FUNCIONAL F2: MUDAR <i>STATUS</i> DE FERRAMENTAS	44
QUADRO 4 – REQUISITO FUNCIONAL F3: GERENCIAMENTO DO SISTEMA	45
QUADRO 5 – REQUISITO FUNCIONAL F4: GERENCIAR MANUTENÇÃO	46
QUADRO 6 – REQUISITO FUNCIONAL F5: GERAR RELATÓRIOS.....	47
QUADRO 7 – CASOS DE USO	48
QUADRO 8 – CONSULTAS DO SISTEMA	50
QUADRO 9 – MÉTODO <code>ADD()</code> PARA ABRIR NOVA MANUTENÇÃO	63
QUADRO 10 – NÍVEIS DE ABSTRAÇÃO DE DADOS	65
QUADRO 11 – CRIAÇÃO DAS TABELAS DA BASE DE DADOS	67
QUADRO 12 – ASSOCIAÇÕES ENTRE MODELOS	73
QUADRO 13 – REGRAS UTILIZADAS PARA INCLUSÃO DE DADOS NOS CAMPOS DOS MODELOS	73
QUADRO 14 – MODELO ‘FERRAMENTA’	74
QUADRO 15 – MODELO ‘MANUTENCAO’	74
QUADRO 16 – MODELO ‘USER’	75
QUADRO 17 – MODELO ‘MANUTENCAOHASUSER’	75
QUADRO 18 – CONTROLADOR DO MODELO ‘USER’	76
QUADRO 19 – CONTROLADOR DO MODELO ‘FERRAMENTA’	77
QUADRO 20 – CONTROLADOR DO MODELO ‘MANUTENCAO’	77
QUADRO 21 – CONTROLADOR DO MODELO ‘MANUTENCAOHASUSER’	78
QUADRO 22 – VISÕES DO SISTEMA.....	78
QUADRO 23 – ELEMENTOS DE VISÕES.....	79

LISTA DE SIGLAS

IBGE:	Instituto Brasileiro de Geografia e Estatística
BP:	British Petroleum
TSS:	Testing & Subsea
PSL:	Product Service Line
TT:	Test Tools
SSS:	Subsea Safety Systems
DAS:	Data Acquisition Services
SAM:	Sampling
SWT:	Surface Well Testing
BHA:	Bottom Hole Assembly
TFR:	Teste de formação de poços revestidos
APR:	Annulus Pressure Responsive
AST:	Árvore Submarina de Teste
BOP:	Blow Out Preventer
PVT:	Pressão, Volume e Temperatura
PU:	Processo Unificado
EV:	Evento de Sistema
RS:	Resposta de Sistema
UML:	Unified Modeling Language
OO:	Orientado a Objeto
SGBD:	Sistema de Gerenciamento de Banco de Dados
PHP:	Hypertext Preprocessor - acrônimo recursivo
HTML:	HyperText Markup Language
SQL:	Structured Query Language
BD:	Banco de Dados
MVC:	Model-View-Control

GLOSSÁRIO TÉCNICO DA INDÚSTRIA DO PETRÓLEO E GÁS

Cimentação: Operação de deslocamento de uma pasta de cimento, íntegra e não contaminada, no espaço anular entre um poço e a coluna de revestimento, de modo a isolar hidraulicamente as diversas formações atravessadas.

Perfilagem: é uma operação de registro das características físicas das formações geológicas e dos fluidos presentes nas mesmas condições mecânicas do poço, através de sensores apropriados.

Completação: Conjunto de operações realizadas com o objetivo de condicionar o poço para sua colocação em produção.

Revestimento: Coluna de tubos de grande diâmetro, descida logo após a perfuração de determinado trecho de poço, que tem como função básica a contenção mecânica da parede do poço.

Coluna: Sequencia de tubos conectados uns aos outros por roscas ou flanges. Empregada nas operações de perfuração, completação e elevação dentre outras.

Espaço anular: Espaço que se forma dentro do poço, entre a parede interna do revestimento e a parede externa da coluna.

Amostras representativas: A amostra pode ser considerada representativa quando representa fielmente em condições de reservatório características desejadas para avaliação.

Análise PVT: Simulação das condições de reservatório através da análise do comportamento do fluido durante sua vida produtiva. Análise do comportamento das fases em função da variação de pressão.

*Fonte: Dicionário do Petróleo (Fernández; Junior; Pinho; 2009)

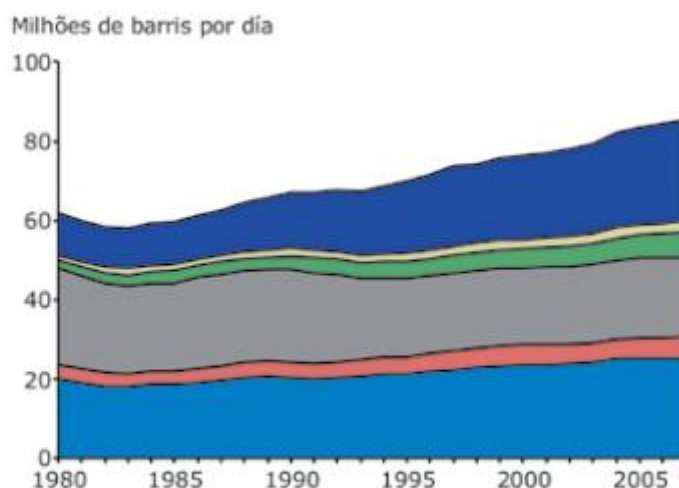
CAPÍTULO 1: INTRODUÇÃO

Há quem diga que não exista bem material de que a humanidade seja mais dependente do que o petróleo. Esta dependência ficou evidente no século XX com as revoluções Industriais que alteraram drasticamente a forma de produção e o consumo da sociedade. Segundo (CERQUEIRA), a geração e distribuição de energia se tornaram essenciais para a obtenção de resultados satisfatórios no setor industrial, pois o trabalho executado pelas máquinas, que são cada vez mais comuns no processo produtivo, só é possível através de energia. E neste contexto a sociedade que antes era dependente da força de trabalho manual e em sua maioria agrícola, a partir daí iniciou sua dependência por utilização de combustíveis, colocando em destaque os de origem fóssil.

1.1. O PETRÓLEO E A INDÚSTRIA

Ainda que pesquisas caminhem em direção a uma fonte de energia menos poluente, renovável e que permita o progresso mesmo quando o petróleo se esgotar, segundo pesquisas recentes (IBGE, 2010) 48,7% da energia utilizada no Brasil é gerada a partir de petróleo e gás natural. Com isso pode-se concluir que atualmente não se pode abrir mão deste combustível.

A moderna indústria do petróleo tem seu início datado de 1859 quando o norte-americano Edwin Laurentine Drake perfurou o primeiro poço nos Estados Unidos. A partir de então, esta indústria vem acompanhando o consumo mundial crescendo gradualmente ao longo dos anos como visto na Figura 1 e atualmente se tornou a indústria mais lucrativa com seis companhias entre as 10 mais lucrativas do mundo, segundo Antonia Juhasz em seu livro *A Tirania do Petróleo* (JUHASZ, 2009).



Fonte: BP STATISTICAL REVIEW (2008). Elaboração Bain & Company.

Figura 1 - Crescimento do consumo mundial de petróleo ao longo dos anos.

1.2. O BRASIL NO MERCADO DO PETRÓLEO

Embora não figure entre os maiores produtores mundiais segundo a *Key World Energy Statistics, 2012* (ver Figura 2), o Brasil se encontra em posição confortável uma vez que hoje figura na 2ª posição da América Latina com grande crescimento na sua produção, ao contrário da Venezuela, atual 1ª colocada, que demonstra estagnação em sua produção como mostra a BP no gráfico abaixo (Figura 3) datado de 2010.

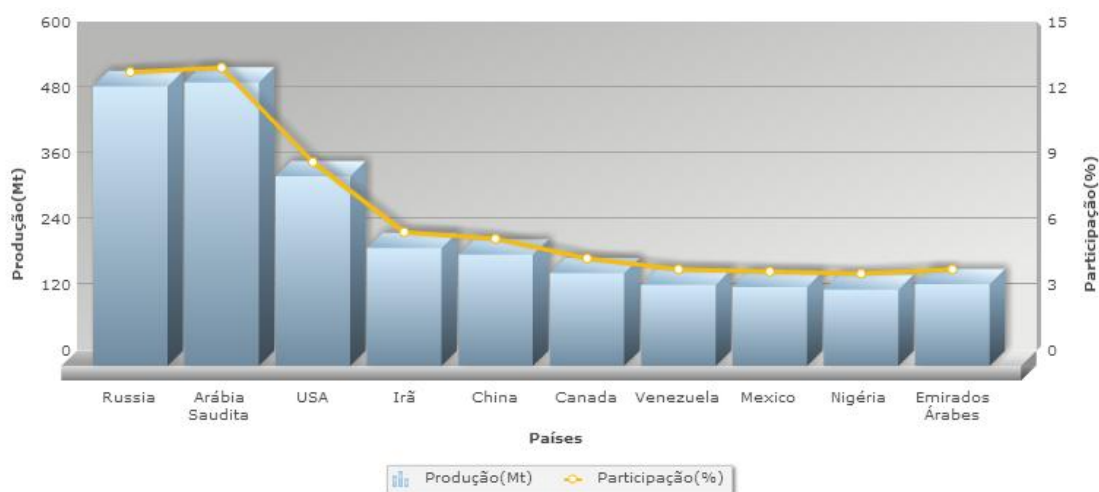


Figura 2 - Maiores produtores mundiais de petróleo

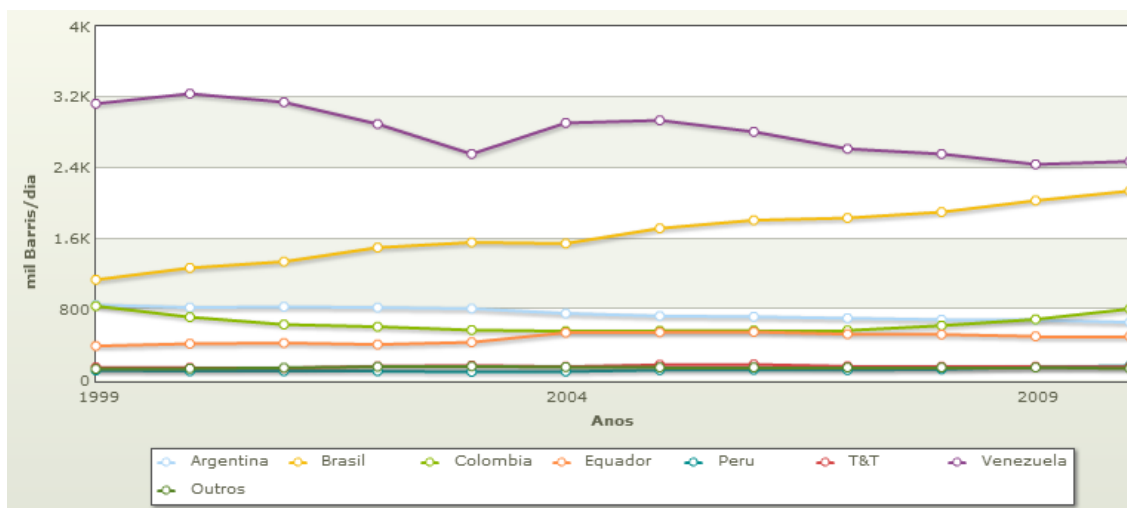


Figura 3 - Perfil da produção dos países da América do Sul nos últimos anos.

Segundo maior produtor de petróleo na América do Sul, o Brasil vive em constante crescimento no setor. No fim dos anos 1970, o País produzia, em média, 200 mil barris de petróleo por dia. Em 2009, alcançou a marca de dois milhões de barris diários.

O crescimento da produção neste período associou-se às grandes descobertas marítimas de petróleo e gás na Bacia de Campos, que começou com a descoberta do campo de Garoupa (RJ), em 1974, iniciando a busca em águas cada vez mais profundas. Nos anos 1980 e 1990, foram descobertos campos gigantes naquela bacia.

No primeiro semestre de 2008, a Petrobras anunciou a descoberta de um campo de petróleo no pré-sal (abaixo de uma camada geológica de dois quilômetros de espessura de sal) na Bacia de Santos. O potencial dessa descoberta e das seguintes na região do pré-sal coloca o Brasil no mesmo nível de reserva dos grandes produtores mundiais.

Uma vez que estas reservas se encontram em sua maioria em águas ultraprofundas (profundidade maior que 1000 metros) como mostra a Figura 4, mais do que nunca tecnologias de ponta são necessárias para atingir estes reservatórios.

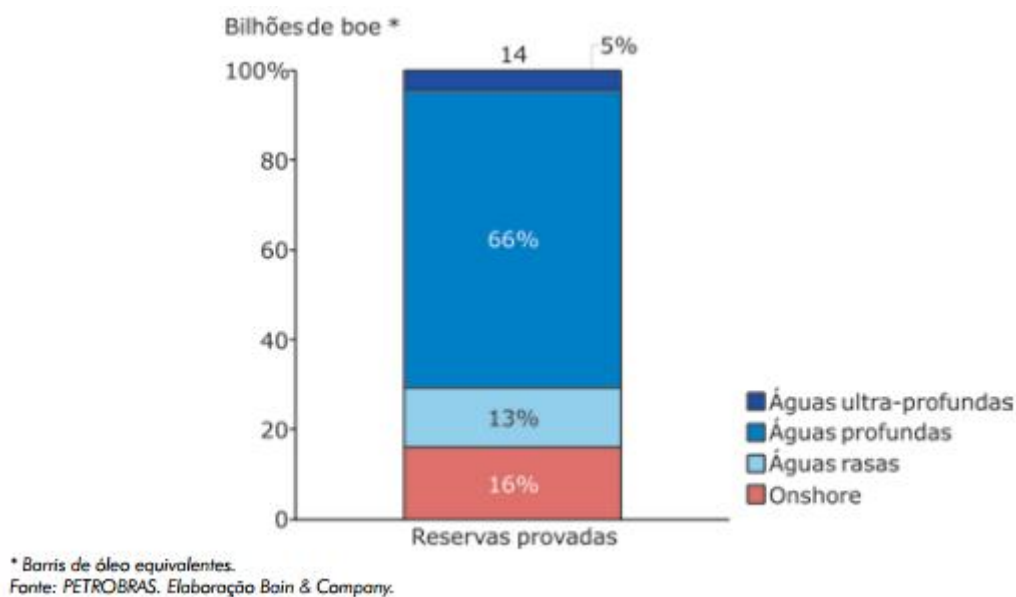


Figura 4 - Reservas de petróleo e gás natural no Brasil

1.3. HALLIBURTON ENERGY SERVICES

A Halliburton é uma empresa multinacional fundada em 1919 nos Estados Unidos, hoje presente em mais de 80 países. Ela é a segunda maior empresa do mundo em fornecimento de produtos e serviços para a indústria de petróleo e gás, presente há 50 anos no Brasil com bases em diversos estados. Hoje em dia, conta com mais de 70 mil colaboradores em todo o mundo.



Figura 5 - Logotipo Halliburton

Segundo o livro (International Directory of Company Histories, 2003), a empresa fundada em 1919 é uma das maiores fornecedoras mundiais de produtos e serviços para a indústria de petróleo e gás.

A Halliburton revela um foco contínuo em inovação e expansão, que começou com o fundador da empresa, Erle P. Halliburton. Após empréstimo de um vagão, mulas e uma bomba, ele construiu uma caixa para mistura feita de madeira, iniciando um empreendimento de cimentação de poços em Duncan, Oklahoma.

Após décadas de expansão e prestação de serviços por todo o globo, a Halliburton chega ao século XXI consolidada no mercado. A empresa realinhou seu trabalho em operações nos hemisférios oriental e ocidental, em 2006 e em 2007 segregou suas ofertas de serviços em duas divisões: Completação e Produção (*Completion and Production*), e Perfuração e Avaliação (*Drilling and Evaluation*). A organização setorial da empresa é mostrada na Figura 6.



Figura 6 - Divisão setorial por áreas de atuação.

Hoje, a Halliburton oferece a mais ampla gama mundial de produtos, serviços e soluções integradas para exploração de petróleo e gás, desenvolvimento e produção, apresentando um valor de mercado de US\$31,88 bilhões, receita anual de US\$24,83 bilhões e sede em Houston, no Texas.

1.4. TESTING & SUBSEA

Cada um dos setores mostrados anteriormente que atuam em diferentes áreas é chamado de PSL (*Product Service Line*). Para entender

melhor o setor em que o presente projeto foi desenvolvido deve-se ter uma noção básica do que é a avaliação de reservatórios e a importância deste processo para as empresas exploratórias.

Segundo (THOMAS, 2004), denominam-se “Avaliação das Formações” as atividades e estudos que visam definir em termos qualitativos e quantitativos o potencial de uma jazida petrolífera, isto é, sua capacidade produtiva e a valoração das suas reservas de óleo e gás.

Apesar dos indícios durante a perfuração e a perfilagem indicarem a presença de hidrocarbonetos na formação, isto não significa que podem ser produzidos economicamente. Somente o teste de formação (ou seja, colocar o poço em fluxo, de modo geral) poderá confirmar, com segurança, a presença de hidrocarbonetos na formação e fornecer dados a respeito das condições de fluxo nas imediações do poço.

É nesta etapa que o PSL de Testing and Subsea (TSS) da Halliburton exerce sua função de fundamental importância para que o cliente tome decisões econômicas e de engenharia a respeito da completação e produção do poço.

Os Testes de Formação correspondem a uma completação provisória do poço com o objetivo de identificar os fluidos, determinar os parâmetros de reservatório associados à produtividade e avaliar a extensão da jazida. Os testes nos poços de petróleo e gás são realizados em diversas etapas da perfuração, conclusão e a produção. Os objetivos do teste em cada etapa abrangem desde a simples identificação dos fluidos produzidos até a determinação da disponibilidade do reservatório para a identificação de características complexas.

O Teste de Formação consiste em provocar uma perturbação no sistema e analisar as repostas de pressão estática e dinâmica do reservatório. A interpretação da resposta de pressão, variação de vazão e análise de amostras de fluida representativas coletadas no fundo do poço permitirão a

obtenção de algumas propriedades do sistema poço-reservatório que darão condições para que importantes características do reservatório sejam inferidas.

O setor de Testing & Subsea (TSS) é subdividido por sua vez em cinco subPSL's. Apesar de unidos por um mesmo propósito, cada subsetor é responsável por uma função específica na operação de teste.

São eles: Test Tools (TT), Subsea Safety Systems (SSS), Data Acquisition Services (DAS), Fluid Sampling (SAM) e Surface Well Testing (SWT). A Figura 7 ilustra essa divisão.

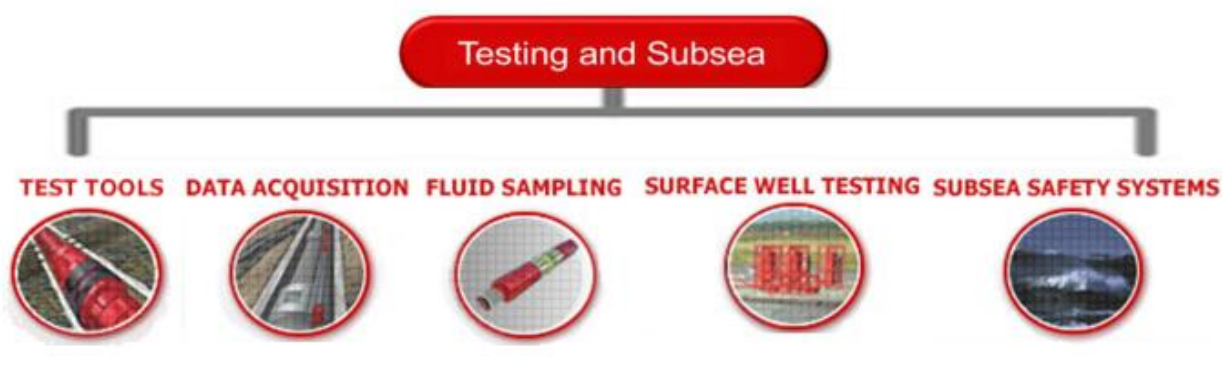


Figura 7 - Divisão do setor de TSS Halliburton

1.4.1. **Test Tools (TT)**

O subPSL chamado Test Tools é o responsável por ferramentas que ficam na coluna de fundo (BHA-Bottom hole assembly) Estas ferramentas juntamente com a coluna de produção formam a coluna de teste.

Podem-se destacar funções vitais para o teste da formação como de responsabilidade de Test Tools, por exemplo, a abertura e fechamento do poço, assentamento da coluna de teste, circulação de fluido no poço, transporte de amostradores e registradores de pressão e temperatura, compensação de variação no comprimento da coluna, teste da coluna, etc. As ferramentas de TT são importantes, pois têm a função de colocar o poço nas

suas condições estática e dinâmica para que seja possível a análise destes dados, ou seja, o teste da formação.

E é este teste que dará condições para os clientes tomarem decisões importantes em relação à exploração do reservatório. Os dados resultantes da operação de teste são fundamentais para tomadas de decisões financeiras, de engenharia, de logística, entre outras.

1.4.1.1. Ferramentas principais

As ferramentas de Testing Tools são, na sua maioria, do tipo APR (*Annulus Pressure Responsive*), ou seja, são atuadas pela pressão hidrostática do espaço anular. Algumas possuem disco de ruptura que, ao atingir uma determinada pressão, se rompe, acionando o mecanismo de funcionamento da ferramenta.

A Figura 8 ao lado mostra uma coluna de teste básica com as principais ferramentas de fundo que serão detalhadas na sequência.

Testadora de tubulação: válvula responsável por testar a integridade da vedação da coluna após a descida.

Junta telescópica: Junta móvel com função de compensar variações do comprimento da coluna devido à variação de temperatura.

Porta-registradores: tubo com *slots* para registradores de pressão e temperatura.

Válvula de circulação secundária.

Válvula de circulação primária.

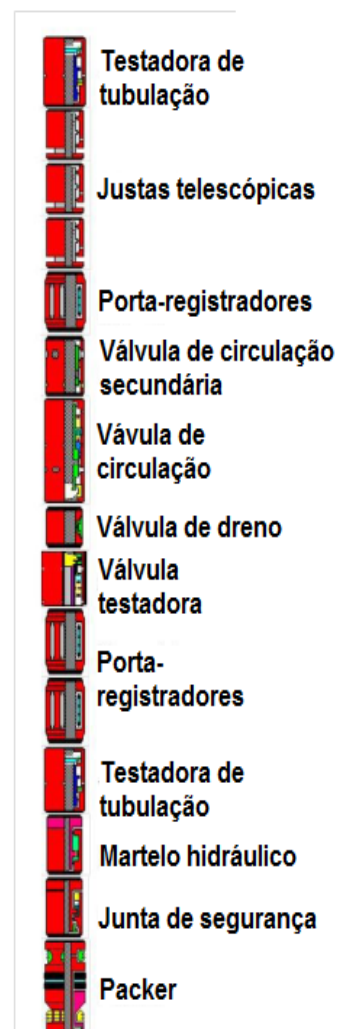


Figura 8 - Coluna básica de testes

Válvula de dreno: Válvula de dreno para uso no caso de pressão armazenada na coluna.

Válvula testadora: Válvula de fechamento de fundo responsável por criar os períodos de estática e dinâmica no poço.

Martelo hidráulico: força retirada do Packer.

Junta de segurança: para retirada da coluna acima no caso de Packer preso.

Packer: Isola o espaço anular da formação abaixo através de elastômero de vedação.

1.4.2. Subsea Safety Systems (SSS)

O subPSL de Subsea é responsável pela segurança nas operações de teste. As três principais ferramentas deste subsetor são posicionadas acima do leito marinho, sendo a AST (árvore submarina de teste) e a válvula retentora no BOP e a válvula lubrificadora a cerca de 40m abaixo do piso da plataforma.

As ferramentas possuem a função básica de permitir a vedação interna da coluna para desconexão em caso de emergência e posterior reconexão como ilustrado na Figura 9.

Tendo em mente as funções básicas deste setor, pode-se observar a grande importância de SS para a segurança da operação de TFR e da integridade da sonda.

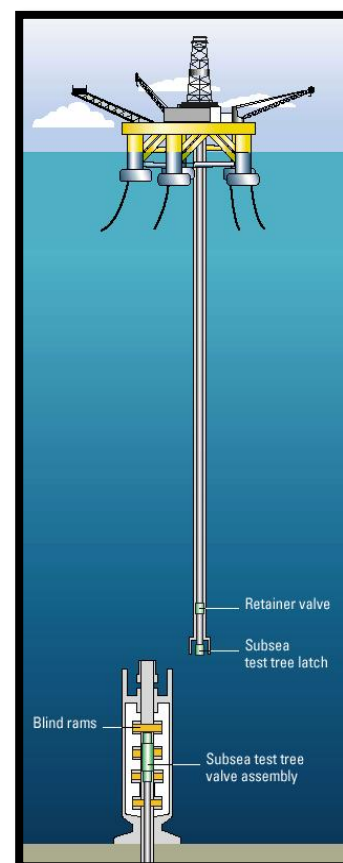


Figura 9 - Esquema de desconexão da coluna de teste

1.4.2.1. Ferramentas principais

As válvulas do BOP fazem isolamento do fluxo e a vedação interna da coluna, no sentido cima para baixo (árvore submarina de teste) e baixo para cima (válvula retentora).

Existe uma terceira válvula destinada ao teste da integridade da coluna e de equipamentos de superfície, garantido a segurança da operação.

1.4.3. Data Acquisition Service (DAS)

A aquisição de dados no processo de teste de reservatórios é de grande importância uma vez que representa o “produto final” adquirido pelo cliente. Este arquivo com registros de pressão e temperatura terá grande valia uma vez que poderá determinar detalhes cruciais do futuro da exploração do reservatório. E é nesse contexto que o setor de DAS executa seu serviço, buscando dados confiáveis no menor tempo hábil, se possível em tempo real.

E para se buscar estes dados de superfície e fundo, são utilizados equipamentos específicos para registro dos dados, os registradores de memória eletrônicos e sensores instalados na planta de superfície.

1.4.3.1. Ferramentas principais

Registradores de memória eletrônicos: Como o nome sugere, são medidores de pressão e temperatura de fundo (Figura 10).

Sensores de superfície: Sensores de pressão, temperatura e vazão espalhados na planta de superfície com sistema supervisorio das grandezas.



Figura 10 - Registradores eletrônicos de memória

1.4.4. **Fluid Sampling (SAM)**

O subPSL de *Sampling* tem por objetivo coletar fluidos nas condições de reservatório ou amostras de óleo e gás em superfície. A análise destes fluidos coletados é a chave para aperfeiçoar a rentabilidade de um campo de petróleo ou gás e caracterizar as viabilidades econômicas de se produzi-lo.

A Halliburton é especializada em amostragem de fluidos do reservatório, e os serviços de análise de laboratório, denominadas genericamente análises PVT (Pressão, Volume e Temperatura), fornecem propriedades dos fluidos, tais como: fator volume de formação; viscosidade; compressibilidade; razão de solubilidade; pressão de saturação.

1.4.4.1. Ferramentas principais

A principal ferramenta é o amostrador de fundo, que descido ao fundo em porta-amostradores fazem amostragens representativas de fluido de fundo mantendo a pressão de reservatório através de um sistema de pressurização com nitrogênio. Este conjunto é visto na (Figura 11).



Figura 11 - Amostradores no porta amostrador

1.4.5. Surface Well Testing

Surface Well Testing tem por objetivo receber, na superfície, o fluido advindo do reservatório durante o teste de formação e controlar o poço para aquisição de dados de pressão, temperatura e medição de vazão de óleo, gás e/ou água (Figura 12). Para isso conta com a válvula de segurança de superfície, *choke manifold*, trocador de calor, vaso separador (trifásico ou quadrifásico), tanque atmosférico, bomba de transferência, bomba de injeção de substâncias químicas, sistema de desligamento de emergência e queimador.

1.4.5.1. Ferramentas principais

Cabeça de teste: equipamento de fechamento de emergência do poço e possui uma linha específica para “matar” o poço, caso seja necessário.

Choke Manifold: meio primário de controle do fluxo do poço através de restrição de área de fluxo.

Separador Quadrifásico: Separa o efluente do poço depois de passar pelo *choke manifold* e aquecedor, normalmente em três fases: óleo; gás e água e ainda suporta produção de sólidos.

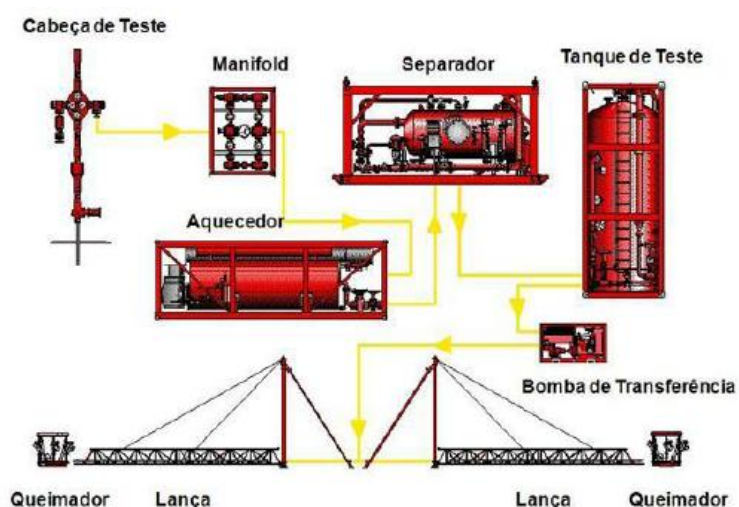


Figura 12 - Planta de superfície

1.5. PROGRAMA TRAINEE 2012.B

O projeto descrito neste documento foi realizado em paralelo com o programa trainee do setor de Testing & Subsea Halliburton.

Este programa visa preparar e buscar jovens estudantes de engenharia cursando o último período de curso para formar futuros operadores de campo. Num período de 6 meses os participantes do programa participam de treinamento de HSE (*Health, Safety & Environment*) e fazem uma rotação entre os 5 subsetores do PSL de TSS.

Durante 1 mês em cada um dos subPSL's (TT, SSS, DAS, SAM e SWT) o aprendizado é intenso e as avaliações técnicas constantes. No período de estágio foram redigidos vários relatórios técnicos e feitas algumas apresentações em cada área. Ao fim do estágio é pedido um projeto final no qual uma operação completa de teste de formação é coordenada e projetada por cada estagiário e por fim apresentada para gerentes e avaliadores da empresa.

1.6. ESTRUTURA DO DOCUMENTO

O presente documento será estruturado de forma a dar uma visão geral do ambiente em que o projeto foi desenvolvido e o contexto que motivou sua implementação.

Este primeiro capítulo enfocou na contextualização da indústria do petróleo em geral e mais especificamente a empresa e o setor onde o estágio foi realizado – a Halliburton Serviços de Energia. Ainda nesta primeira seção introdutória foi feita uma breve explanação sobre cada um dos subsetores e

seus equipamentos principais, informações importantes no entendimento do sistema e sua justificativa.

O capítulo subsequente trás seu enfoque na concepção do projeto. As justificativas e motivações da criação do sistema e por fim os objetivos da implementação de um sistema de gerenciamento de informações de ferramentas no âmbito da oficina de manutenção.

O capítulo 3 entra mais a fundo tecnicamente no desenvolvimento de criação do projeto. Definindo os requisitos de sistema, projetando o fluxo de informações e definindo o modelo conceitual.

O quarto capítulo passa pela escolha das tecnologias que foram consideradas mais apropriadas para este desenvolvimento e como a implementação foi dada. O quinto capítulo trás o detalhamento de sua utilização mostrando as interfaces criadas e a os fluxos de informação que validam os requisitos do usuário.

O capítulo 6 é destinado à implantação no ambiente de trabalho e aos resultados obtidos, e o capítulo final, de número 7, fecha o documento com as conclusões e discussões referentes aos resultados obtidos, dificuldades finais encontradas e futuro do projeto.

1.7. RELAÇÃO COM ENG. DE CONTROLE E AUTOMAÇÃO

Fazendo um paralelo com o conteúdo estudado ao longo das 9 fases de disciplinas obrigatórias e optativas cursadas, pode-se destacar que o projeto tem sua base, principalmente, em algumas das matérias estudadas, são elas:

A matéria de Metodologia para Desenvolvimento de Sistemas (DAS5312) foi importante pois trouxe o conhecimento das metodologias e linguagens aplicadas para as primeiras fases de desenvolvimento do sistema,

possibilitando o levantamento adequado dos requisitos e melhor entendimento do projeto como um todo e de suas fase de desenvolvimento e implementação.

A disciplina Fundamentos de Sistemas de Banco de Dados (INE5225) teve importância fundamental quando do projeto do banco de dados, trazendo valiosos conhecimentos utilizados tanto na parte da criação do banco de dados quanto na parte de escolha das ferramentas a serem utilizadas.

A disciplina Integração de Sistemas Corporativos (DAS5316) foi importante, com o estudo de diversos estudos de caso, foi possível analisar e prever as dificuldades que seriam encontradas ao se introduzir um novo processo em uma grande empresa como a Halliburton.

As matérias introdutórias a indústria do petróleo e gás (DAS5944 - Instrumentação Aplicada à Ind. De Petróleo e Gás, DAS5946 - Introdução à Engenharia do Petróleo e Gás, DAS5947 – Introdução ao Controle para a Indústria do Petróleo e Gás) foram muito úteis não só na familiarização e identificação do problema para realização do projeto como em todo o programa, pois trouxeram desde cedo conceitos básicos que mais tarde teriam de ser entendidos.

CAPÍTULO 2: APRESENTAÇÃO DO PROJETO

Esta segunda seção tem como objetivo contextualizar as dificuldades e necessidades encontradas que justificam e motivam o desenvolvimento do projeto. Ao final são listados os objetivos desejados para o sistema de gerenciamento de informações de ferramentas.

2.1: CONTEXTO

Para realizar os serviços prestados na área de teste e avaliação da formação, os 5 subsetores supracitados contam com ferramentas que possuem funções específicas. Uma vez que o volume de trabalho vem aumentando consideravelmente nos últimos anos, o controle das informações das ferramentas torna-se uma tarefa crítica para organização e planejamento das operações.

Assim que as ferramentas retornam de algum trabalho, é obrigatório, segundo o padrão 510 que regulamenta a manutenção, que uma folha de manutenção seja iniciada em até 5 dias. A manutenção deve ser feita seguindo-se um procedimento global e ao final das etapas de manutenção e testes, o equipamento é considerado operacional e deve ser armazenado aguardando trabalho.

Durante este processo, é de suma importância se ter conhecimento do *status* de cada equipamento (o *status* do equipamento informa a condição atual do respectivo equipamento, este pode estar em manutenção, em trabalho ou então operacional para uso, por exemplo) e também especificações de cada ferramenta e de sua manutenção. Estas informações, se acessadas de maneira rápida, podem trazer maior agilidade e eficiência aos supervisores no momento de gerenciar trabalhos.

Embora possa parecer uma atividade trivial, a sinalização de equipamentos com etiquetas coloridas para designar seu *status* e a documentação da manutenção sofre muita interferência e, atualmente, se mostra pouco confiável visto que a atualização do *status* nem sempre é feita, as informações de cada equipamento encontram-se em pastas de difícil acesso e os relatórios de manutenção devem ser impressos e armazenados a cada abertura de manutenção, gerando incertezas e dualidades.

Um esquema do fluxo de manutenção de registradores de memória de DAS é mostrado abaixo na Figura 13.

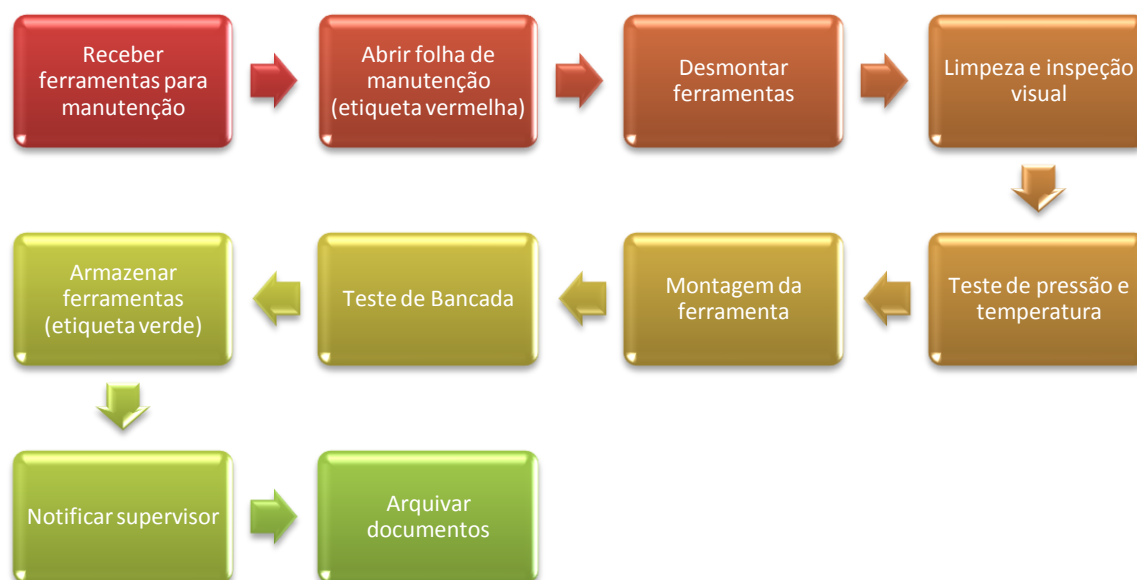


Figura 13 - Fluxograma da manutenção de registradores de memória de DAS

2.2: PADRÃO 510

O padrão de manutenção 510 é um padrão global mínimo para a manutenção de equipamentos de todas as áreas de TSS. O sistema Halliburton dá acesso aos cinco documentos que dão diretrizes para a manutenção dos equipamentos dos 5 subsetores: TT-510, SSS-510, DAS-510, SAM-510 e SWT-510. Estes documentos são muito importantes porque definem um padrão

mínimo seguido no mundo todo de como questões referentes à organização e conservação das ferramentas devem ser procedidas. As exigências abrangidas no documento são as mínimas para as atividades das oficinas ou laboratórios. Muitas vezes adotam-se prazos menores e exigências mais rígidas a fim de tornar o serviço mais qualificado. Porém, nunca se pode proceder com padrões de qualidade menos rígidos que os especificados no documento.

O documento aborda e regulamenta diferentes temas relacionados desde a burocracia envolvendo compra ou transporte de ferramentas até a disponibilização visual de informações das ferramentas.

Um trecho do padrão mínimo global de manutenção de *Test Tools* (TT-510) é dado a seguir:

Um inventário completo do status das ferramentas será visível por meio de um "SAP Equipment Status Board", planilha de Excel ou método semelhante que registre as informações seguintes:

- *Descrição do conjunto de ferramenta*
- *Número do SAP referente ao equipamento*
- *Tipo de O-ring*
- *Conexões*
- *Status da Ferramenta – ex., em um Trabalho, na Oficina (Status Verde), na Oficina (Status Vermelho)*
- *Nome do Cliente/Plataforma/País - se em um trabalho, ou designado a um trabalho*
- *Data de Mobilização*
- *Data que a última manutenção foi realizada*
- *Data de execução da última MPI das conexões*
- *Data de execução da última MPI completa*
- *Comentários*

2.3: MOTIVAÇÃO/JUSTIFICATIVA

Após algum tempo de trabalho nas oficinas e laboratórios de manutenção e ter conhecimento destes padrões mínimos regulamentadores citados acima foi possível visualizar necessidades e dificuldades enfrentadas com os métodos e procedimentos atuais.

O primeiro ponto observado foi uma não conformidade com o padrão 510. Em uma das oficinas de manutenção, por exemplo, é exposto um quadro com fichas representando as ferramentas. Dependendo do *status* de cada ferramenta a ficha é colocada com uma cor aparente ou então colocada em uma coluna destinada às operações, como é visto na Figura 14.



Figura 14 – Quadro de *status* de equipamentos

O exemplo acima se repete em outros setores de diferentes formas, tendo em comum a desatualização destas informações. Assim surgiu a primeira sugestão de digitalizar e tornar este sistema informativo de *status* das ferramentas mais intuitivo e acessível.

Todas as ferramentas para manutenção na oficina possuem folhas de engenharia (Figura 15) mostrando os componentes e servindo de referência para montagem e desmontagem. Muitas vezes estes documentos se encontram desatualizados e continuam sendo base de conhecimento.

Aliando este problema com a recente aquisição de aparelhos televisores espalhados pelas oficinas pensou-se na possibilidade de integrar o sistema de *status* com a disponibilização eletrônica destas folhas de engenharia

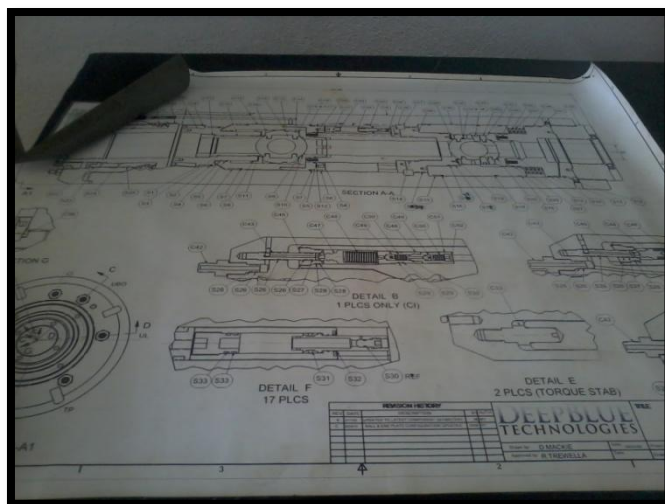


Figura 15 – Folha de engenharia de ferramenta

Atualmente um inventário completo de informações das ferramentas é feito com uma planilha de Excel. Esta planilha possui muitas redundâncias e informações que não são utilizadas e, por vezes, desatualizadas. Assim, pensou-se na possibilidade integrar as funcionalidades relacionadas à disponibilidade de informações em um único sistema.

HALLIBURTON			Situação dos Equipamentos				INFORMAÇÕES DE EMBARQUE					
			Localização País - Brasil									
			Localização País - Brasil									
Polegada	Tipo de Equipamento	Equipment Trace Number/ Tech ID	Manutenção Verde/ Vermelha	Top Conn.	Bottom Conn.	Proprietário	SITUAÇÃO	LOCAÇÃO	LOCAL E DOCUMENTO DE EMBARQUE	Mob Date	Des-Mob Date	DAT INÍ
6	SUPER SAFETY VALVE 15K	10028074	Yes	3788 CAS	3788 CAS	Macaé	Transfered	Noruega	FRPAMENTO TRANSFERIDA PARA NORUEGA			
560	SELECT TESTER VALVE 5 IN	SLCT - 1086505	Yes	3788 CAS	3788 CAS	Macaé	Offshore	Macaé	NS-21RT-3080H454	9-ago-12		3009
561	SELECT TESTER VALVE 5 IN	SLCT - 11235363	Yes	3788 CAS	3788 CAS	Macaé	Base	Macaé	Base Macaé	19-fev-12	21-jun-12	1905
562	SELECT TESTER VALVE 7 IN	SLCT - 11383951	Yes	5194 CAS	5194 CAS	Macaé	Base	Macaé	Base Macaé	5-mai-12	20-jun-12	0300
563	SELECT TESTER VALVE 5 IN	SLCT - 11384497	Yes	3788 CAS	3788 CAS	Macaé	Base	Macaé	Base Macaé	13-mai-12	28-jun-12	1905
564	SELECT TESTER VALVE 5 IN	SLCT - 11393410	Yes	3788 CAS	3788 CAS	Macaé	Base	Macaé	SS-77 RT-3080H4122	18-ago-12		2300
565	SELECT TESTER VALVE 5 IN	SLCT - 11416568	Yes	3788 CAS	3788 CAS	Macaé	Transfered	Macaé	FRPAMENTO TRANSFERIDA PARA MOSSOES			2409
566	SELECT TESTER VALVE 5 IN	SLCT - 11506091	Yes	3788 CAS	3788 CAS	Macaé	Offshore	Macaé	HRT 08-07-11			2200
567	SELECT TESTER VALVE 5 IN	SLCT - 11506094	Yes	3788 CAS	3788 CAS	Macaé	Base	Macaé	Base Macaé	25-jan-12	30-jan-12	1405
568	SELECT TESTER VALVE 5 IN	SLCT - 11716856	Yes	3788 CAS	3788 CAS	Macaé	Base	Macaé	Base Macaé	22-jul-12	14-ago-12	1905
569	SELECT TESTER VALVE 5 IN	SLCT - 11776856	Yes	3788 CAS	3788 CAS	Macaé	Base	Macaé	BASE DE MACAÉ			1905
570	SELECT TESTER VALVE 5 IN	SLCT - 11825829	Yes	3788 CAS	3788 CAS	Macaé	Base	Macaé	Base Macaé	30-jan-12	28-mar-12	2501
571	SELECT TESTER VALVE 5 IN	SLCT - 11835466	Yes	3788 CAS	3788 CAS	Macaé	Base	Macaé	Base Macaé	19-fev-12	21-06-12	0600
572	SELECT TESTER VALVE 5 IN	SLCT - 11889373	Yes	3788 CAS	3788 CAS	Macaé	Base	Macaé	Base Macaé	30-jan-12	28-mar-12	2401
573	SELECT TESTER VALVE 5 IN	SLCT - 11889373	Yes	3788 CAS	3788 CAS	Macaé	Base	Macaé	Base Macaé			2401
574	SELECT TESTER VALVE 5 IN	SLCT - 11914533	No	3788 CAS	3788 CAS	Macaé	Base	Macaé	Base Macaé	22/07/2012	21-8-12	1200
575	SELECT TESTER VALVE 5 IN	SLCT - 10316781	Yes	3788 CAS	3788 CAS	Macaé	Base	Macaé	Base Macaé			
576	SLIP JOINT ROUND MANDREL 5 IN	SLJ - 10085625	No	3788 CAS	3788 CAS	Macaé	Base	Macaé	Base Macaé	22-jul-12	21-9-12	1200
577	SLIP JOINT ROUND MANDREL 5 IN	SLJ - 12083319	Yes	3788 CAS	3788 CAS	Macaé	Base	Macaé				
578	SLIP JOINT ROUND MANDREL 5 IN	SLJ - 12083320	Yes	3788 CAS	3788 CAS	Macaé	Base	Macaé				
579	SLIP JOINT ROUND MANDREL 5 IN	SLJ - 10229030	Yes	3788 CAS	3788 CAS	Macaé	Base	Macaé	Base de Macaé	19-fev-12	21-jun-12	3000
580	SLIP JOINT ROUND MANDREL 5 IN	SLJ - 10229031	Yes	3788 CAS	3788 CAS	Macaé	Base	Macaé	Base Macaé	21-mai-12	28-jun-12	2600
581	SLIP JOINT	SLJ - 10229032	Yes	3788 CAS	3788 CAS	Macaé	Offshore	Macaé	NS-21RT-3080H454	09/08/2012		
582	SLIP JOINT ROUND MANDREL 5 IN	SLJ - 10253956	Yes	3788 CAS	3788 CAS	Macaé	Offshore	Macaé	SS-77 Danie_025093	22-jul-12		1905
583	SLIP JOINT ROUND MANDREL 5 IN	SLJ - 12082438	Yes	3788 CAS	3788 CAS	Macaé	Base	Macaé				2401

Figura 16 – Tabela Excel de controle de informações de ferramentas

Outro problema observado foi o procedimento de documentação da manutenção. Todo o procedimento é documentado através de uma folha de manutenção impressa. Apesar de estas folhas possuírem locais específicos de armazenamento de acordo com o andamento da manutenção, muitas vezes

ocorrem perdas ou dificuldades em localizar determinada folha, resultando em retrabalho e, conseqüentemente, perda de tempo.

Ao conversar sobre necessidades de melhorias com os gestores responsáveis pelo setor de qualidade, foi levantada uma necessidade de quantificar a taxa de retrabalho e de manutenções preventivas/corretivas.

Aliando os dois últimos problemas constatados, pensou-se na possibilidade de relacionar as ferramentas do sistema com manutenções, digitalizando este processo e possibilitando a geração de relatórios com informações pontuais.

2.4: OBJETIVOS

Uma vez que as dificuldades apresentadas no processo atual foram visualizadas e o desenvolvimento de um sistema de gerenciamento de informações de ferramentas foi justificado, começou-se a pensar nos objetivos do desenvolvimento e implementação do projeto.

O objetivo geral é criar um sistema que possa unificar as informações das ferramentas de forma digital e mostrá-las de forma ágil e intuitiva. Dentro dessas informações deve estar em destaque o *status* atual da ferramenta.

O sistema também deve gerar formulários de manutenção e salvá-los conforme o andamento das etapas da manutenção. Deve ser possível salvar e imprimir este formulário para armazenamento em arquivo físico e digital.

Outro objetivo geral é gerar relatórios de acordo com as informações alimentadas no sistema, de forma a possibilitar que sejam tomadas decisões

baseando-se nas estatísticas de *status* de ferramentas, andamento da manutenção e taxa de retrabalho, por exemplo.

CAPÍTULO 3: CRIAÇÃO E CONCEPÇÃO DO SISTEMA

Este terceiro capítulo objetiva ilustrar como o problema inicial foi abordado para que a implementação do *software* pudesse ser feita. Aqui será feita uma breve abordagem sobre o Processo Unificado (PU), metodologia utilizada para organização e definição de requisitos no que se refere à suas 4 etapas que serão expostas a seguir.

Após uma breve contextualização, é detalhado o processo de concepção e elaboração do sistema seguindo a metodologia anteriormente citada. O projeto do banco de dados também é feito ao final deste capítulo.

3.1: PROCESSO UNIFICADO DE DESENVOLVIMENTO DE SOFTWARE (PU)

O Processo Unificado foi utilizado com o objetivo de transformar requisitos de usuário em uma ideia concreta de sistema que pudesse ser aplicado e implementado na realidade, suprimindo os requisitos de forma satisfatória.

Como explica (SCOTT, 2002), o Processo Unificado faz uso extensivo da *Unified Modeling Language* (UML). No núcleo da UML, está o modelo, o qual no contexto de um processo de desenvolvimento de *software* é uma simplificação da realidade que ajuda a entender alguns processos complexos inerentes ao *software*.

O PU foi utilizado com o intuito de auxiliar no desenvolvimento do sistema de gerenciamento de informações de ferramenta devido à sua capacidade de organização e de tornar o processo de criação mais intuitivo, incremental e iterativo. Além de tornar o desenvolvimento mais ágil prevenindo e antecipando-se a possíveis erros.

A vida de um sistema de *software* pode ser representada por uma série de ciclos que são separados por marcos importantes onde os tomadores de decisão podem inferir sobre o prosseguimento do projeto e as novas metas a serem cumpridas. As principais fases do Processo Unificado são detalhadas a seguir e mostradas na Figura 17.

Concepção: esta primeira fase tem como objetivo estabelecer a viabilidade da proposta do sistema. Dentre as atividades inerentes a esta fase estão a definição do escopo do projeto bem como de um esboço para o sistema. Ainda nesta fase é feita uma identificação dos riscos e das viabilidades do projeto.

Elaboração: Nesta fase são capturados os requisitos funcionais em sua maioria. Além disto, é feita uma análise final da viabilidade do projeto e preparado um plano com detalhes para orientar a próxima fase (construção). Em suma, esta fase é dedicada à análise e projeto do sistema.

Construção: esta fase acontece iterativamente com a fase de elaboração, ou seja, o sistema é construído de forma a revisar constantemente a viabilidade do sistema. O principal objetivo desta fase é construir e testar um sistema capaz de operar ainda que de forma experimental.

Transição: O objetivo principal desta fase é entregar ao cliente um sistema completamente operacional.

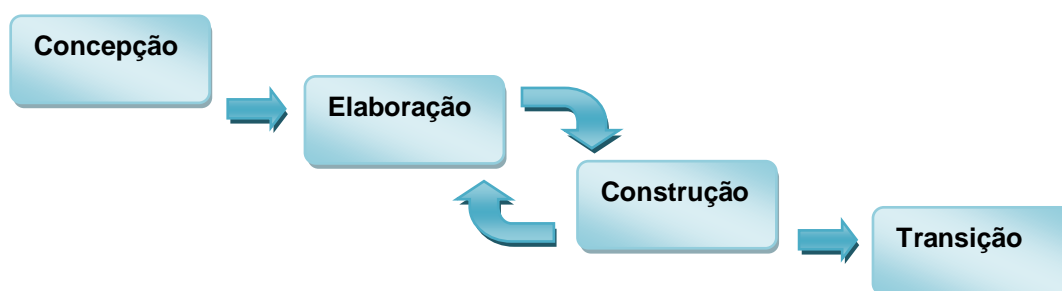


Figura 17 – 4 Fases do Processo Unificado

3.2: UNIFIED MODELING LANGUAGE (UML)

A UML é uma linguagem de modelagem que auxilia a visualização de forma gráfica do sistema a ser desenvolvido em diagramas padronizados.

Em seu livro, (FOWLER, 2004), define de maneira um tanto quanto simplificada a UML como uma família de notações gráficas, apoiada por um metamodelo único, que ajuda na descrição e no projeto de sistemas de *software*, particularmente daqueles construídos utilizando o estilo orientado a objetos (OO).

É comum utilizar-se da UML para fazer esboços do sistema. No desenvolvimento, o esboço UML é feito anteriormente ao código e visa, sobretudo, transmitir ideias e alternativas sobre o que se está prestes a fazer.

A UML descreve 13 tipos de diagramas oficiais mostrados na Figura 18 são e listados abaixo:

- Diagrama de classes
- Diagrama de objetos
- Diagrama de componentes
- Diagrama de estruturas compostas
- Diagrama de casos de uso
- Diagrama de sequências
- Diagrama de comunicação
- Diagrama de máquinas de estados
- Diagrama de atividades
- Diagrama de implementação
- Diagrama de pacote
- Diagrama de temporização
- Diagrama de visão geral da interação

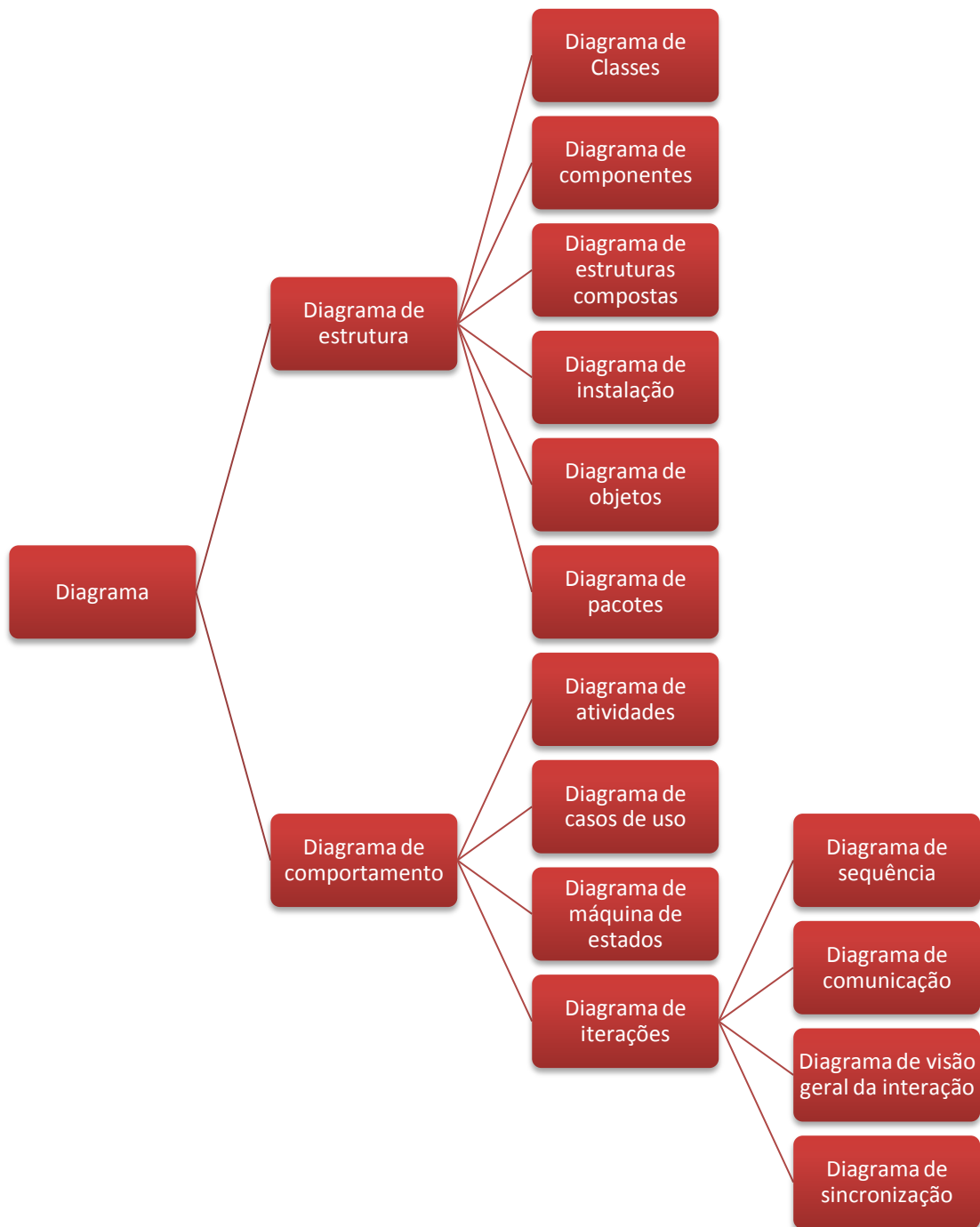


Figura 18 – Classificação dos tipos de diagrama UML

3.3: FASE 1: CONCEPÇÃO

Em seu livro, Larman (LARMAN, 2005, p. 73) define a fase de concepção em apenas uma frase: “Conceber o escopo do produto, a visão e o caso de negócio”.

Nesta fase os esforços foram focados em conversas e reuniões com supervisores a fim de entender as necessidades e propor soluções. Esta fase da concepção deve ser relativamente breve e visa ter uma ideia do problema e a certeza de que o mesmo é viável. É importante levantar e organizar os requisitos do sistema.

Na concepção do sistema foram levantados requisitos funcionais (comportamentais) e não funcionais (os demais) e alguns artefatos do PU foram escolhidos para organização destes requisitos, são eles:

Visão: resume requisitos de alto nível que são detalhados no modelo de casos de uso e especificação suplementar, e resume o caso de negócios do projeto. É um documento curto contendo uma visão geral para entendimento rápido das ideias do projeto. A visão geral do projeto foi utilizada para que pudessem ser obtidos os requisitos do sistema que, posteriormente, seriam organizados utilizando o modelo de casos de uso.

Modelo de Casos de Uso: Um conjunto de cenários típicos do uso de um sistema. Eles são principalmente para requisitos funcionais (comportamentais).

3.3.1. Visão geral e levantamento dos requisitos do sistema

Após algum tempo de trabalho na oficina e reuniões com o supervisor responsável e alguns engenheiros, chegou-se à visão geral do sistema descrita abaixo no Quadro 1:

<p style="text-align: center;">Visão geral do sistema de gerenciamento de informações de ferramentas para o setor de Testing & Subsea.</p> <p>O projeto proposto prevê criação de um sistema que reúne a informação do <i>status</i> das ferramentas, juntamente com outras informações específicas de forma que o acesso a estas informações seja ágil e a informação seja confiável. Substituindo os atuais métodos físicos de <i>Tools Status</i>.</p> <p>Ainda espera-se que a manutenção das ferramentas possa ser documentada através do sistema, evitando o retrabalho e trazendo de forma confiável um relatório geral de manutenção para que os supervisores possam tomar decisões estratégicas com maior certeza e agilidade.</p>

Quadro 1 – Visão geral do sistema

Uma vez que a visão geral do sistema foi definida, pode-se levantar e definir os requisitos. Estes podem ser divididos como funcionais, representando as funções básicas do sistema, ou então do tipo não funcionais, apresentando funções secundárias vinculadas aos requisitos funcionais e suas restrições.

Os requisitos funcionais serão nomeados como F1, F2, até Fn. Os requisitos não funcionais tem sua nomeação de acordo com o requisito funcional a que estão relacionados, por exemplo, F1.1, F1.2, F2.1, etc. Ainda podem ser classificados de acordo com a especificação em obrigatórios ou desejáveis; de acordo com o tempo de utilização, se são permanentes ou transitórios; e de acordo com a sua categoria, em requisitos de segurança, desempenho, usabilidade, confiabilidade, etc.

Nesta fase foram levantados os requisitos funcionais F1, F2, F3, F4 e F5:

- F1: Mostrar *Status* de ferramentas (Quadro 2)
- F2: Mudar *Status* de ferramentas (Quadro 3)
- F3: Gerenciamento do sistema (Quadro 4)
- F4: Gerenciar Manutenção (Quadro 5)
- F5: Gerar Relatórios (Quadro 6)

Os quadros abaixo registram os principais requisitos funcionais analisados, bem como seus requisitos não funcionais relacionados:

F1 Mostrar informações de ferramentas		Oculto ()		
Descrição: O sistema deve permitir a qualquer usuário visualizar informações de <i>Status</i> das ferramentas e as demais informações presentes.				
Requisitos Não Funcionais				
Nome	Restrição	Categoria	Classe	Permanente
NF 1.1 Apresentação	As informações devem ser dispostas de maneira intuitiva e de fácil acesso.	Interface	Obrigatório	(X)
NF 1.2 Distinção por cores	A informação de <i>status</i> da ferramenta deve ser mostrada em cores diferente a fim de facilitar a visualização.	Interface	Obrigatório	(X)
NF 1.3 Agrupamento	Deve ser possível agrupar grupos de ferramentas com mesmo <i>status</i> .	Usabilidade	Desejável	(X)

Quadro 2 – Requisito Funcional F1: Mostrar informações de ferramentas

F2 Mudar <i>Status</i> de ferramentas		Oculto ()		
Descrição: O sistema deve permitir a troca do <i>Status</i> das ferramentas.				
Requisitos Não Funcionais				
Nome	Restrição	Categoria	Classe	Permanente
NF 2.1 Autenticação	Apenas usuários autenticados podem mudar o <i>status</i> de ferramentas.	Segurança	Obrigatório	(X)
NF 2.2 Troca automática	Ao se criar uma manutenção para a ferramenta o seu <i>status</i> deve ser trocado automaticamente.	Especificação	Desejável	(X)
NF 2.3 Simplicidade	O processo de troca de <i>status</i> deve ser rápido e direto.	Usabilidade	Desejável	(X)

Quadro 3 – Requisito Funcional F2: Mudar *Status* de ferramentas

F3 Gerenciamento do sistema		Oculto ()		
Descrição: O sistema deve disponibilizar uma interface para que seja possível a inclusão, edição e exclusão de ferramentas e usuários.				
Requisitos Não Funcionais				
Nome	Restrição	Categoria	Classe	Permanente
NF 3.1 Interface diferenciada	O sistema deve disponibilizar um menu de gerenciamento específico para o nível do usuário.	Interface	Obrigatório	(X)
NF 3.2 Autenticação - Admin	Apenas administradores perante autenticação podem incluir, excluir e editar ferramentas e outros usuários.	Segurança	Obrigatório	(X)
NF 3.3 Adição de ferramenta	Ferramentas devem ser adicionadas sendo obrigatórios a inclusão de um número de equipamento único, tipo e status.	Especificação / Segurança	Obrigatório	(X)
NF 3.4 Adição de usuários	Usuários devem ser adicionados sendo obrigatórios a inclusão de um número de identificação único, nome, senha e nível.	Especificação / Segurança	Obrigatório	(X)

Quadro 4 – Requisito Funcional F3: Gerenciamento do Sistema

F4 Gerenciar Manutenção		Oculto ()		
Descrição: O sistema deve disponibilizar a criação, edição e exclusão de manutenções relacionadas às ferramentas.				
Requisitos Não Funcionais				
Nome	Restrição	Categoria	Classe	Permanente
NF 4.1 Campos específicos	A manutenção deve possuir campos específicos obtidos a partir do portal Halliburton para geração de formulário de manutenção.	Especificação	Obrigatório	(X)
NF 4.2 Autenticação - Usuário	Qualquer usuário perante autenticação pode incluir, excluir e editar manutenções.	Segurança	Obrigatório	(X)
NF 4.3 Preenchimento parcial	Deve ser possível fazer o preenchimento parcial da manutenção, possibilitando a diferentes usuários participar da mesma manutenção.	Especificação	Obrigatório	(X)
NF 4.4 Geração de documento PDF	O sistema deve prover um documento PDF da manutenção com as informações alimentadas pelo usuário a fim de possibilitar a impressão e o armazenamento.	Especificação	Obrigatório	(X)

Quadro 5 – Requisito Funcional F4: Gerenciar Manutenção

F5 Gerar Relatórios		Oculto ()		
Descrição: O sistema deve gerar relatórios com base nas informações de <i>status</i> e manutenções de ferramentas.				
Requisitos Não Funcionais				
Nome	Restrição	Categoria	Classe	Permanente
NF 5.1 Acesso restrito	Apenas usuários administradores do sistema têm permissão para gerar relatórios.	Especificação	Obrigatório	(X)
NF 5.2 Relatório geral de manutenção	O sistema deve gerar um relatório listando todas as ferramentas cadastradas, seu <i>status</i> e sua etapa de manutenção.	Segurança	Obrigatório	(X)
NF 5.3 Relatório de <i>status</i>	O sistema deve gerar um relatório que contabilize as ferramentas e seus <i>status</i> .	Especificação	Obrigatório	(X)
NF 5.4 Geração de relatório PDF	O sistema deve possibilitar que todos os relatórios possam ser gerados em formato PDF.	Especificação	Obrigatório	(X)

Quadro 6 – Requisito Funcional F5: Gerar Relatórios

3.3.2. Casos de uso e organização dos requisitos

Uma vez levantados os requisitos do sistema, pode-se partir para a fase de organização destes requisitos. Os requisitos podem ser organizados em diferentes grupos.

- **Casos de Uso:** Envolvem os principais processos do sistema e modificam os estados do sistema;

- **Consultas:** Envolvem pesquisas dos dados armazenados e relatórios impressos. Não mudam informações do sistema.

O **Erro! Fonte de referência não encontrada.** demonstra formalmente os casos de uso do sistema.

Nome	Descrição	Categoria
Abrir Manutenção	O usuário cria uma nova manutenção relacionada a uma ferramenta existente.	F4
Editar Manutenção	O usuário edita campos de uma manutenção existente.	F4
Excluir Manutenção	O usuário exclui e uma manutenção existente.	F4
Incluir Ferramenta	O usuário cria uma nova ferramenta.	F3
Editar Ferramenta	O usuário edita campos de uma ferramenta existente.	F3
Excluir Ferramenta	O usuário exclui uma ferramenta existente.	F3
Mudar Status de Ferramenta	O usuário altera o <i>Status</i> de uma ferramenta existente.	F2
Adicionar Usuários	O usuário, com permissão, cria um novo usuário.	F3
Editar Usuários	O usuário, com permissão, edita um usuário existente.	F3
Excluir Usuários	O usuário, com permissão, exclui um usuário existente.	F3
Gerar Relatórios	O usuário poderá gerar relatórios das ferramentas através de solicitação no sistema	F5
Login	Usuário faz autenticação com “username” e “senha” e tem permissão para acessar campos restritos.	F2,F3,F4,F5
Logout	Usuário autenticado sai do sistema.	F2,F3,F4,F5

Quadro 7 – Casos de uso

Os casos de uso mostrados abaixo na Figura 19 mostram a relação dos atores com os processos que de alguma forma modificam o estado do sistema. Os atores neste caso são os usuários: “Usuário Não Autenticado”, “Usuário Autenticado” e “Administrador”. Nota-se como o nível dos usuários influencia na relação com os processos.

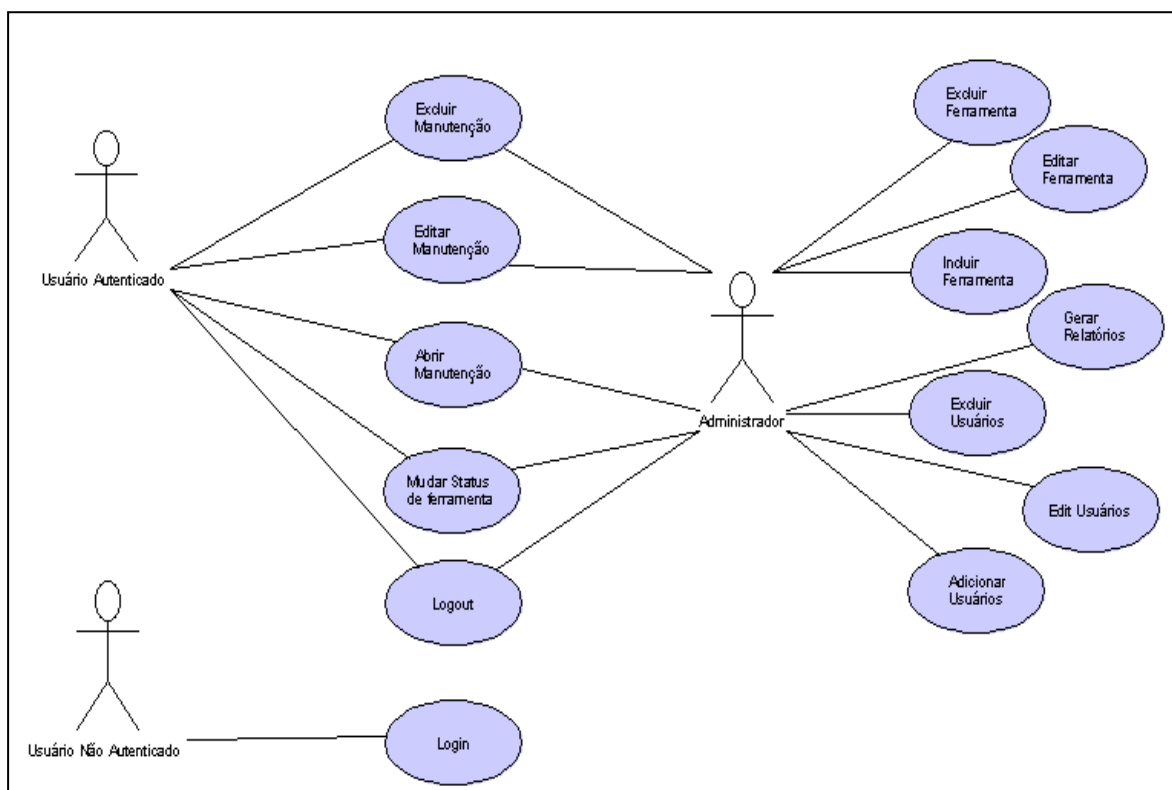


Figura 19 – Diagrama de Casos de Uso

As consultas do sistema estão relacionadas às operações dos usuários que não modificam de alguma forma a estrutura do sistema. Cada consulta está relacionada a algum requisito de usuário. Em suma, cada operação desta natureza é destinada à visualização (consulta) de alguma informação do sistema.

Nome	Descrição	Categoria
Lista de ferramentas.	Listagem das ferramentas cadastradas. Podem ser ordenadas a partir dos atributos "Equipment Number", "Tipo", "Status" e "Grupo".	F1
Visualização de informações de ferramenta	Visualizar informações específicas da ferramenta selecionada.	F1
Lista de manutenções	Listagem das manutenções já realizadas. Podem ser ordenadas a partir dos atributos "Iniciada Por" e "Data".	F4
Visualização de informações de manutenção.	Visualizar informações específicas de manutenção selecionada.	F4
Lista de usuários cadastrados	Listagem das manutenções já realizadas. Podem ser ordenadas a partir dos atributos "Nome" e "Username". Apenas feita por usuários administradores.	F3
Visualização de dados de usuário	Visualizar informações específicas do usuário selecionado. Apenas feito por usuários administradores.	F3
Visualização de relatórios	Visualizar relatórios gerados pelo sistema. Apenas feito por usuários administradores.	F5

Quadro 8 – Consultas do Sistema

3.4: FASE 2: ELABORAÇÃO

Os objetivos principais da fase de Elaboração são capturar a maioria dos requisitos funcionais e estabelecer uma base arquitetônica sólida para o desenvolvimento futuro, isto é, criar uma base estável para o esforço da fase de construção, que será abordada posteriormente. A etapa se desenvolve a partir do exame e do detalhamento dos requisitos mais significativos (SCOTT, 2002).

É nesta fase que se inicia a parte iterativa do Processo Unificado. Esta iteração consiste na liberdade em, sempre que necessário, retornar para a fase de elaboração caso já se esteja na fase de construção (próxima fase), isto acontece para que se obtenham os resultados conforme as especificações.

Nessa etapa do projeto foram gerados alguns documentos a fim de aumentar o entendimento para as próximas etapas. São eles:

- Expansão dos casos de uso;
- Determinação dos eventos do sistema (operações e consultas);
- Modelo de sistema em WireFrame;

3.4.1. Expansão dos casos de uso

Após a descoberta dos casos de uso, já na fase de elaboração, são feitas as expansões dos casos de uso, o que corresponde ao aprofundamento da análise de requisitos e consiste no detalhamento do processo de negócio (WAZLAWICK, 2004). O objetivo desse documento é especificar uma sequência de passos de tudo o que se faz no processo.

Como explica (WAZLAWICK, 2004), nesta etapa deve-se descrever passo a passo como o caso de uso ocorre, como é a interação entre os usuários e o sistema. A expansão dos casos de uso começa criando o Fluxo

Principal que é conhecido como o ideal para o processo, no qual tudo dá certo, porque não são descritas as exceções. Após, são descritas as interações para os casos nos quais ocorre alguma exceção, os chamados Fluxos Alternativos, que são os procedimentos para contornar os problemas do Fluxo Principal. Para cada caso de uso, além do Fluxo Principal e Alternativo, devem-se indicar também as pré-condições e pós-condições de execução.

As pré-condições são condições necessárias que devem ser satisfeitas antes da execução do fluxo principal de um caso de uso e as pós-condições representam os resultados de um caso de uso após sua execução.

Para o caso de uso de “Abrir Manutenção”, foi feita a expansão do caso de uso e esta é mostrada abaixo.

Caso de Uso: Abrir Manutenção
Pré-condições: O usuário deve estar cadastrado no sistema e ter feito <i>login</i> . A ferramenta cuja manutenção será aberta deve existir no sistema.
Pós-condições: Criada manutenção relacionada à ferramenta. Salva para posterior visualização.
Requisitos correlacionados: F4
Fluxo principal: <ol style="list-style-type: none"> 1) O usuário seleciona a opção de “Abrir nova manutenção”. [EV] 2) O usuário informa a ferramenta da qual a manutenção se refere. [EV] 3) O usuário preenche os campos referentes ao nome de quem executa a manutenção e data. [EV] 4) O usuário informa dados específicos da ferramenta e do último trabalho. [EV] 5) O usuário preenche os campos das etapas da manutenção executadas com o nome de quem as executou e a data da execução. [EV] 6) Usuário finaliza e salva manutenção. [EV] 7) O sistema informa que manutenção foi incluída com sucesso. [RS]
Tratamento de exceção:

2a. O usuário verifica o disparo de algum alarme crítico no sistema

- 2.a.1) O usuário não seleciona ferramenta para manutenção. [EV]
- 2.a.2) Sistema informa erro.[RS]
- 2.a.3) O usuário informa a ferramenta da qual a manutenção se refere. [EV]
- 2.a.4) Usuário finaliza e salva manutenção. [EV]
- 2.a.5) O sistema informa que manutenção foi criada com sucesso. [RS]

Para este caso, as pré-condições estão relacionadas à existência de cadastro do usuário que está abrindo uma nova manutenção e também se a ferramenta em questão para manutenção foi devidamente incluída no sistema. A pós-condição para este caso é da manutenção, com todos os campos, ser salva para posterior visualização no sistema.

No detalhamento gráfico anterior são utilizadas as siglas *EV* e *RS*. *EV* significa que o passo é relativo a eventos do sistema e a sigla *RS* para as respostas do sistema. Como estas etapas de elaboração visam um entendimento do problema para que a etapa de construção seja facilitada, não foram expandidos os casos de uso referentes à criação, edição e exclusão uma vez que estes não influenciam no entendimento do sistema como um todo.

3.4.2. Operações e consultas do sistema

As operações do sistema são definidas como ações tomadas decorrentes de um evento gerado pelo usuário. Geralmente ocasionam alterações na informação que o sistema gerencia, como bancos de dados. Já as consultas não alteram a informação gerenciada pelo sistema, apenas apresentam o que se encontra armazenado (WAZLAWICK, 2004).

A partir desta definição, para o projeto em questão pode-se exemplificar como operação de sistema a alteração do *status* de uma ferramenta e um exemplo de consulta do sistema a simples visualização das informações de uma ferramenta.

As operações e consultas representam as relações dinâmicas do software e devem ser modeladas de modo a ilustrar o fluxo de informações entre os objetos do sistema e os atores que de alguma forma com eles interagem.

Normalmente, uma ferramenta utilizada para representação dos mesmos são os diagramas de sequência da UML. O diagrama de sequência é um diagrama que representa, para um caso de uso em particular, as informações passadas entre objetos e eventos gerados por atores externos. O diagrama de sequência, mesmo para programas de grande porte, consegue representar a sequência global de informação de uma forma simples, intuitiva e lógica uma vez que trata todos os sistemas como “caixas pretas” e representa atores e elementos participantes muito próximos do processo físico (LARMAN, 2005).

Para o caso de uso expandido na seção anterior (Abrir Manutenção) foi feito o diagrama de sequência mostrado a seguir na Figura 20. Os quadros amarelos simbolizam os objetos do sistema que se relacionarão entre si. O usuário relaciona-se com o sistema que por sua vez faz consultas e operação ao banco de dados. As linhas pontilhadas verticais simbolizam uma linha do tempo que dá sentido às sequências de ações.

O usuário ao solicitar uma nova manutenção recebe um formulário com os campos a serem preenchidos. Após preencher todos os dados, o usuário solicita que o formulário seja salvo. No caso de haver algum dado incompatível (ferramenta inexistente, por exemplo), o sistema retorna uma mensagem de erro indicando a informação inválida. Se todos os dados forem válidos o sistema deve adicionar as informações de manutenção no banco de dados e retornar uma mensagem de que a manutenção foi criada com sucesso.

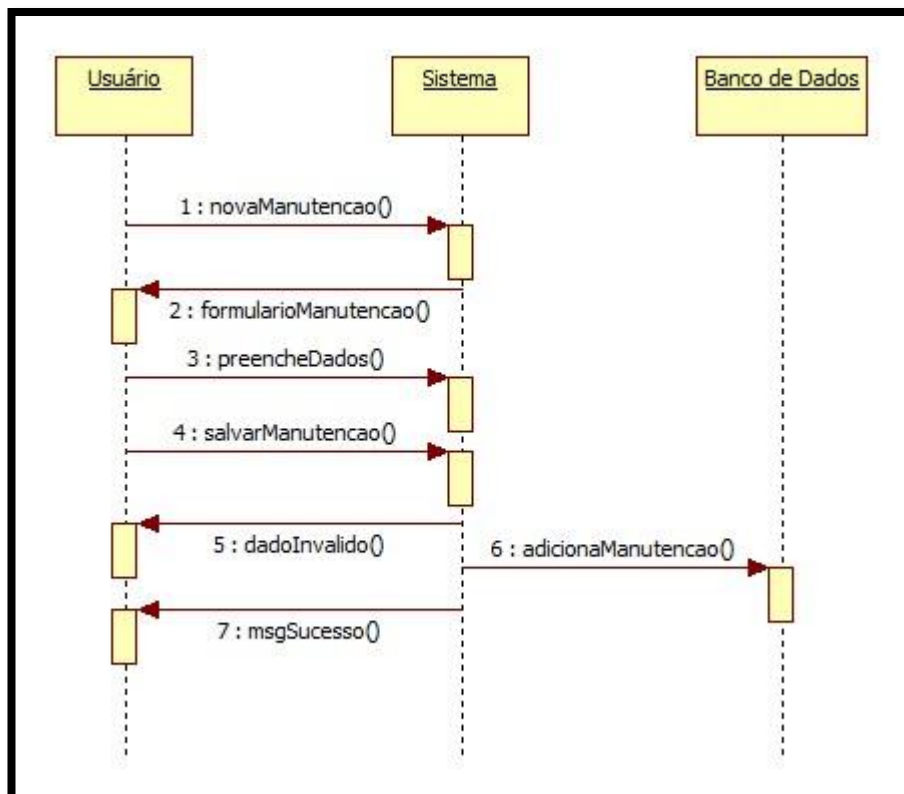


Figura 20 – Diagrama de sequência para o caso de uso “Abrir Manutenção”.

3.4.3. Modelo de sistema em *Wireframe*

O modelo em *Wireframe* é um desenho básico, como um esqueleto, que demonstra de forma direta a arquitetura de como a interface final será de acordo com as especificações relatadas e os requisitos levantados até esta etapa.

No final da etapa de elaboração do sistema, um modelo simples em *Wireframe* foi desenvolvido com auxílio do *software Balsamiq Mockups* com o objetivo de auxiliar no entendimento dos requisitos que foram recolhidos junto ao cliente com relação às funções e objetos do sistema a ser construído.

Algumas telas deste modelo são mostradas abaixo na Figura 21.

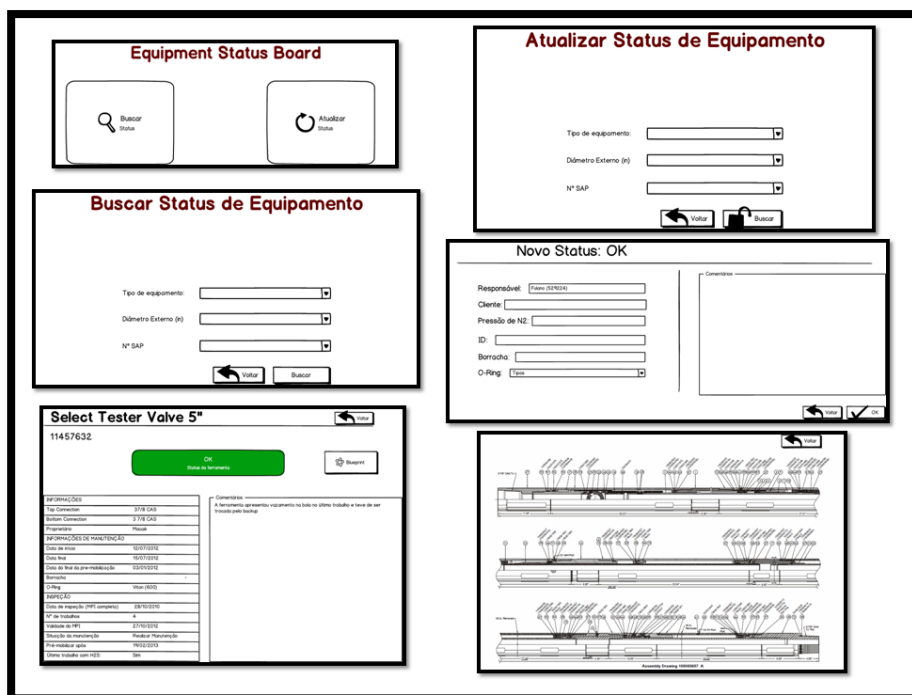


Figura 21 – Wireframes

Apesar de estas interfaces terem sido modificadas quase que totalmente no produto final devido a novas ideias e formas mais práticas tanto para o desenvolvimento quanto para usabilidade, este processo foi muito útil para a consolidação dos requisitos levantados e para que o entendimento do problema por parte do desenvolvedor e do cliente pudesse ser alinhado.

3.5: PROJETO DE BANCO DE DADOS

Para finalizar a fase de elaboração do projeto foi feito um modelo conceitual. Este modelo ilustra conceitos importantes do domínio do problema, as suas associações e atributos.

Estes conceitos são importantes, pois são o primeiro passo para se desenvolver as futuras classes e os seus atributos de software. Cada conceito ou entidade, assim como seus relacionamentos, deve aparecer direta ou indiretamente nas descrições dos casos de uso.

Uma vez que o modelo conceitual UML para o sistema proposto é muito similar ao modelo conceitual do banco de dados, optou-se pela modelagem conceitual do banco de dados devido a sua maior abrangência e detalhamento do sistema.

Assim como o projeto do *software* possui fases de desenvolvimento, o projeto do Banco de Dados também apresenta uma sequencia de etapas para ser seguida, esta sequencia é mostrada na Figura 22.



Figura 22 – Fases de projeto de BD

O projeto conceitual captura as necessidades da organização em termos de armazenamento de dados e é independente do SGBD (não trata de detalhes de implementação). Este modelo é mostrado abaixo na Figura 23.

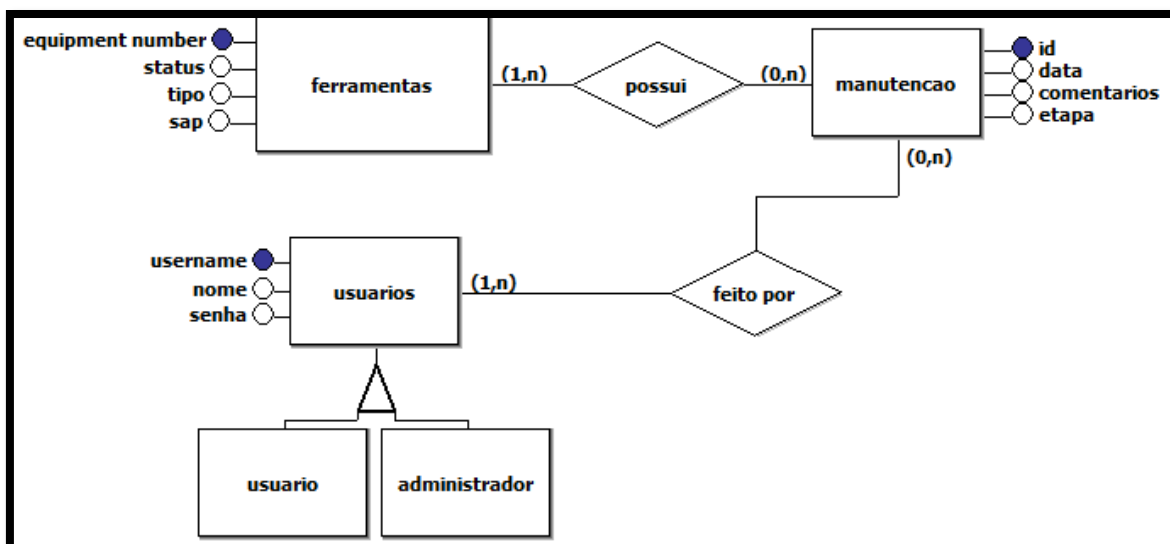


Figura 23 – Modelo Conceitual do Banco de Dados

O modelo conceitual exibido anteriormente possui três entidades principais que se relacionam entre si. A entidade “usuários” se divide em usuários comuns e administradores. Todos os usuários possuem “username” único, senha e nome do colaborador. A entidade “manutenção” possui um número de identificação (id), data, comentários e etapa de manutenção. Uma manutenção pode ser feita por vários usuários e um usuário pode participar de várias manutenções.

Por fim, a entidade “ferramenta” possui atributos relacionados às informações das ferramentas do setor. O “equipment number” é sua identificação única dada pelo sistema da empresa. O número “SAP” é relacionado ao grupo a que a ferramenta se enquadra. Ainda se tem como atributos o “tipo” e o “status” em que a ferramenta se encontra, podendo ser Operacional, em manutenção, em trabalho, etc.

Através deste modelo se tem uma visão geral de como se relacionam as entidades e do ambiente de dados. O modelo conceitual também não depende de nenhum SGBD, logo, qualquer alteração no software ou hardware, não terá efeito no nível conceitual.

Ainda cabe ressaltar que este modelo conceitual ilustra as entidades padrão do sistema com os atributos básicos. Este modelo será desenvolvido e seu projeto será mais bem compreendido nos capítulos à frente, porém o projeto para cada subPSL trará atributos extras de acordo com as ferramentas e com as informações relevantes para cada área.

CAPÍTULO 4: DESENVOLVIMENTO E IMPLEMENTAÇÃO DO SISTEMA

O capítulo anterior teve como objetivo fazer a definição do resultado final esperado para que o sistema proposto suprisse as necessidades requeridas. Como produto final das técnicas de projeto de software aplicadas anteriormente se tem os requisitos do sistema, casos de uso, diagrama de sequencia e modelo conceitual. A partir destas informações bem definidas e com o fluxo de informações em mente, pôde-se dar início às fases finais do desenvolvimento do software. Este quarto capítulo visa demonstrar os passos e as escolhas referentes à fase de construção do desenvolvimento do projeto.

Uma vez que os conceitos foram definidos, chega-se ao momento de entrar mais especificamente nas tecnologias e ferramentas que serão utilizadas para que o que já foi definido possa ser colocado em prática.

É muito importante que a partir dos requisitos levantados, características fundamentais do sistema sejam definidas para que as ferramentas e tecnologias selecionadas para a implementação possam satisfazer todas (ou em grande parte) as necessidades do desenvolvedor.

O sistema proposto deve ser de grande facilidade uma vez que seria utilizado por diferentes usuários, desde supervisores até técnicos de manutenção com pouca familiaridade com computação. Também se faz necessário o acesso às informações do sistema de diferentes locais e com computadores diferentes também.

A usabilidade intuitiva e ágil também foi um ponto muito ressaltado uma vez que o maior desafio da implementação de um sistema novo em um ambiente empresarial é a resistência à sua utilização. Desta forma, uma interface amigável e uma forma de fluxo que pudesse atender à necessidade com poucos cliques seriam imprescindíveis.

Ainda é desejado que as informações cadastradas pudessem ser guardadas de forma segura e que o acesso à informação fosse restrito aos funcionários da empresa e perante autorização.

4.1: LINGUAGEM DE PROGRAMAÇÃO

Uma vez que foi definida a necessidade de que o sistema apresente facilidade de uso, agilidade de operação e acessibilidade irrestrita de diferentes meios (desde que internos à rede da empresa), chegou-se à conclusão que o desenvolvimento de um software seria menos apropriado que uma aplicação *web* uma vez que haveria a necessidade de instalação em cada computador dedicado ao uso e esta tarefa deveria ser repetida a cada atualização do sistema.

Além do mais, uma aplicação *web* proporciona a utilização intuitiva uma vez que a grande maioria dos usuários utiliza o navegador da internet como ferramenta de trabalho e também trás grande alcance ao sistema, pois qualquer computador na rede Halliburton teria livre acesso.

Após definir que o mais apropriado para o projeto seria construí-lo como uma aplicação *web*, a linguagem a ser utilizada passou a ser o foco dos trabalhos e pesquisas.

Pesquisando linguagens populares de desenvolvimento para aplicações *web*, chegou-se a conclusão de que a melhor opção seria fazer o desenvolvimento em PHP (*Hypertext PreProcessor*) uma vez que esta oferecia vantagens de velocidade, estabilidade, eficiência e complexidade em relação à outras linguagens orientadas a objetos como *ASP*, *ColdFusion* e *Perl*.

Outro motivo que levou à escolha de PHP como linguagem de desenvolvimento foi a prévia implantação de outro sistema na empresa utilizando esta linguagem, tornando mais fácil o acesso a informações, possíveis futuras manutenções e atualizações do sistema.

Uma seção dedicada a detalhar esta linguagem e como, em particular, foi empregada no projeto é mostrada a seguir.

4.1.1: PHP (*Hypertext PreProcessor*)

PHP (*Hypertext PreProcessor*) é uma linguagem de programação utilizada para criar sites dinâmicos. Esses sites são os que permitem uma interação com o usuário mediante formulários, parâmetros de URL, links, etc., e que retornam páginas criadas em tempo real. Aplicações PHP são geradas automaticamente pelo servidor com desempenho muito satisfatório. O usuário não tem acesso ao código PHP, somente ao HTML. Esse tipo de detalhe é muito importante quando se lida, por exemplo, com senhas e acesso a banco de dados. (MORAZ, 2005)

O PHP pode ser utilizado na maioria dos sistemas operacionais e também é suportado pela maioria dos atuais servidores *web*. As funcionalidades desta linguagem não se resumem apenas a gerar HTML, incluindo arquivos PDF e geração de imagens. Além disso, o PHP oferece suporte a uma ampla variedade de Banco de Dados, tornando esta operação simples e rápida. (MORAZ, 2005)

Juntamente com as qualidades anteriormente citadas, pode-se destacar que para o sistema em questão é muito importante que seja utilizada uma linguagem simples e eficiente. Outro ponto positivo está ligado à popularidade do PHP e o fato de ser *open-source*. Uma linguagem difundida, aberta e sem custos oferece uma grande gama de fontes de pesquisa e suporte na internet, qualidade esta muito desejável uma vez que o projeto foi desenvolvido por apenas um desenvolvedor com pouca experiência.

Para edição e criação do código foi utilizado o editor *Notepad++*. Este *software open-source* é compatível com o sistema operacional Windows e

permite autocomplemento, busca e substituição de expressões regulares, divisão de tela, zoom, favoritos, etc.

Um trecho de programação feita em PHP e utilizada no sistema é mostrado abaixo no Quadro 9.

```

public function add() {
    if ($this->request->is('post')) {
        $this->Manutencao->create();
        if ($this->Manutencao->save($this->request->data)) {
            $ferramentas_id = $this->Manutencao->field('ferramentas_id');
            $this->set('ferramentas_id', $ferramentas_id); //define a
variável 'ferramentas_id'
            $this->loadModel('Ferramenta', $ferramentas_id); //carrega o
modelo 'Ferramenta'
            $this->Ferramenta->save( //altera o 'status' da ferramenta que
a folha de manutenção está sendo aberta para 'Manutencao'
                array( 'status' => 'Manutencao' ));
            $this->Session->setFlash(__('Manutenção salva com sucesso!'));
            $this->redirect(array('action' => 'index'));
        } else {
            $this->Session->setFlash(__('Erro. Tente Novamente.'));
        }
    }
    $ferramentas = $this->Manutencao->Ferramenta->find('list',
array('fields' => array('Equip_Num')));
    $this->set(compact('ferramentas'));
}

```

Quadro 9 – Método add() para abrir nova manutenção

Este trecho faz referência ao método `add()` chamado para abrir uma nova manutenção na classe `Manutencao`. Segundo (MORAZ, 2005), o que define uma programação orientada a objetos são os objetos. De um modo geral, os objetos podem ser definidos como entidades físicas pertencentes ao mundo real, como neste caso, ferramentas, manutenções e usuários.

No método supracitado, bem como nos outros tantos utilizados no código, faz-se uso do condicional *if* e também da variável `$this`. Esta variável é usada na definição da classe para se utilizar o próprio objeto.

4.2: SISTEMA DE GERENCIAMENTO DE BANCO DE DADOS (SGBD)

Tão importante quanto definir a linguagem de programação em que o código será escrito, é definir um sistema de gerenciamento de banco de dados adequado.

Um SGBD consiste no software que gerencia todo o acesso a uma ou mais bases de dados, permitindo a definição, acesso concorrente, manipulação e controle dos dados, assegurando a integridade, segurança e recuperação das bases de dados (NEVES e RUAS, 2005) (WELLING e THOMSON, 2005).

A decisão de optar pelo MySQL foi mais simples devido ao fato deste SGBD não ser totalmente desconhecido uma vez que foi utilizado na disciplina de Fundamentos de Sistemas de Banco de Dados do curso. Aliado a este conhecimento prévio, tem-se características desejáveis como o grande número de usuários em todo o mundo (suporte acessível) e também por ser usado de forma integrada à aplicação PHP em muitos casos.

4.2.1: MySQL

O MySQL é um sistema de gestão de bases de dados relacionais, suporta SQL (Linguagem de Consulta Estruturada, do inglês *Structured Query Language*), é *open source* e é um dos SGBD's para utilização profissional mais utilizados e conhecidos a nível mundial.

Após escolhido o sistema de gerenciamento de banco de dados, pôde-se prosseguir com o projeto do banco de dados. O modelo conceitual feito anteriormente e mostrado na Figura 23 é tido como o de mais alto grau de

abstração. Depois de selecionada a tecnologia do SGBD é possível dar prosseguimento ao projeto com os modelos de menor grau de abstração, o modelo lógico e o modelo físico. Os níveis de abstração de dados para cada um dos modelos citados é mostrado no Quadro 10.

MODELO	GRAU DE ABSTRAÇÃO	FOCO	INDEPENDÊNCIA
Conceitual	Médio-Alto	Visão global dos dados (independe do modelo do BD)	Hardware e Software
Lógico	Médio	Modelo específico de BD	Hardware
Físico	Baixo	Métodos de armazenamento e acesso	Nem Hardware nem software

Quadro 10 – Níveis de abstração de dados

4.2.1.1: Modelo Lógico

Como explica (MACÊDO, 2011), o modelo lógico constitui uma representação específica de um modelo interno, utilizando as estruturas de BD suportada pelo banco escolhido. Este nível é dependente do software utilizado e qualquer alteração feita no SGBD exige que o modelo interno seja modificado para adequar-se às características e exigências de implementação do modelo de BD. Nesta etapa de construção do modelo lógico serão definidos os padrões, as nomenclaturas e as chaves primárias e estrangeiras, sempre levando em conta o modelo conceitual criado anteriormente.

Pode-se ver no modelo lógico mostrado na Figura 24 que apesar da semelhança com o modelo conceitual feito anteriormente, neste modelo as relações que antes eram vistas com um alto nível de abstração passaram a ser parte das tabelas. A relação de um para muitos que relacionava a ferramenta às manutenções passou a ser definida como um campo de chave estrangeira

na tabela Manutenção. E o relacionamento de muitos para muitos que interligava as manutenções com os usuários passou a ser escrita como uma quarta tabela de relacionamento.

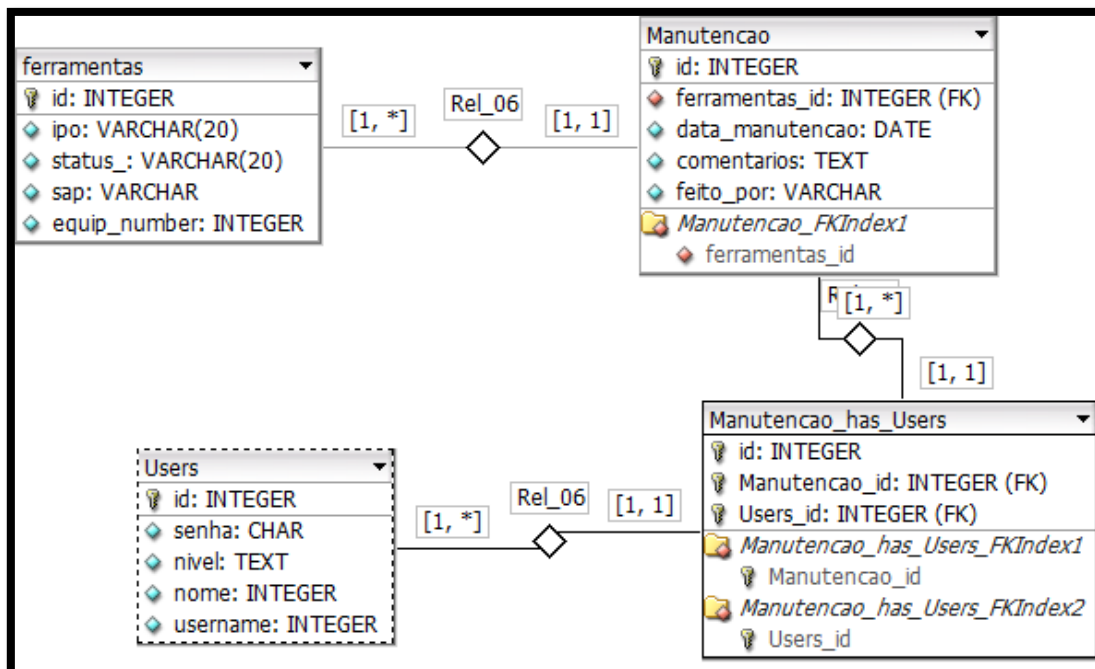


Figura 24 – Modelo lógico

4.2.1.2: Modelo Físico

O modelo de nível mais baixo de abstração descreve o modo como os dados são salvos em meios de armazenamentos. O esquema de banco de dados é enriquecido com detalhes que influenciam no desempenho do banco de dados, mas não interferem em sua funcionalidade. Neste modelo é exigida a definição tanto dos dispositivos de armazenamento físico como dos métodos de acesso necessários para se chegar aos dados nesse dispositivo, o que o torna dependente do SGBD utilizado (MACÊDO, 2011).

Para a criação do modelo físico, isto é, das próprias tabelas no banco de dados, faz-se o uso do código abaixo mostrado no Quadro 11 gerado

pelo *software* de projeto de BD. O exemplo abaixo se refere à criação da tabela 'Usuarios'.

```
CREATE TABLE Users (
  id INTEGER UNSIGNED NOT NULL AUTO_INCREMENT,
  senha CHAR NULL,
  nivel TEXT NULL,
  nome INTEGER UNSIGNED NOT NULL,
  username CHAR NULL,
  PRIMARY KEY(id)
```

Quadro 11 – Criação das tabelas da base de dados

Para auxílio no gerenciamento de informações do banco de dados foi utilizado um programa *open-source* chamado phpMyAdmin. Esta ferramenta faz a administração do banco de dados através do navegador *web* e possibilita que sejam feitas algumas operações através dele, como por exemplo, criar, modificar ou excluir bases de dados, campos ou linhas das tabelas do BD, executar operações SQL e gerenciar usuários e permissões de uso.

A interface de gerenciamento do phpMyAdmin é mostrada abaixo na Figura 25 com detalhe para a tabela de ferramentas.

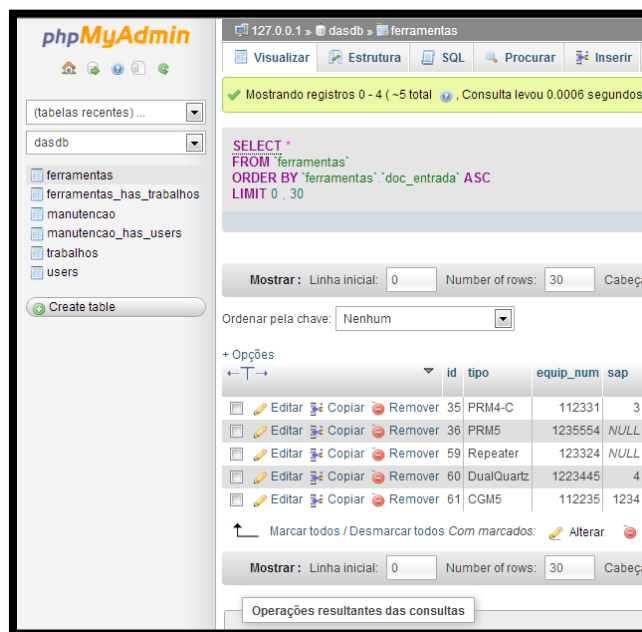


Figura 25 – Interface phpMyAdmin

4.3: SERVIDOR

Um servidor é um sistema de computação centralizada que fornece serviços a uma rede de computadores. Desta maneira, para que o sistema possa ser acessado de todos os computadores que estiverem na rede interna da empresa é importante que um servidor web seja escolhido para hospedar o sistema e permitir que os clientes (navegadores dos usuários) façam solicitações.

A empresa disponibilizou um computador servidor para que o sistema fosse armazenado e disponibilizado. Este servidor já funciona de forma a disponibilizar aplicações para uso interno da empresa.

4.3.1: Apache

O servidor *web* selecionado para ser instalado em um computador servidor para prover a aplicação para os usuários foi o Apache devido ao fato de ser grátis e *open-source*, bem como ser o servidor *web* mais bem sucedido e ser vastamente utilizado em todo o mundo (aproximadamente de 50% dos servidores são Apache).

Para facilitar a compatibilidade de versões dos *softwares*, foi escolhido um programa que funciona como um pacote integrado do Apache, MySQL, PHP e Perl independente de plataforma, o Xampp.

4.3.2: Xampp

O XAMPP é um pacote com os principais servidores de código aberto do mercado, incluindo FTP, banco de dados MySQL e Apache com os interpretadores para linguagens de script: PHP e Perl. Também é incluído

neste pacote alguns módulos extras, dentre ele o phpMyAdmin citado anteriormente neste documento (HIGA, 2012).

A grande vantagem do uso deste pacote é ter vários dos aplicativos necessários para o desenvolvimento do projeto de forma integrada, pré-configurada e compatível, apenas sendo necessário descompactar o arquivo e instalá-lo para começar a utilização.

Este servidor Xampp foi instalado no computador servidor da empresa de forma a inicializar automaticamente sempre que ligado. Neste servidor também estão os arquivos do banco de dados e os códigos do sistema. Assim, sempre que ligado este servidor provê acesso livre ao sistema e as informações previamente cadastradas a todos os usuários dentro da rede Halliburton.

4.4: FRAMEWORK

A necessidade de construção rápida sem perder funcionalidades e mantendo o sistema estável fez com que a utilização de um *framework* fosse considerada importante.

Segundo (FAYAD e SCHIMIDT, 1997), um framework é um conjunto de classes que colaboram para realizar uma responsabilidade para um domínio de um subsistema da aplicação. Assim, a necessidade de criar um sistema com alta produtividade e rapidez sem abrir mão da estabilidade, flexibilidade e qualidade faz com que o processo de desenvolvimento seja apoiado pelo reuso de estruturas pré-existentes.

4.4.1: CakePHP

O framework escolhido para uso no projeto foi o cakePHP. Sua escolha se deu por ser escrito em PHP e devido ao fácil acesso a informações. Por ser *open-source* e gratuito, existem diversas comunidades para discussão e suporte on-line.

A idéia padrão de trabalho por trás do CakePHP está ligada ao modelo MVC (Model View Controller). MVC é uma forma de programação que isola a lógica de negócio (como a aplicação funciona) da camada de exibição (a parte visual). O Model cuida do banco de dados, o View cuida da camada de exibição e o Controller cuida da parte lógica. Isto torna o trabalho mais rápido e de forma mais organizada e modularizada.

A estrutura do CakePHP destacada na Figura 26 possui a característica de usar as classes de Controlador, Modelo e Visão. Além destas, o *framework* também possui classes e objetos adicionais que tornam o desenvolvimento em MVC mais rápido. Componentes, Behaviors e Ajudantes são classes que proporcionam extensibilidade e reuso para adicionar funcionalidades rapidamente à base MVC das aplicações.

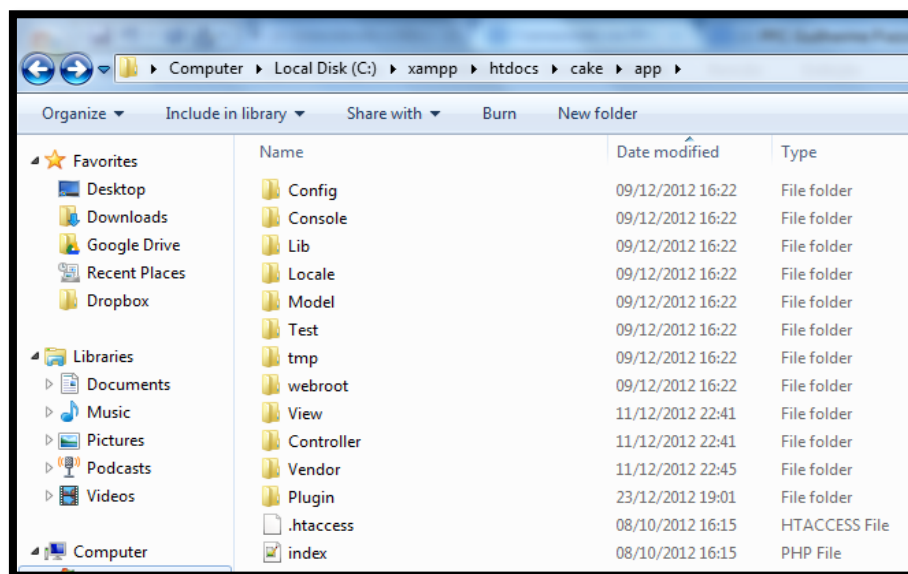


Figura 26 – Estrutura do CakePHP

4.5: MVC (*MODEL-VIEW-CONTROLLER*)

O modelo MVC, cada vez mais utilizado na engenharia de *software*, isola a parte lógica da aplicação da interface do usuário, permitindo desenvolver, editar e testar separadamente cada parte, oferecendo maior organização e modularidade ao trabalho. Estas características caíram muito bem no que se esperava do desenvolvimento do projeto uma vez que funcionalidades poderiam ser adicionadas ou editadas a qualquer momento e de forma isolada. (CAKE SOFTWARE FOUNDATION, 2012)

A aplicação feita em MVC separa sua aplicação em três partes principais:

Modelo: representa os dados;

Visão: representa a visualização dos dados;

Controlador: manipula e roteia as requisições dos usuários.

A Figura 27 mostra um exemplo de uma requisição MVC em CakePHP que será feita no sistema. Esta requisição ilustra o fluxo de informações da operação de abrir novo formulário de manutenção que é descrita abaixo:

1. A partir do navegador, o usuário clica no link que aponta para a função adicionar manutenção e o navegador faz uma requisição ao sistema;
2. O expedidor verifica o endereço requisitado (adicionar manutenção) e redireciona para o controlador correto;
3. O controlador executa a lógica específica da aplicação. Por exemplo, verifica se o usuário está autenticado e seu nível de acesso é coerente com a requisição e também usa os modelos para acessar os dados da aplicação. Neste exemplo, o controlador usa o modelo para trazer ao usuário os dados da folha de manutenção requisitada;

4. Depois que o controlador operou sobre os dados, ele os repassa para a visão que por sua vez faz com que os dados fiquem prontos para a apresentação ao usuário.
5. Uma vez que a visão tenha usado os dados provenientes do controlador para construir a página, o conteúdo é retornado ao navegador do usuário.

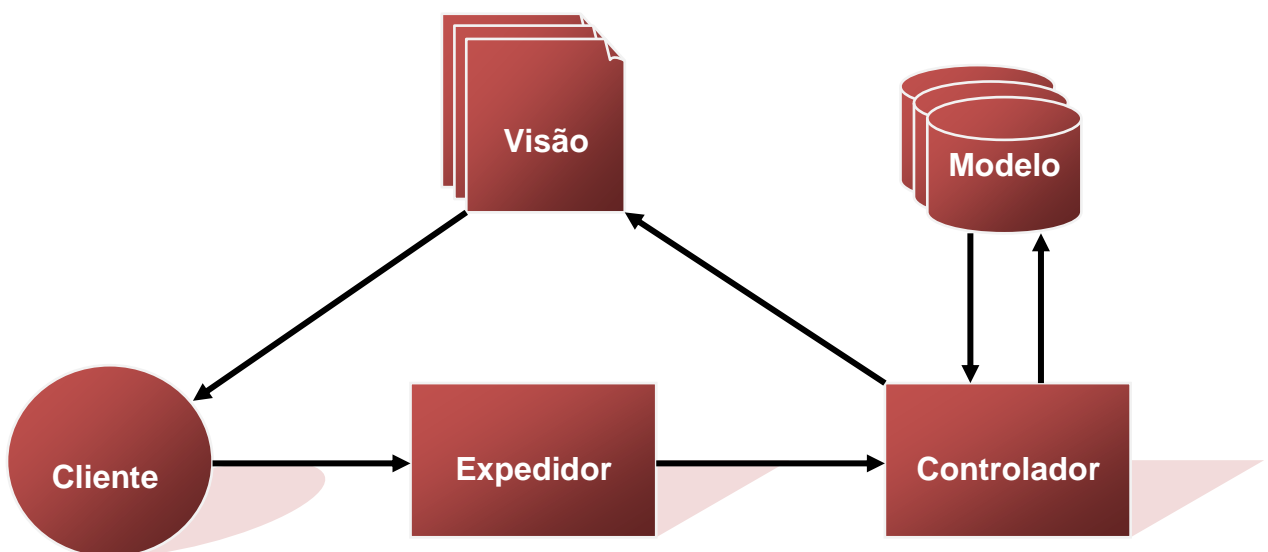


Figura 27 – Requisição MVC

Apesar da sequência acima detalhada ser específica para uma operação do sistema, praticamente toda requisição da aplicação seguirá o modelo básico do modelo.

4.5.1: Modelos

Os modelos do sistema representam os dados do banco de dados e são responsáveis por fornecer informações aos controladores.

O cakePHP permite que no código dos modelos se crie regras de inclusão de dados bem como se defina as relações entre modelos (análogo às relações entre entidades do modelo conceitual do BD).

Conforme do padrão de nomenclaturas do cakePHP os relacionamentos podem ser do tipo *hasOne*, *hasMany*, *belongsTo* e *hasAndBelongsToMany*. Cada um destes tipos é exemplificado abaixo no Quadro 12.

Relacionamento	Tipo de associação	Exemplo
Um para um	hasOne	Uma ferramenta possui uma locação (caso fictício não presente no projeto).
Um para muitos	hasMany	Uma ferramenta tem várias manutenções.
Muitos para um	belongsTo	Muitas manutenções pertencem a uma ferramenta.
Muitos para muitos	hasAndBelongsToMany	Uma manutenção é feita por vários usuários e um usuário participa de muitas manutenções.

Quadro 12 – Associações entre modelos

As regras para os campos do modelo são definidas por códigos de programação no modelo. São elas:

Nº	Regra	Descrição
1	notEmpty	Preenchimento obrigatório.
2	inList	Valor de preenchimento deve constar na lista predefinida.
3	isUnique	Valor do campo deve ser único na tabela.
4	numeric	Valor deve apresentar apenas números.
5	between(x,y)	Valor numérico deve estar entre x e y.

Quadro 13 – Regras utilizadas para inclusão de dados nos campos dos modelos

O sistema de gerenciamento de informações de ferramentas foi projetado com os modelos apresentados no Quadro 14, Quadro 15, Quadro 16 e Quadro 17.

Modelo: Ferramenta	
<u>Descrição:</u> Modelo dedicado às ferramentas cadastradas no sistema. Este modelo tem um relacionamento de <i>um para muitos</i> com o modelo Manutencao.	
<u>Campos com regras:</u> equipment_number: 1,3,4,5(6,6) tipo: 1 sap: 1,4 status: 1	<u>Relacionamento:</u> <pre>public \$hasMany = array('manutencao' => array('className' => 'manutencao', 'foreignKey' => 'ferramentas_id' ...);</pre>

Quadro 14 – Modelo 'Ferramenta'

Modelo: Manutencao	
<u>Descrição:</u> Modelo dedicado às manutenções cadastradas no sistema. Este modelo tem um relacionamento de <i>muitos para um</i> com o modelo Ferramenta e <i>muitos para muitos</i> com o modelo User.	
<u>Campos com regras:</u> data: 1 ferramentas_id: 1, 2	<u>Relacionamento:</u> <pre>public \$belongsTo = array('Ferramenta' => array('className' => 'Ferramenta', 'foreignKey' => 'ferramentas_id' ...); public \$hasAndBelongsToMany = array('user' => array('className' => 'user', 'joinTable' => 'manutencao_has_users', 'foreignKey' => 'Manutencao_id', 'associationForeignKey' => 'manutencao_id' ...);</pre>

Quadro 15 – Modelo 'Manutencao'

Modelo: User	
Descrição: Modelo dedicado às manutenções cadastradas no sistema. Este modelo tem um relacionamento de <i>muitos para muitos</i> com o modelo Manutencao.	
Campos com regras: <i>username: 1, 3, 4, 5(6,6)</i> <i>senha: 1</i> <i>nível: 1,2</i> <i>nome: 1</i>	Relacionamento: <pre>public \$hasAndBelongsToMany = array ('Manutencao' => array ('className' => 'manutencao', 'joinTable' => 'manutencao_has_users', 'foreignKey' => 'users_id', 'associationForeignKey' => 'manutencao_id' ...);</pre>

Quadro 16 – Modelo ‘User’

Modelo: ManutencaoHasUser	
Descrição: Modelo dedicado a fazer o relacionamento modelado como “manutenção é feita por muitos usuários”.	
Campos: <i>manutenção_id: 4</i> <i>user_id: 4</i>	Relacionamento: <pre>public \$belongsTo = array ('Manutencao' => array ('className' => 'Manutencao', 'foreignKey' => 'manutencao_id' ...), 'User' => array ('className' => 'User', 'foreignKey' => 'users_id', ...);</pre>

Quadro 17 – Modelo ‘ManutencaoHasUser’

4.5.2: Controladores

Os controladores do aplicativo são uma parte importante no sistema como um todo. Passa por estes componentes a lógica do sistema e o processamento das requisições do usuário, sendo mecanismos intermediários entre as visões e os modelos. Sua lógica é feita a partir de funções definidas internamente, os métodos. Estes métodos são chamados a partir de requisições de usuários, fazem processamentos específicos do sistema e mostram e alimentam as visões com dados provenientes das consultas aos modelos.

Os controladores de aplicação são classes que estendem a classe CakePHP ApplicationController, a qual por sua vez estende a classe núcleo Controller. A classe ApplicationController deve conter métodos que são compartilhados entre todos os controladores. (CAKE SOFTWARE FOUNDATION, 2012)

Foram criados para o sistema 4 controladores que, por sua vez, gerenciam a lógica de cada um dos 4 modelos já detalhadas anteriormente. Cada um destes controladores é descrito juntamente com seus métodos no Quadro 19, Quadro 20, Quadro 18, Quadro 21.

Controlador: UsersController

Descrição: Este controlador é responsável pelas operações e consultas dos usuários do sistema.

Métodos:

beforeFilter(): controla as permissões de acesso aos métodos pelos usuários;

index(): faz a listagem de todos os usuários cadastrados;

login(): método para autenticação de usuários cadastrados;

logout(): métodos para que usuários autenticados façam logout;

view(\$id): visualiza as informações do usuário selecionado;

add(): adiciona um novo usuário;

edit(\$id): edita um usuário selecionado;

delete(\$id): exclui um usuário selecionado;

Quadro 18 – Controlador do modelo 'User'

Controlador: FerramentasController

Descrição: Este controlador é responsável pelas operações e consultas das ferramentas do sistema.

Métodos:

***beforeFilter():** controla as permissões de acesso aos métodos pelos usuários;*

***index():** gera a página inicial do sistema listando todas as ferramentas;*

***view(\$id):** visualiza as informações da ferramenta selecionada;*

***add():** adiciona uma nova ferramenta;*

***edit(\$id):** edita uma ferramenta selecionada;*

***delete(\$id):** exclui uma ferramenta selecionada;*

***place_order(\$grupo):** abre arquivo predeterminado localizado no servidor;*

***report(\$option):** gera relatório de manutenção;*

***report_pdf(\$option):** gera arquivo PDF do relatório de manutenção;*

***status(\$id):** altera o status de uma ferramenta selecionada;*

***reportStatus ():** gera relatório de status das ferramentas;*

***reportStatusPDF ():** gera arquivo PDF do relatório de status das ferramentas;*

Quadro 19 – Controlador do modelo 'Ferramenta'

Controlador: ManutencaosController

Descrição: Este controlador é responsável pelas operações e consultas das manutenções do sistema.

Métodos:

***beforeFilter():** controla as permissões de acesso aos métodos pelos usuários;*

***index():** faz a listagem todas as manutenções;*

***view(\$id):** visualiza as informações da manutenção selecionada;*

***add(\$id):** adiciona uma nova manutenção;*

***edit(\$id):** edita uma manutenção selecionada;*

***delete(\$id):** exclui uma manutenção selecionada;*

***view_pdf(\$option):** gera formulário de manutenção em formato PDF;*

Quadro 20 – Controlador do modelo 'Manutencao'

Controlador: ManutencaoHasUsersController

Descrição: Controlador do modelo de relacionamento das manutenções com os usuários.

Métodos:

add(): *adiciona relação de manutenção com usuário que a criou;*

Quadro 21 – Controlador do modelo ‘ManutencaoHasUser’

4.5.3: Visões

As visões do sistema no modelo MVC compreendem as interfaces do usuário. Cada tela projetada e pensada anteriormente (no modelo Wireframe, por exemplo) será uma visão que será mostrada pelo controlador. As visões do sistema devem ser criadas em pastas específicas para cada controlador.

A maioria das ações (métodos) possui uma visão relacionada destinada à visualização dos resultados das operações e consultas do sistema. As visões de cada controlador do sistema são listadas abaixo no Quadro 22.

Controlador	Visões
Ferramentas	<i>index, add, edit, view, place_order, report, report_status, status, report_pdf, report_status_pdf.</i>
Manutencaos	<i>index, add, edit, view, view_pdf.</i>
Users	<i>index,add, edit, view, login,</i>

Quadro 22 – Visões do sistema

As visões do sistema serão mostradas na seção a frente destinada ao detalhamento do sistema onde as sequencias e fluxos de informação serão testados para confirmação de atendimento dos requisitos.

4.5.3.1: Elements

Elements são trechos de código que poderão ser usados por uma ou mais visões em busca de maior eficiência na programação, reduzindo o tempo gasto com retrabalho e priorizando boas práticas como o reaproveitamento de código.

Esta boa prática se mostrou muito útil uma vez que existem elementos visuais comuns a várias visões, como por exemplo, os menus de navegação que são mostrados em quase todas as telas.

Para que este trecho de código fosse parametrizado facilitando a possível inclusão de novas funcionalidades no menu, estes foram programados como *Elements* e apenas sendo chamados nas visões de acordo com o nível do usuário autenticado.

O trecho de código mostrado abaixo no Quadro 23 demonstra como os elementos são selecionados de acordo com o nível do usuário autenticado. Caso seja necessário que o menu do administrador apresente novos *links*, basta fazer a modificação uma única vez no código do elemento *menu_admin.ctp* que todas as visões terão seu menu atualizado.

```
if($logged_user['role']=='user') echo $this->element('menu_user');  
if($isAdmin) echo $this->element('menu_admin');  
if($logged_user=='') echo $this->element('menu_unlogged');
```

Quadro 23 – Elementos de visões

Os três menus do exemplo anterior são mostrados abaixo na Figura 28.

The image displays three distinct user navigation menus for different user levels, all for the user **Guilherme Piazzetta**.

- Left Panel (User Level):** Greeting: "Olá, **Guilherme Piazzetta** (user).". Menu items: "Ferramentas", "Manutenções". Login fields: "SAP:*" and "Senha*" with corresponding input boxes. A green "Login" button is present.
- Middle Panel (User Level):** Greeting: "Olá, **Guilherme Piazzetta** (user).". Menu items: "Ferramentas", "Manutenções", "Nova Manutenção", "Logout".
- Right Panel (Admin Level):** Greeting: "Olá, **Guilherme Piazzetta** (admin).". Menu items: "Ferramentas", "Manutenções", "Nova Manutenção", "Nova Ferramenta", "Usuários", "Novo Usuário", "Logout". Reports section: "Relatórios" with items "Relatório - Gauges", "Relatório - Repeaters", and "Relatório - Status".

Figura 28 – Menus de navegação para os 3 níveis de usuários

CAPÍTULO 5: DETALHAMENTO

Este quinto capítulo documenta a conclusão do período de construção do projeto e o período de transição para implantação do sistema na empresa.

O capítulo 3 foi destinado à formalização do problema com o levantamento dos requisitos e aspectos conceituais que, com o auxílio de técnicas e metodologias de desenvolvimento, foram mais bem compreendidos.

No capítulo 4, o sistema foi implementado através da utilização de tecnologias e ferramentas consideradas mais bem adequadas para sua resolução.

Este quinto capítulo fecha o ciclo do desenvolvimento com a validação da solução escolhida e desenvolvida anteriormente através do detalhamento das interfaces (visões) do sistema a fim de verificar a eficiência da solução encontrada para atender o problema tratando cada caso de uso e requisito levantado inicialmente.

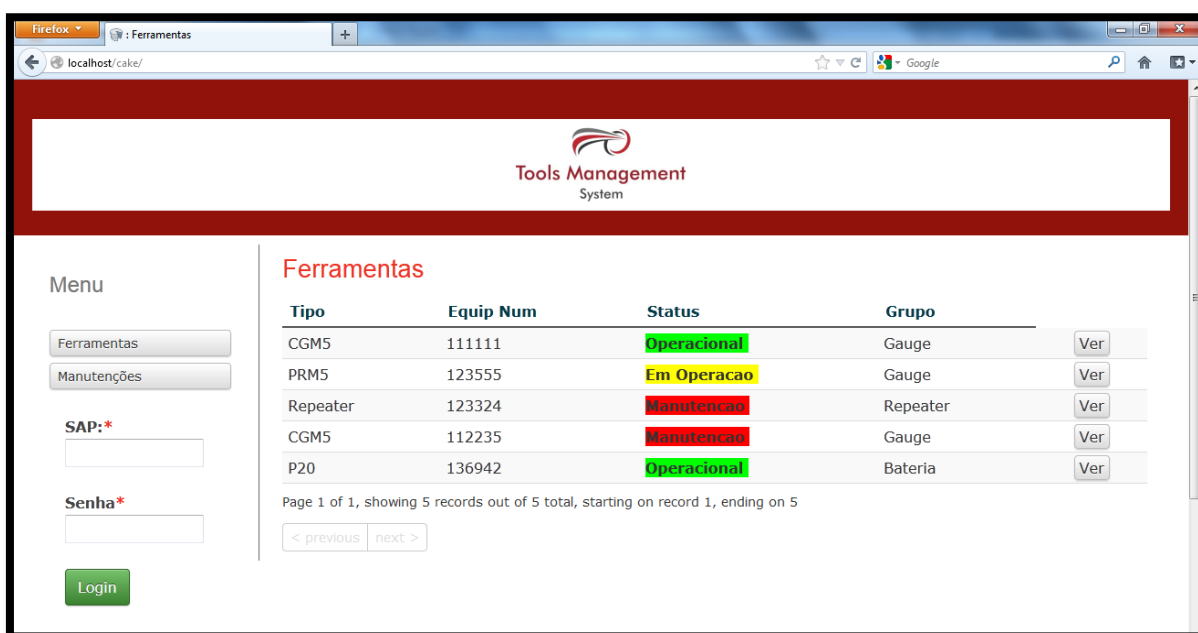
5.1: PERMISSÕES

Desde o início foi requisitado que diferentes usuários pudessem acessar diferentes funções com permissões também diferenciadas. Para suprir esta exigência foi definido que ao se cadastrar um usuário, este deve ter seu campo de 'nível' selecionado como usuário (comum) ou administrador. A seguir será feita uma análise das permissões dos 3 possíveis usuários que farão uso do sistema.

5.1.1: Usuário não autenticado

O primeiro tipo de usuário a ser considerado é o não autenticado. Este usuário pode ter esta condição por não estar cadastrado no sistema ou então por desejar apenas fazer consultas rápidas ao sistema que não necessitem sua autenticação. Vale ainda ressaltar que qualquer usuário apenas tem acesso ao sistema se conectado à rede interna da empresa.

Ao acessar o sistema o usuário irá visualizar a tela inicial com uma lista das ferramentas cadastradas com destaque para o *status* atual. O menu localizado na barra lateral esquerda é específico para este tipo de usuário e conta com um *link* para as manutenções e campos para autenticação. Esta interface é mostrada na Figura 29.



The screenshot shows the 'Tools Management System' interface. The header is red and contains the system logo. The left sidebar has a 'Menu' section with buttons for 'Ferramentas' and 'Manutenções', and login fields for 'SAP:*' and 'Senha*' with a 'Login' button. The main content area is titled 'Ferramentas' and displays a table with the following data:

Tipo	Equip Num	Status	Grupo	Ver
CGM5	111111	Operacional	Gauge	Ver
PRM5	123555	Em Operacao	Gauge	Ver
Repeater	123324	Manutencao	Repeater	Ver
CGM5	112235	Manutencao	Gauge	Ver
P20	136942	Operacional	Bateria	Ver

Page 1 of 1, showing 5 records out of 5 total, starting on record 1, ending on 5

Figura 29 – Página inicial para usuário não autenticado

Ainda na tela inicial, o usuário não autenticado poderá visualizar mais informações das ferramentas ao clicar no *link* “ver”. Nesta tela, existe um *link* disponível para que o manual (ou folha de engenharia) da ferramenta seja aberto. A Figura 30 mostra como acessar este manual.

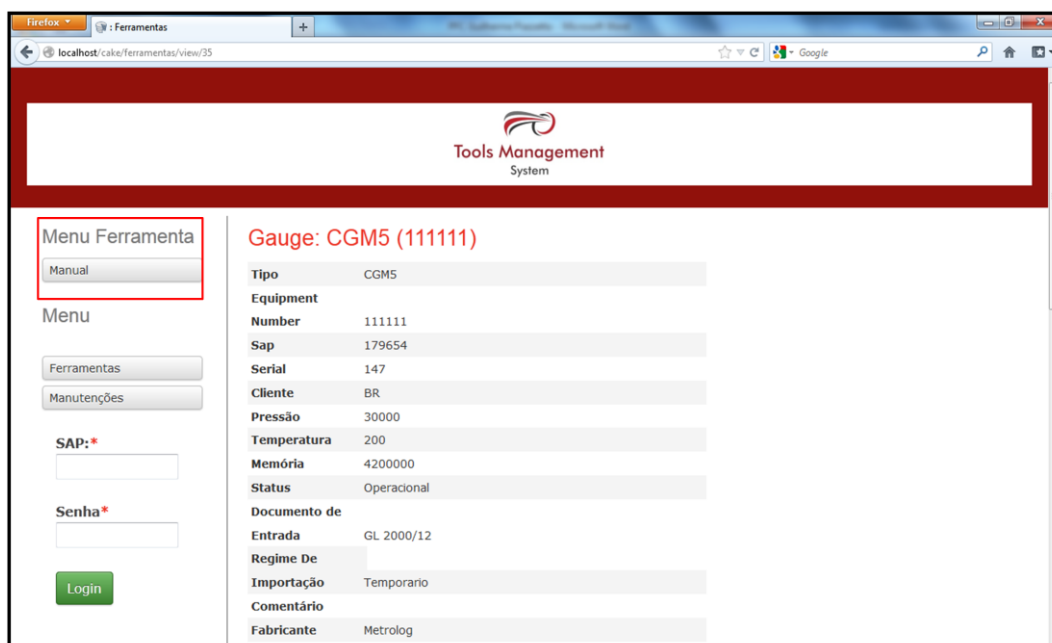


Figura 30 – Informações de ferramenta com destaque para *link* para manual da ferramenta.

O usuário, ainda na condição de não autenticado, pode visualizar as manutenções cadastradas clicando no *link* manutenção do menu principal (Figura 31), que o redirecionará para uma lista de manutenções que pode ser ordenada pela data da manutenção, nome do usuário que abriu a manutenção ou *equipment number* da ferramenta.

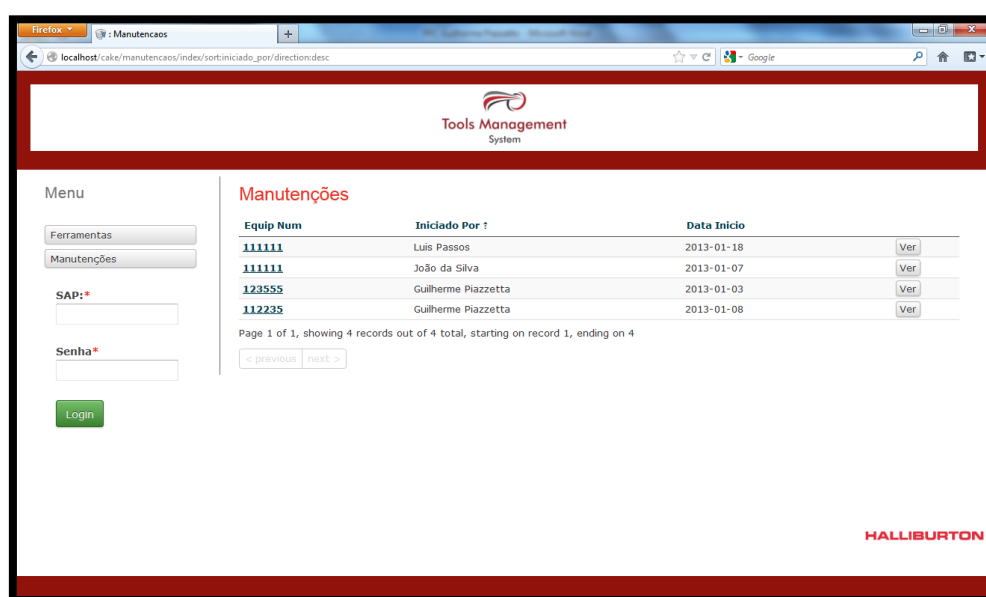


Figura 31 – Lista de manutenções ordenadas pelo *Equipment Number*

Ao selecionar uma manutenção da lista para ver com mais detalhes, o usuário não autenticado terá todas as informações do formulário de manutenção disponíveis e ainda terá um *link* para gerar um arquivo PDF deste formulário. A manutenção e o *link* para geração de formulário em PDF é mostrado abaixo na Figura 32 e o formulário PDF gerado na Figura 33.

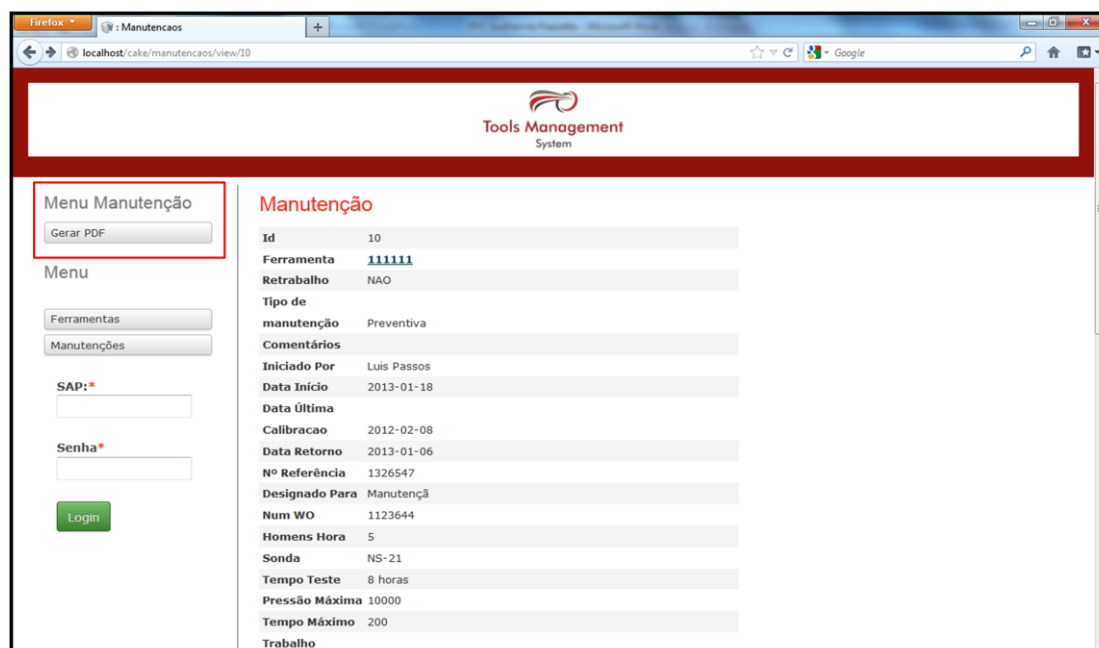


Figura 32 – Visualização de manutenção.



Figura 33 – Formulário de manutenção em PDF.

5.1.2: Usuário comum

Todo usuário imediatamente ao fazer o acesso ao sistema é considerado um usuário não autenticado. Após preencher os campos do menu com o *username* e a senha e submeter o *login*, este usuário estará autenticado no sistema.

Usuários cadastrados no nível mais baixo são considerados usuários comuns. Estes usuários possuem, além das permissões do usuário não autenticado, alguns outros privilégios.

Na tela inicial é possível ver que novos botões são mostrados abrindo novas possibilidades de uso ao usuário bem como sua identificação. Na lista de ferramentas, ao lado do *link* para visualização de mais informações, o *link* “Status” direciona o usuário para uma página de mudança rápida de *status* da ferramenta. Estas mudanças na interface podem ser vistas abaixo na Figura 34.

The screenshot displays the 'Tools Management System' interface. At the top, the system logo is centered. Below it, the user is greeted as 'Guilherme Piazzetta (user)'. A menu on the left contains buttons for 'Ferramentas', 'Manutenções', 'Nova Manutenção', and 'Logout'. The main content area is titled 'Ferramentas' and contains a table with the following data:

Tipo	Equip Num	Status	Grupo	Ver	Status
P20	136942	Operacional	Bateria	Ver	Status
CGM5	111111	Manutenção	Gauge	Ver	Status
PRM5	123555	Em Operacao	Gauge	Ver	Status
CGM5	112235	Manutenção	Gauge	Ver	Status
Repeater	123324	Manutenção	Repeater	Ver	Status

Below the table, it indicates 'Page 1 of 1, showing 5 records out of 5 total, starting on record 1, ending on 5'. Navigation buttons for '< previous' and 'next >' are present. The HALLIBURTON logo is located in the bottom right corner.

Figura 34 – Página inicial do usuário comum.

No menu do usuário, o *link* “Nova Manutenção” permite que este abra uma nova folha de manutenção. Esta operação será mais bem detalhada na subseção dedicada aos casos de uso.

O usuário que visualiza a lista de manutenções pode notar a presença de botões de gerenciamento de manutenção (em destaque na Figura 35). Além de criar uma nova manutenção, este usuário tem permissão para edição e exclusão de manutenções anteriormente abertas. Por fim o usuário pode fazer o *logout* através do menu principal.

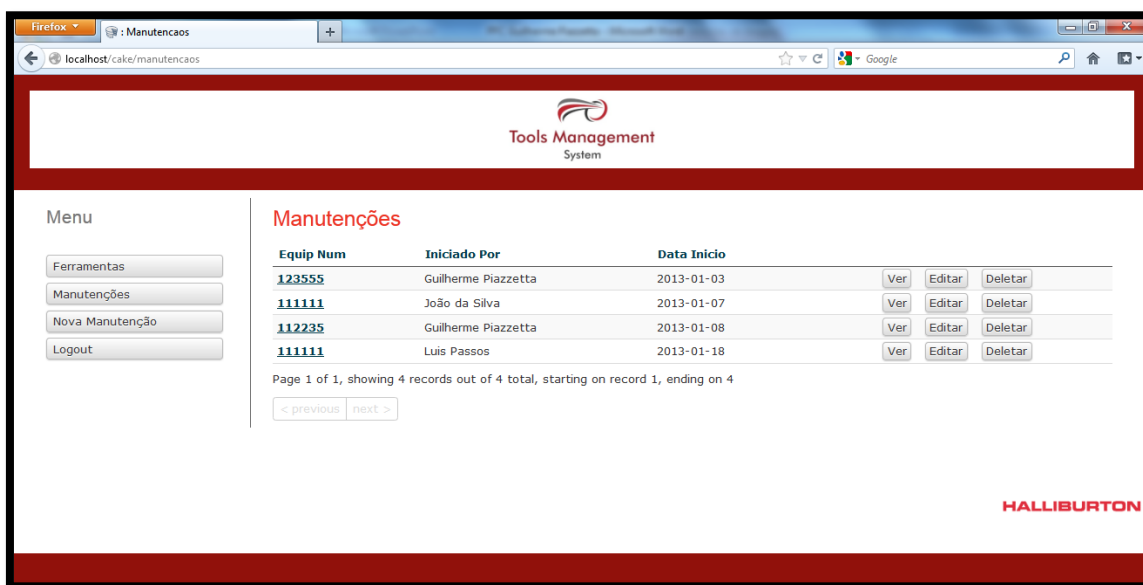


Figura 35 – Menu de gerenciamento de manutenção.

5.1.3: Administrador

Os administradores do sistema possuem permissão total para todas as funcionalidades. Logo após a autenticação, administrador pode ver na barra lateral a identificação pessoal, o menu de comandos com mais funções que o do usuário comum e também um menu dedicado à geração de relatórios.

O menu principal apresenta, além das funções já detalhadas anteriormente para o usuário normal, permissões de gerenciamento de ferramentas e usuários. Cada ferramenta da lista possui, além dos *links* para visualização e mudança de *status*, botões para edição e exclusão como observado na Figura 36.

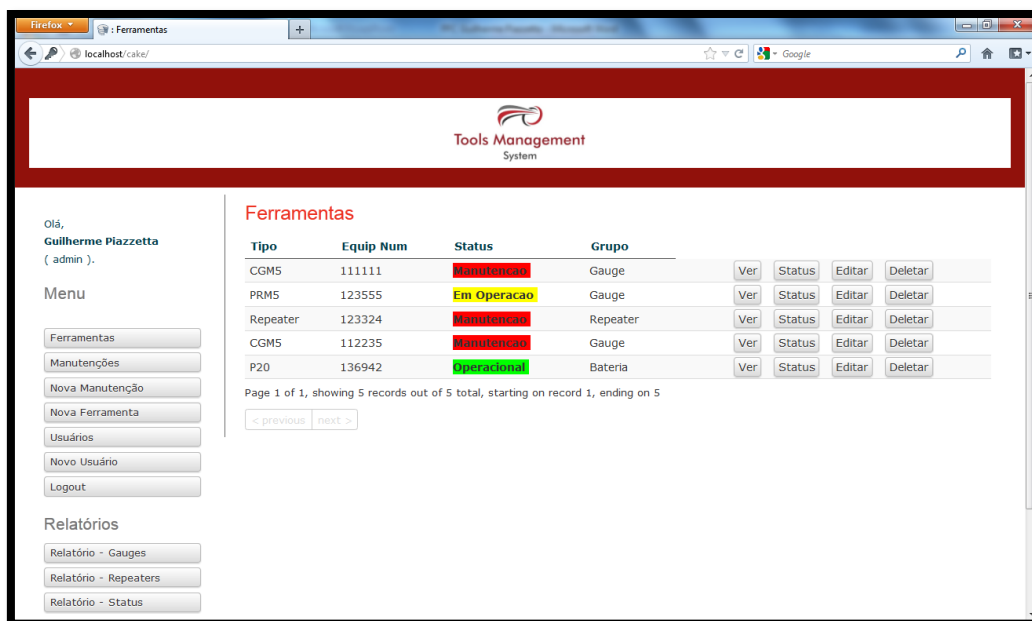


Figura 36 – Página inicial do administrador.

O administrador tem permissão também para visualizar os usuários cadastrados através do *link* do menu principal e para fazer o gerenciamento destes através dos botões de “Novo Usuário”, “Editar” e “Deletar”. A página dedicada à visualização e gerenciamento dos usuários cadastrados é mostrada a seguir na Figura 37.

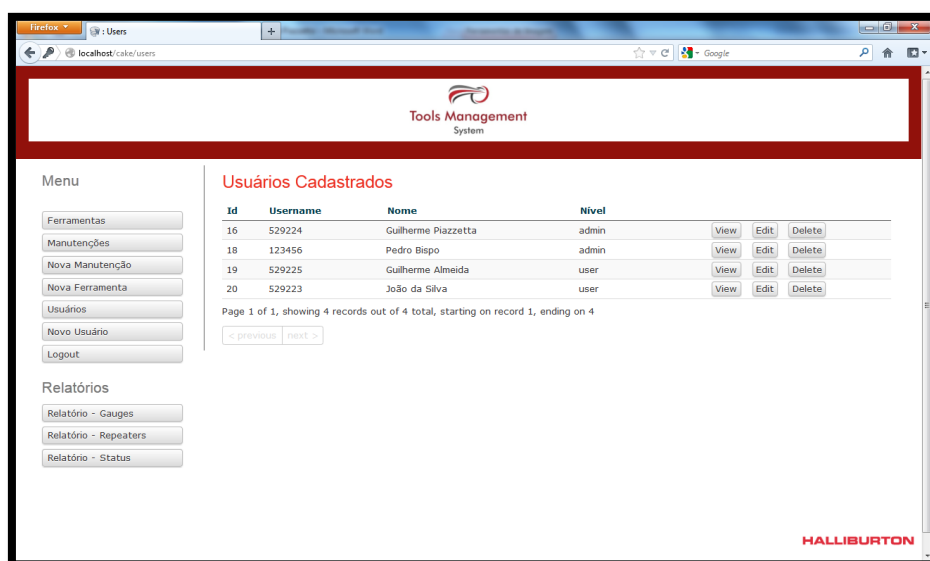


Figura 37 – Página de gerenciamento de usuários.

Abaixo do menu principal está o menu de relatórios do administrador. Através deste menu é possível que sejam gerados relatórios de manutenção de equipamentos e também o relatório de *status*. Todos os relatórios são mostrados em uma página *web* com possibilidade de geração do arquivo PDF por meio do *link* disponível no início na página. Todos os relatórios são gerados apresentando em seu corpo a data atual. As interfaces dos relatórios gerados serão mostradas na seção dedicada ao detalhamento deste caso de uso.

5.2: CASOS DE USO

Para validação dos requisitos levantados, esta seção irá detalhar os principais casos de uso do sistema e suas exceções através das visões do usuário.

5.2.1: Mostrar *Status* de ferramentas (F1)

Visando uma maior agilidade na busca por informações e uma navegação mais intuitiva, buscou-se atender este requisito da maneira mais simples possível para o usuário.

A consulta pela informação do *status* da ferramenta é feita por qualquer usuário acessando a página inicial do sistema. Nela são listadas as ferramentas cadastradas e, em destaque, é mostrado o *status* de cada uma com sua cor padrão (vermelho para manutenção, amarelo para em operação e verde para operacional) para uma melhor e mais rápida visualização. A página inicial com esta informação do *status* das ferramentas já foi mostrada anteriormente na Figura 29, Figura 34 e Figura 36.

5.2.2: Mudar *Status* de ferramentas (F2)

A mudança de *status* das ferramentas deve ser feita também de maneira rápida e intuitiva com o objetivo de motivar os funcionários a alimentar o sistema com esta importante informação para a linha de supervisão.

Ao fazer a autenticação, o usuário encontra na página inicial onde as ferramentas são listadas um *link* direto para a troca de *status* da ferramenta. Ao clicar neste botão, uma nova página será aberta mostrando o *status* atual e disponibilizando uma lista com os possíveis *status* para a ferramenta. Basta selecionar o novo *status* e confirmar no botão verde para que a mensagem de sucesso de operação seja mostrada. Esta operação é realizada pelo método *status()* do controlador Ferramenta e a interface desta operação é mostrada abaixo na Figura 38.

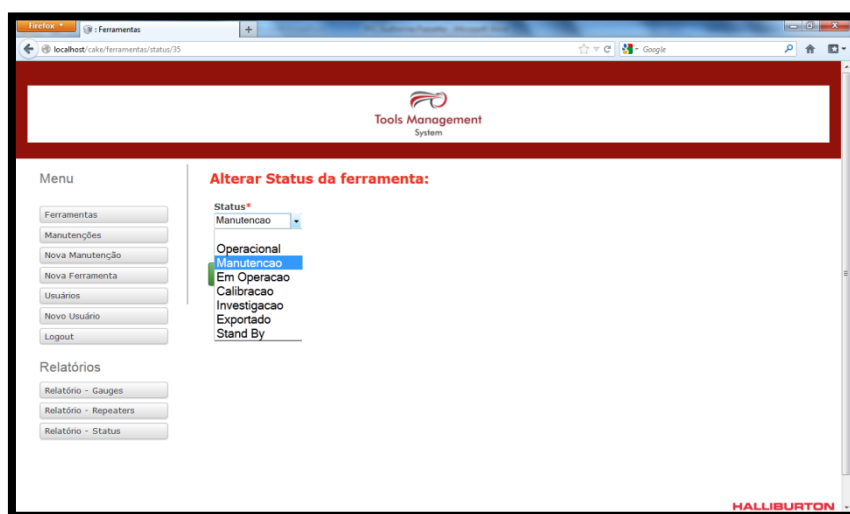
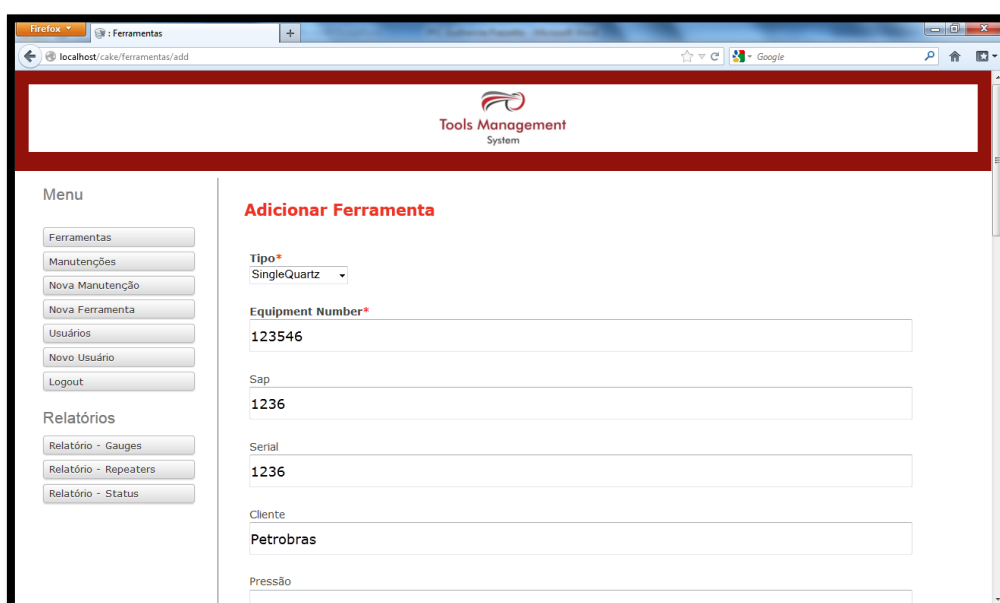


Figura 38 – Alteração do *status* de ferramenta.

A troca de *status* automática é feita quando uma folha de manutenção é aberta para determinada ferramenta, tornando seu *status* como “Manutenção”.

5.2.3: Gerenciamento do sistema (F3)

O gerenciamento de ferramentas se dá através das funções de adição, edição e exclusão. Para adicionar uma nova ferramenta basta clicar no botão “Nova Ferramenta” do menu principal. O usuário então é redirecionado para um formulário com campos referentes à informações da ferramenta a ser adicionada. Ao final do preenchimento dos campos, o botão verde deve ser pressionado para que a ferramenta seja salva no banco de dados através do método *add()*. A Figura 39 ilustra a página de adição de uma nova ferramenta.



The screenshot shows a web browser window with the URL `localhost/cake/ferramentas/add`. The page title is "Ferramentas". The main content area is titled "Adicionar Ferramenta" and contains a form with the following fields:

- Tipo***: A dropdown menu with "SingleQuartz" selected.
- Equipment Number***: A text input field containing "123546".
- Sap**: A text input field containing "1236".
- Serial**: A text input field containing "1236".
- Cliente**: A text input field containing "Petrobras".
- Pressão**: A text input field.

On the left side, there is a "Menu" section with buttons for "Ferramentas", "Manutenções", "Nova Manutenção", "Nova Ferramenta", "Usuários", "Novo Usuário", and "Logout". Below the menu is a "Relatórios" section with buttons for "Relatório - Gauges", "Relatório - Repeaters", and "Relatório - Status".

Figura 39 – Parte superior da página para adição de nova ferramenta no banco de dados

As operações de edição e exclusão de ferramentas podem ser feitas de duas maneiras: buscando a ferramenta desejada na lista da página inicial; e clicando nos botões ao lado da descrição da ferramenta ou então selecionando a ferramenta para visualização (como mostrado em detalhe na Figura 40). Após abrir a página de informações da ferramenta selecionada, botões para “Editar Ferramenta” e “Deletar Ferramenta” estão disponíveis na barra lateral esquerda.

Ao se selecionar a função de editar ferramenta, o método *edit()* será chamado para redirecionamento à uma página com campos editáveis referentes aos atributos da ferramenta similar à página de adição. Após fazer as modificações necessárias, o usuário deve confirmar e finalizar a edição pressionando o botão verde ao fim da página.

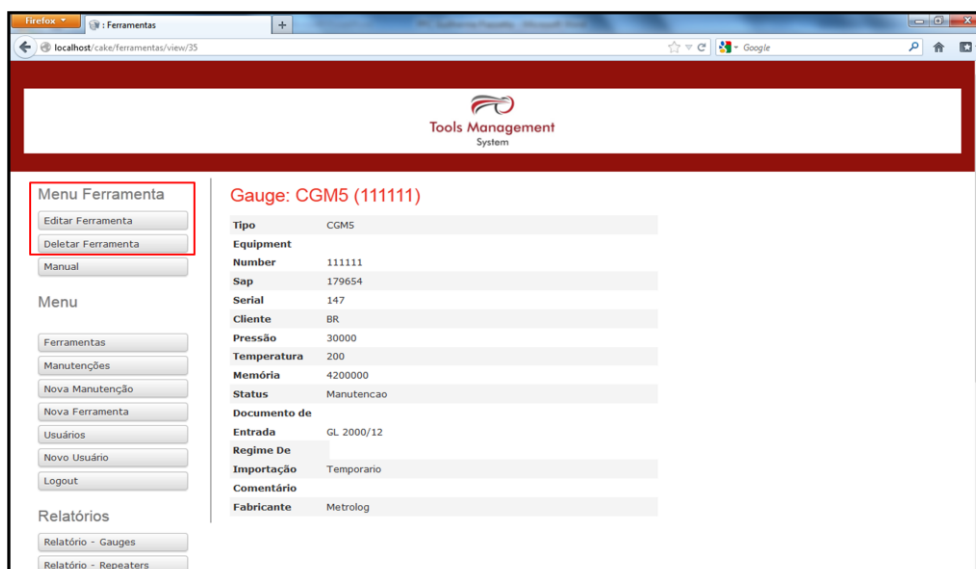


Figura 40 – Visualização de ferramenta por administrador com detalhe para menu de gerenciamento.

Para excluir uma ferramenta do sistema, o administrador, após selecionar esta opção no *link* "Deletar", receberá uma mensagem de confirmação para evitar que ferramentas sejam excluídas por engano. Após fazer a confirmação, uma mensagem de sucesso é exibida e a ferramenta deixa de figurar na lista da página inicial. Estas mensagens ao usuário são mostradas na Figura 41 e Figura 42.

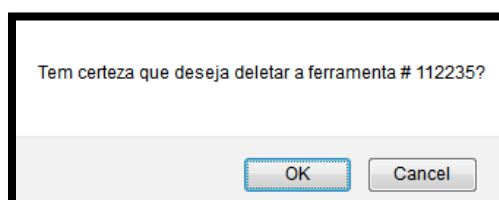


Figura 41 – Confirmação de exclusão de ferramenta.

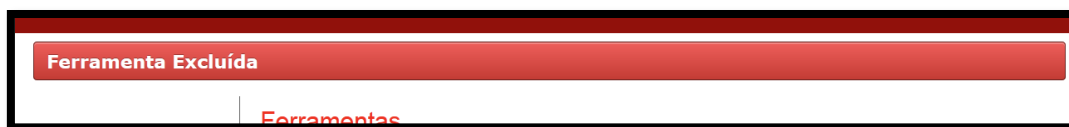


Figura 42 – Mensagem de exclusão com sucesso.

O gerenciamento de usuário se dá de forma análoga ao de ferramentas. No menu principal do administrador é possível acessar as funções de listagem de usuários e adição de novo usuário (métodos *index()* e *add()* do controlador User, respectivamente). Ao selecionar a opção de adicionar novo usuário, uma página com os campos para que os atributos do usuário sejam preenchidos é disponibilizada. Ao final da operação deve-se submeter a inclusão a partir do botão verde ao fim da página. As opções de edição e exclusão estão disponíveis ao lado das informações de cada usuário na página de listagem dos usuários cadastrados. Existe ainda uma segunda forma de editar e excluir um usuário: ao acessar suas informações a partir do *link* “ver”, botões com estas operações são disponíveis na barra lateral esquerda.

Da mesma forma que no caso do gerenciamento de ferramentas, a edição se dá buscando os atributos do usuário e permitindo que sejam feitas mudanças (a interface para esta função é mostrada na Figura 43).

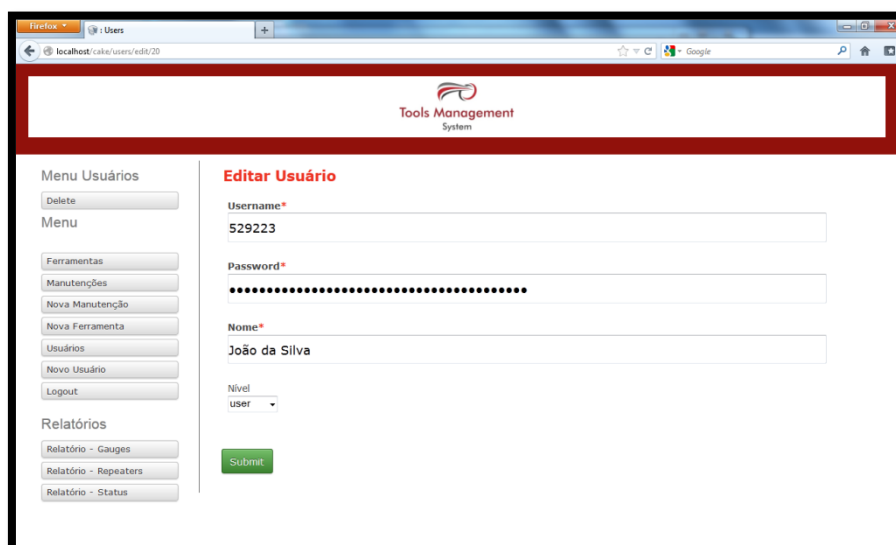


Figura 43 – Edição de usuário.

Foi implementado no sistema uma função que codifica a senha cadastrada a fim de garantir que apenas o usuário tenha acesso à sua conta.

A exclusão, assim como no caso do método *delete()* do controlador Ferramenta, é feita perante confirmação. A janela que é aberta para que seja confirmada a exclusão é mostrada na Figura 44.

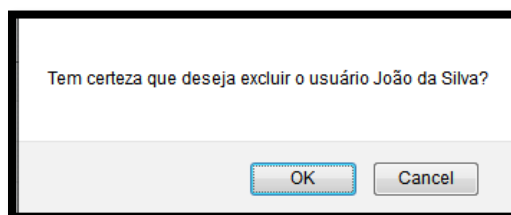


Figura 44 – Confirmação de exclusão de usuário.

5.2.4: Gerenciar Manutenção (F4)

As operações de gerenciamento de manutenções correspondem a um importante requisito do sistema. Este requisito foi desvinculado do F3 pois corresponde à funções abertas à qualquer usuário autenticado.

Existem duas maneiras de se abrir uma nova manutenção no sistema: clicando-se no botão “Nova Manutenção” do menu principal ou então no botão “Abrir Manutenção” na tela de visualização de informações da ferramenta, logo abaixo do histórico de manutenções da ferramenta. Este segundo método trás o diferencial de vincular automaticamente o *equipment number* da ferramenta vista com o formulário de manutenção aberto. O caminho para abertura da folha de manutenção a partir da página de visualização da ferramenta é mostrada na Figura 45.

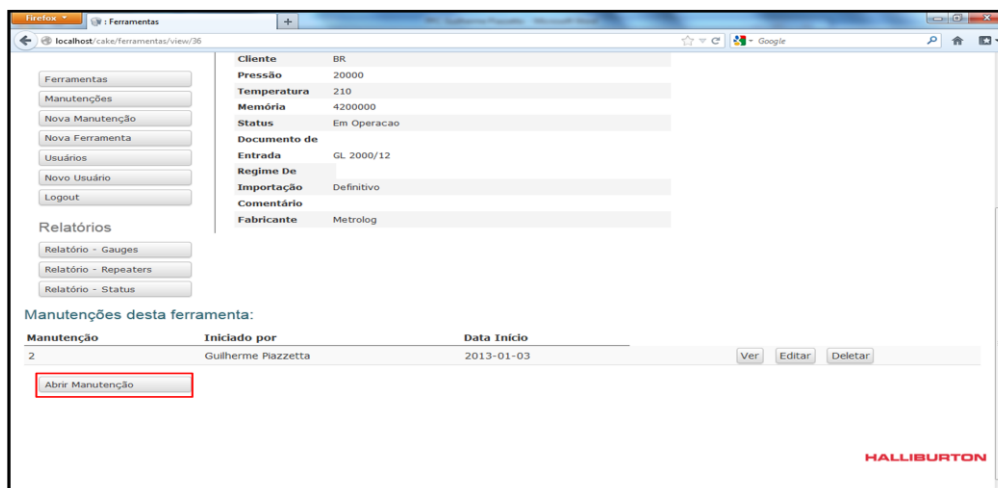


Figura 45 – Lista de manutenções da ferramenta selecionada.

Após selecionar o comando para abertura de nova manutenção, o usuário deve preencher os campos solicitados no formulário e, por fim, clicar no botão verde ao fim da página.

Uma vez que a manutenção pode ser feita em partes, é interessante que haja uma forma de editar uma manutenção criada anteriormente (o método *edit()*). Para que seja aberta a página de edição de manutenção deve-se buscar a manutenção na lista de manutenções ou então nas manutenções relacionadas a certa ferramenta e selecionar o botão “editar” para que o usuário seja redirecionado à uma página semelhante a de abertura de manutenção, porém com campos já preenchidos.

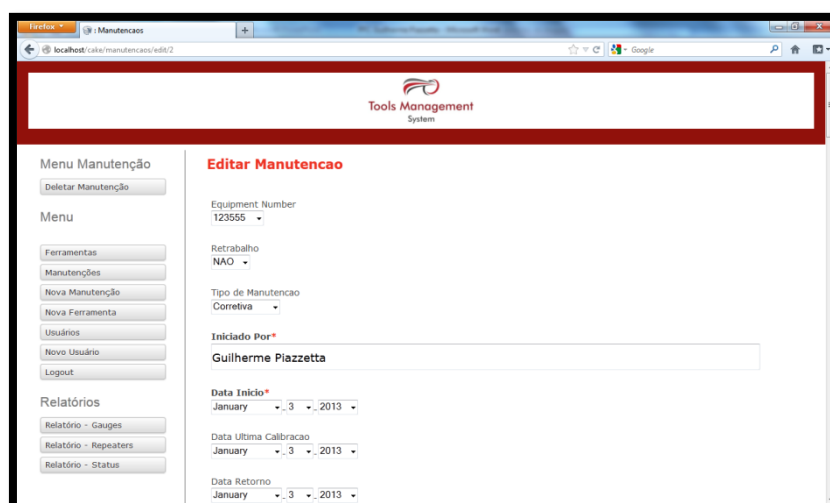


Figura 46 – Página de edição de manutenção.

Para a exclusão de manutenção o procedimento é feito da mesma maneira que nos casos de exclusão de ferramenta ou de usuário. Deve-se buscar a manutenção que se deseja excluir e clicar no botão “Deletar” ao lado, confirmando a ação em seguida na janela mostrada na Figura 47.

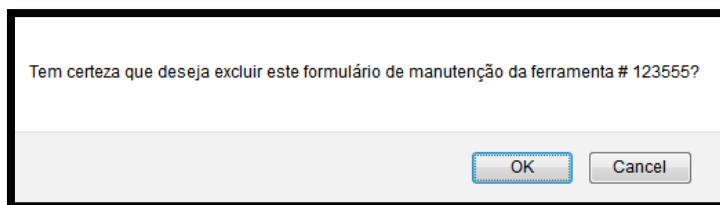


Figura 47 – Confirmação de exclusão de manutenção.

5.2.5: Gerar Relatórios (F5)

O último requisito a ser atendido pelo sistema é talvez o mais importante para os tomadores de decisão do processo. Os relatórios gerados pelo sistema fornecem informações valiosas para que decisões sejam tomadas de forma mais ágil e com maior certeza.

O administrador, ao fazer o *login*, tem em seu menu uma área destinada aos relatórios do sistema. Para que sejam gerados relatórios de manutenção de determinado tipo de ferramenta, basta que o *link* seja selecionado. O relatório de manutenção trás uma lista com todas as ferramentas do tipo selecionado e seu *status* atual. As ferramentas que se encontram em manutenção terão uma indicação da etapa da manutenção em que se encontram. Todo relatório gerado exibe a opção de geração de um arquivo PDF similar à visão mostrada. A página que mostra ao usuário o relatório gerado é mostrada na Figura 48 colocando em detalhe o *link* para geração do documento em PDF.

Gerar PDF
HALLIBURTON

Relatório de manutenção
Data: January 18, 2013
Grupo: Gauge

Equipment Number	Tipo	Status	etapa_1	etapa_2	etapa_3	etapa_4	etapa_5	etapa_6	etapa_7
111111	CGM5	Manutencao	✓	✗	✗	✗	✗	✗	✗
123555	PRM5	Em Operacao							
132465	CGM4	Manutencao	✓	✓	✓	✗	✗	✗	✗
467898	CGM5	Manutencao	✓	✓	✓	✓	✗	✗	✗
987653	2704	Operacional							
963258	iQuartz	Manutencao	✓	✓	✓	✗	✗	✗	✗

Etapas da manutenção dos Gauges Metrolog:

Etapa 1:

- 1) INSPEÇÃO VISUAL DO GAUGE E DO HOUSING DA BATERIA POR DANOS;
- 2) INSPEÇÃO VISUAL DAS ROSCAS DO GAUGE POR DANO;
- 3) LIMPEZA E INSPEÇÃO VISUAL DOS COMPONENTES POR DANOS;
- 4) INSPECIONAR ELETRÔNICA,

Etapa 2:

- 5) EXECUTAR TESTE COM PRESSÃO;

Etapa 3:

- 6) EXECUTAR TESTE DE AQUECIMENTO;

Etapa 4:

- 7) EXECUTAR TESTE DE BANCADA;

Figura 48 – Relatório de manutenção de Gauges (DAS)

Da mesma forma que os relatórios de manutenção, o relatório de *status* possui um *link* de fácil acesso no menu do administrador. Ao ser redirecionado para esta página, o usuário terá a disposição uma tabela que contabiliza os equipamentos cadastrados no sistema e seus respectivos *status*. Na parte superior da página mostrada na Figura 49 – Relatório de *status* de ferramentas é possível gerar um arquivo PDF com estas informações.

Gerar PDF
HALLIBURTON

Relatório de status dos equipamentos:
Data: January 19, 2013

Total de Gauges	Em Manutenção	Operacionais	Em Operação	Investigação
3	1	2	0	0
Total de Repeaters	Em Manutenção	Operacionais	Em Operação	Investigação
1	1	0	0	0
Outros equipamentos	Em Manutenção	Operacionais	Em Operação	Investigação
2	0	2	0	0

HALLIBURTON

Figura 49 – Relatório de *status* de ferramentas.

CAPÍTULO 6: RESULTADOS, CONCLUSÕES E FUTURO DO PROJETO.

Este capítulo final visa descrever o desfecho do desenvolvimento e implantação do sistema na empresa. Os três subcapítulos que compõem esta seção descrevem: os resultados obtidos e a situação atual da implantação do sistema, as conclusões obtidas após este tempo de criação, listando as dificuldades e as condições que propiciaram a implementação do projeto e, por fim, o futuro do projeto e o que ainda pode ser feito para que maiores melhorias sejam buscadas a partir desta primeira versão.

6.1: RESULTADOS

Após a finalização do sistema em termos de funcionalidades e estabilidade, uma versão específica para o subPLS de *Data Acquisition Services* foi desenvolvida para implantação gradual. Este subPLS foi o primeiro a ter o sistema funcional pelo fato de possuir muitos equipamentos e o volume de manutenções ser muito grande, tornando crítica esta atividade e a melhoria proposta imediata. Desde o início do projeto, o supervisor desta área foi o que se mostrou mais interessado no sistema e participou diretamente de sua criação com frequentes reuniões para acompanhamento do andamento do trabalho.

Atualmente o sistema se encontra em fase de testes, funcionando em paralelo com o método manual de preenchimento e arquivamento de folhas de manutenção. Esta fase prevê que os equipamentos sejam cadastrados no sistema e as manutenções sejam abertas juntamente com a impressão dos formulários físicos, como era feito anteriormente.

A etapa também objetiva a adaptação dos colaboradores ao novo método de controle de equipamentos. Sabe-se que uma das maiores barreiras

a se enfrentar na introdução de um novo processo é a carga cultural a se superar e a resistência por parte dos funcionários que estão habituados a fazer desde sempre o processo de uma mesma maneira. Um ponto positivo que pode tornar esta questão menos árdua é o fato da empresa possuir muitos funcionários novos com menos de 1 ano de casa, evitando possíveis vícios que dificultem a introdução de novas práticas e também uma maior motivação para utilizar novas ferramentas que facilitem o trabalho.

A alimentação do banco de dados se deu também de forma gradual, dividindo-se por grupos de equipamentos. Primeiramente foram inseridos no sistema registradores de memória (Gauges), a principal e mais numerosa ferramenta do setor. Após isto foram inseridos os repetidores, baterias e assim por diante. A grande dificuldade desta fase foi o grande volume de trabalho despendido para criar formulários de manutenção distintos para cada grupo.

Passada esta fase de testes, possíveis melhorias de usabilidade serão feitas conforme as primeiras impressões dos usuários responsáveis por testar e alimentar o banco de dados.

6.2: CONCLUSÕES

A Halliburton é uma grande empresa multinacional com considerável investimento interno, ainda assim podem-se identificar diversos processos muito ultrapassados que necessitam de melhoria a partir da automação e digitalização, por exemplo. Durante o período de estágio, várias reuniões com supervisores de todas as áreas foram feitas com o intuito de identificar problemas que caberiam melhorias através de projetos como este.

Algumas propostas foram levantadas e chegou-se à conclusão de que o sistema de gerenciamento de informações de ferramentas se encaixaria nas propostas do PFC e também seria de grande valia para os processos e fluxos de informação do setor de TSS como um todo.

A identificação do problema e das possibilidades de solução foi a primeira dificuldade encontrada. Identificar semelhanças e diferenças entre subPSL's de um mesmo PSL (TSS) foi fundamental para a ideia de um sistema genérico que, ao mesmo tempo, atendesse cada subsetor com suas particularidades sem que fosse necessário grande modificação. Tudo isto deveria ser feito sem custos, uma vez que o programa de estágio não visava a criação deste sistema.

O tempo disponível para o desenvolvimento do sistema também foi um grande obstáculo encontrado. Além do trabalho intenso durante o horário de expediente, o programa trainee visava o aprendizado de todas as áreas. Apresentações e relatórios eram cobrados semanalmente, restando pouco tempo livre destinado exclusivamente ao projeto.

Visto que o tempo destinado apenas ao projeto seria escasso, ferramentas que trouxessem agilidade ao processo de criação foram consideradas importantes. As metodologias de desenvolvimento permitiram que fossem bem definidos requisitos e fluxos de informação, evitando posterior retrabalho. Já o *framework* selecionado ofereceu a possibilidade de reutilização

de códigos de forma que as funções básicas pudessem ser programadas de forma ágil, estruturada e robusta.

Mais uma dificuldade superada foi a inexperiência em relação à programação de sistemas. A solução para este revés foi selecionar ferramentas de programação bem difundidas que ofereciam vasto suporte on-line. Através de tutoriais e *forums* de discussão foi possível avançar e passar por cada problema que o desenvolvimento do sistema apresentou.

Por fim, todas as dificuldades encontradas ao longo do período foram superadas e o sistema foi implementado de forma satisfatória suprimindo todos os requisitos levantados inicialmente. Conforme o supervisor de DAS, o projeto superou as expectativas e, ao cabo desta fase experimental de teste e alimentação do banco de dados, o sistema trará grande melhoria ao processo no que diz respeito à organização de informações e agilidade nas tomadas de decisões.

6.3: FUTURO DO PROJETO

O futuro do projeto anda por duas vertentes. A primeira está diretamente ligada à expansão para os demais subPSL's. Desde o início, o projeto foi concebido para que pudesse ser utilizado em todo o setor, apenas fazendo pequenas alterações nos atributos das entidades.

Esta primeira modificação é mais simples e imediata. Apenas com algumas modificações nos modelos e views o sistema pode ser adaptado para outro subPSL.

A segunda possibilidade de expansão do projeto diz respeito à estrutura do projeto em si. Nas reuniões anteriormente citadas, surgiu a ideia de que trabalhos pudessem ser cadastrados no sistema de forma que ferramentas “em operação” pudessem ser ligadas a determinado trabalho. Isto seria muito importante uma vez que geraria um histórico dos trabalhos de cada ferramenta no sistema. Uma ideia do novo modelo conceitual do sistema já incluída a entidade “trabalhos” é mostrada abaixo na Figura 50.

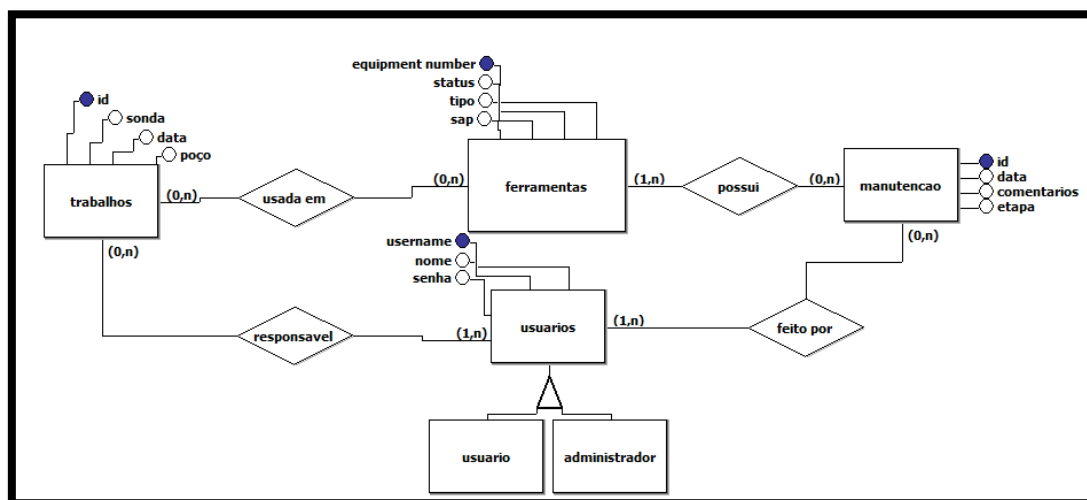


Figura 50 – Modelo conceitual segunda versão do sistema

CAPÍTULO 7: BIBLIOGRAFIA

BNDES. **Relatório III - DESENVOLVIMENTO DA CADEIA PRODUTIVA DE PETRÓLEO**. Banco Nacional do Desenvolvimento. [S.I.]. 2009.

CAKE SOFTWARE FOUNDATION. Entendendo o Model-View-Controller (MVC). **cakePHP**, 2012. Disponível em: <<http://book.cakephp.org/1.3/pt/The-Manual/Beginning-With-CakePHP/Understanding-Model-View-Controller.html>>. Acesso em: Janeiro 2013.

CERQUEIRA, W. D. Combustíveis Fósseis. **Brasil Escola**. Disponível em: <<http://www.brasilecola.com/geografia/combustiveis-fosseis.htm>>. Acesso em: dez. 2012.

CERQUEIRA, W. D. Combustíveis Fósseis. **Brasil Escola**. Disponível em: <<http://www.brasilecola.com/geografia/combustiveis-fosseis.htm>>. Acesso em: 15 dez. 2012.

FAYAD, M.; SCHIMIDT, D. **Object-Oriented Application Frameworks**. New York: [s.n.], 1997.

FOWLER, M. **UML Essencial**. 3ª Edição. ed. [S.I.]: Bookman, 2004.

FERNANDEZ, E.; JUNIOR, O.; PINHO, A. **Dicionário do petróleo em língua portuguesa**. Rio de Janeiro : Lexikon, 2009.

HIGA, P. O que é XAMPP e para que serve. **TechTudo**, 2012. Disponível em: <<http://www.techtudo.com.br/dicas-e-tutoriais/noticia/2012/02/o-que-e-xampp-e-para-que-serve.html>>. Acesso em: Janeiro 2013.

IBGE. **Portal IBGE**, 2010. Disponível em: <http://www.ibge.gov.br/home/presidencia/noticias/noticia_visualiza.php?id_noticia=2161&id_pagina=1>. Acesso em: dez. 2012.

IBGE. **Portal IBGE**, 2010. Disponível em: <http://www.ibge.gov.br/home/presidencia/noticias/noticia_visualiza.php?id_noticia=2161&id_pagina=1>. Acesso em: 15 dez. 2012.

International Directory of Company Histories. [S.I.]: St. James Press, v. Vol. 55, 2003. Acesso em: dez. 2012.

JUHASZ, A. **A Tirania do Petróleo**. [S.l.]: [s.n.], 2009.

LARMAN, C. **Utilizando UML e padrões**. 3ª edição. ed. [S.l.]: Bookman, 2005.

MACÊDO, D. Modelagem Conceitual, Lógica e Física de Dados.

Diego Macêdo – Analista de T.I., 2011. Disponível em: <<http://www.diegomacedo.com.br/modelagem-conceitual-logica-e-fisica-de-dados/>>. Acesso em: Janeiro 2013.

MACÊDO, D. Modelagem Conceitual, Lógica e Física de Dados.

Diego Macêdo – Analista de T.I., 2011. Disponível em: <<http://www.diegomacedo.com.br/modelagem-conceitual-logica-e-fisica-de-dados/>>. Acesso em: 12 Janeiro 2013.

MORAZ, E. **Treinamento Prático em PHP**. [S.l.]: Digerati , 2005.

NEVES, P.; RUAS, R. **O guia prático do MySQL**. Lisboa: Centro Atlantico, 2005.

SCOTT, K. **O Processo Unificado**. [S.l.]: Bookman, 2002.

THOMAS, J. E. **Fundamentos da Engenharia de Petróleo**. [S.l.]: Editora Interciência, 2004.

WAZLAWICK, R. S. **Análise e Projeto de Sistemas de Informação**. Rio de Janeiro: Elsevier, 2004.

WELLING, L.; THOMSON, L. **PHP e MySQL - Desenvolvimento web**. 3ª edição. ed. Rio de Janeiro: Elsevier, 2005.