



**FACULDADE ALCIDES MAYA TECNOLOGIA- AMTEC**  
**CURSO SUPERIOR DE TECNOLOGIA EM SISTEMAS PARA INTERNET**

**THIAGO OSS EMER EIZERIK**

**Módulo de Ordens de Serviços Automobilísticos para Sistema em Delphi**

**Porto Alegre**  
**2020**

THIAGO OSS EMER EIZERIK<sup>1</sup>

SISTEMA INTEGRADO PARA GERAÇÃO DE ORDENS DE SERVIÇOS  
AUTOMOBILÍSTICOS

Projeto de Pesquisa apresentado como requisito parcial para obtenção do título de Tecnólogo em Sistemas para Internet, pelo Curso Superior de Tecnologia em Sistemas para Internet da Faculdade de Tecnologia Alcides Maya - AMTEC

Orientador: Prof. Me. João Padilha Moreira<sup>2</sup>

Porto Alegre

2020

---

<sup>1</sup> Acadêmico do Curso de Sistemas para Internet – email: thiago.eizerik@alcidesmaya.edu.br

<sup>2</sup> Orientador – email joao\_moreira@alcidesmaya.edu.br

# LISTA DE TABELAS

Tabela 2 : Cronograma .....	130
-----------------------------	-----

## LISTA DE FIGURAS

Figura 01: Br Modelo.....	18
Figura 02: Usuário Admin.....	22
Figura 03: Botões das Telas de CRUD .....	23
Figura 04: Exemplo de grid das telas de CRUD.....	24
Figura 05: Exemplo de campos de uma tela de CRUD em modo de visualização....	24
Figura 06: Botões das telas de CRUD quando em modo de visualização .....	24
Figura 07: Exemplo de campos de uma tela de CRUD em modo de inserção ou edição .....	25
Figura 08: Botões das tela de CRUD em modo de inserção ou edição .....	25
Figura 09: Mensagem de confirmação para salvar .....	26
Figura 10: Mensagem de confirmação para cancelar .....	26
Figura 11: Mensagem de confirmação para excluir.....	26
Figura 12: Tentativa de fechar uma tela de CRUD quando em modo de inserção ou edição.....	27
Figura 13: Campo obrigatório.....	27
Figura 14: Mensagem de erro devido a não preenchimento de campo obrigatório...28	
Figura 15: Campo obrigatório recebendo foco por não ter sido preenchido.....	28
Figura 16: Mensagem de erro por tentar excluir registro de tabela pai que está ligado a um registro de uma tabela filho .....	29
Figura 17: Diagrama de Atividades para exclusão de registros .....	29
Figura 18: Dropdown de um campo que faz ligação com uma tabela.....	31
Figura 19: Botão a direita do campo levando para tela de consulta.....	32
Figura 20: Diagrama de Inativação .....	33
Figura 21: Consulta dentro de uma query .....	35
Figura 22: Query sem campos adicionados .....	35
Figura 23: Campos disponíveis para adicionar .....	36
Figura 24: Query com os Campos Adicionados .....	36
Figura 25: dmGeral .....	37
Figura 26: Chamando recursos do dmGeral para uma Tela .....	37
Figura 27: Código para popular uma query .....	37
Figura 28: Conf datasource para conectar a uma query .....	38

Figura 29: Conectando componente a um campo da query.....	38
Figura 30: Conectando um TdataProvider a uma query .....	39
Figura 31: Conectando um TClientDataSet a um TdataProvider .....	39
Figura 32: Conectando um dbGrid ao ClienteDataSet .....	39
Figura 33: Função habilitada da tela OS .....	40
Figura 34: Verificando permissão .....	41
Figura 35: Campos que podem ser alterados dependendo do valor do parâmetro da função .....	41
Figura 36: Campos que nunca podem ser alterados.....	41
Figura 37: Verificando permissão para criar uma nova OS.....	42
Figura 38: Configurando os campos e botões para o modo de inserção .....	42
Figura 39: dsTabela append.....	42
Figura 40: Colocando foco no primeiro campo da tela .....	43
Figura 41: Valores default .....	43
Figura 42: Códigos que realizam e exemplificam as ações explicadas acima .....	44
Figura 43: Códigos que realizam e exemplificam as ações explicadas acima 1 .....	44
Figura 44: Código do botão Fechar onClose e código do evento onClose da tela ....	45
Figura 45: Tela Principal .....	46
Figura 46: Acesso ao módulo de ordens de serviço pela tela principal .....	46
Figura 47: Acesso aos relatórios do módulo de serviços .....	47
Figura 48: Telas do módulo de serviços.....	47
Figura 49: Tela cadastro de serviços .....	48
Figura 50: Grid de serviços da OS.....	50
Figura 51: Grid de itens de serviços com foco no campo código .....	51
Figura 52: Código que chama a tela Consulta de Serviços.....	52
Figura 53: Tela Consulta de Serviços com dropdown das possibilidades de filtros aberto.....	52
Figura 54: Serviço selecionado da tela de consulta de serviços .....	53
Figura 55: Código para geração de novo item de serviço .....	54
Figura 56: Recorte do grid de itens de serviço com um serviço recém adicionado.....	54

Figura 57: Propriedade PickList do campo status do grid de itens de serviço .....	55
Figura 58: Adicionando os status possíveis no PickList .....	55
Figura 59: Campo status já configurado recebendo “APROVADO” .....	56
Figura 60: Trecho do código do evento onCellClick.....	56
Figura 61: Trecho de código do evento onCellClick 1 .....	56
Figura 62: Dropdown do Status ativado com 1 click.....	57
Figura 63: Código de exclusão de itens de serviço .....	57
Figura 64: Código de exclusão de itens de serviço 1 .....	58
Figura 65: Código de exclusão de itens de serviço 2 .....	58
Figura 66: Código de exclusão de itens de serviço 3 .....	59
Figura 67: Pede confirmação da Exclusão .....	59
Figura 68: Itens de Serviço da Tela OS depois da exclusão do item 1 .....	60
Figura 69: Código da função atualizagrid .....	60
Figura 70: Inclusão de um Novo Item de Serviço .....	61
Figura 71: Alteração de um item de serviço .....	62
Figura 72: Exclusão de Itens de serviços através do botão “ Excluir” da OS .....	63
Figura 73: Exclusão de Itens de serviços através do botão “Excluir Itens” da OS ....	63
Figura 74: Tela Grupos de Serviços.....	64
Figura 75: Tela Manutenção de Eventos.....	65
Figura 76: Grid de Itens de Serviços da Tela OS .....	66
Figura 77: Trecho de Código do onCellClick do grid de itens de serviço .....	66
Figura 78: Código para abrir a tela Manutenção de Eventos do onCellClick .....	67
Figura 79: Código onShow da tela Manutenção de Eventos .67	
Figura 80: Tela Manutenção de Eventos de Serviços .....	68
Figura 81: Tela Manutenção de Eventos 1 .....	68
Figura 82: Tela Manutenção de Eventos 2.....	69
Figura 83: Tela Manutenção de Eventos 3.....	70
Figura 84: Código para exclusão de um evento .....	70
Figura 85: Código do campo calculado do cdsOs_serviços ..71	
Figura 86: Último Evento cadastrado em um item de serviço .....	71
Figura 87: Código que salva as Alterações dos Eventos da OS no Banco .....	72
Figura 88: Inclusão de Eventos de Itens de Serviços.....	73
Figura 89: Inclusão de Eventos de Itens de Serviços (continuação) .....	73

Figura 90: Alteração de Eventos de Itens de Serviços .....	74
Figura 91: Exclusão de Eventos de Itens de Serviços .....	74
Figura 92: Tela Cadastro de Veículos .....	75
Figura 93: Mensagem de erro ao tentar sair do campo “Cliente” com ele vazio .....	76
Figura 94: Dropdown mostrando os veículos do cliente selecionado .....	76
Figura 95: Tela Consulta de Veículos mostrando os veículos do cliente selecionado .....	77
Figura 96: Tela Consulta de Veículos mostrando os veículos ativos do sistema .....	77
Figura 97: Mensagem de Alerta .....	78
Figura 98: Informações do veículo mostradas na OS.....	78
Figura 99: Mensagem de Erro.....	78
Figura 100: Tela de Alerta 1 .....	79
Figura 101: Evento onDropDown do Campo Placa .....	79
Figura 102: Evento onKeyPress do campo placa.....	80
Figura 103: Evento onExit do campo placa .....	80
Figura 104: Evento onExit do campo placa 1 .....	81
Figura 105: Função placa.....	82
Figura 106: Código onShow da tela Consulta de Veículos...	82
Figura 107: Evento onClick do checkbox “Todos” .....	83
Figura 108: Evento onKeyPress do grid Consulta de veículos.....	83
Figura 109: Informando um veiculo em uma OS .....	84
Figura 110: Validando um Veiculo em uma OS.....	85
Figura 111: Tela Cadastro de Técnicos .....	85
Figura 112: Dropdown do campo Técnico do grid de itens de serviços .....	86
Figura 113: Técnico selecionado.....	87
Figura 114: Configuração do campo técnico no clientDataSet.....	87
Figura 117: Dropdown do campo técnico aberto em 1 click.....	88
Figura 118: Comissões de Serviços .....	89
Figura 119: Faixas de Faturamento de uma comissão de Serviços .....	89
Figura 120: Diagrama de Atividades da Tela Cadastro de Comissões de Serviços..	90
Figura 121: Tela Cadastro de Seguradoras .....	90
Figura 122: Tela Mensagem de Garantia de Serviço .....	91
Figura 123: Botões de uma OS aberta no modo de visualização.....	92

Figura 124: Status disponíveis na criação de uma OS.....	93
Figura 125: Status disponíveis na modificação de uma OS .....	93
Figura 126: Botões da Parte Inferior.....	94
Figura 127: Botões da Parte Superior .....	94
Figura 127: Mensagem de confirmação para alteração do status de uma OS fechada .....	94
Figura 128: Status Alterado.....	94
Figura 129: Tela de Status .....	95
Figura 130: Código que verifica permissão para fechar uma OS .....	95
Figura 131: Código que dispara mensagem de confirmação para fechar a OS ...	95
Figura 132: Código que altera a situação para fechada e o status para entregue ....	96
Figura 133: Código que desabilita os botões depois do fechamento .....	96
Figura 134: Código que registra o fechamento na tabela log .....	96
Figura 135: Código que verifica a permissão para recuperar uma OS fechada .....	97
Figura 136: Código que verifica se já existe uma OS aberta com o mesmo veículo.	97
Figura 137: Código que atualiza qryos_status .....	98
Figura 138: Código que registra na tabela log a recuperação da OS e quem recuperou .....	98
Figura 138-A: Código de verificação de limite de Credito do Cliente .....	99
Figura 138-B: Código onde são Gerados os títulos.....	100
Figura 139: Menu do módulo de OS .....	100
Figura 140: Tela Codigo para abetura de um Relatorio do modulo de ordens de serviços .....	101
Figura 141: Page Header .....	101
Figura 142: Código do Page Header .....	102
Figura 141-A: Itens e eventos da OS .....	104
Figura 141-A: Exemplo do Código de Aplicação dos Filtros em uma Consulta .....	104
Figura 143:Relatório de OS Geradas .....	105
Figura 144: Header do relatório de OS Geradas .....	107
Figura 145: Relatório Analítico Relatório de OS Geradas .....	108
Figura 146: Relatório Analítico Relatório de OS Geradas mensagens .....	108
Figura 147: Relatório Analítico Relatório de OS Geradas itens de serviço .....	109
Figura 148: Relatório Analítico Relatório de OS Geradas peças.....	109



Figura 149: Relatório Analítico Relatório de OS Geradas materiais de consumo ...	109
Figura 150: Relatório Analítico Relatório de OS Geradas total da OS .....	109
Figura 151: Relatório Analítico Relatório de OS Geradas totais gerais da OS ....	110
Figura 152: Relatório de OS Geradas Relatório Resumido .....	110
Figura 153: Filtro por data do relatório de OS Abertas.....	111
Figura 154: Relatório OS Abertas Relatório Resumido .....	111
Figura 155: Filtro por data do Relatório de OS Fechadas .....	112
Figura 156: Relatório de Carga Horária de OS .....	112
Figura 157: Impressão do relatório de Carga Horária de OS .....	113
Figura 158: Impressão do relatório de Carga Horária de OS totalização de horas do período .....	113
Figura 159:Relatório Carga Horária DIA .....	115
Figura 160: Definindo item por item qual data será filtrada .....	115
Figura 161:Impressão do relatório de Carga Horária/Dia.....	116
Figura 162: Recuperando os eventos de cada item de serviço.....	116
Figura 163:Trecho da Consulta que Calcula os Minutos.....	116
Figura 164:Trecho da Consulta que agrupa data .....	117
Figura 165:Relatório de OS por Cliente .....	117
Figura 166: Fluxo de Placas.....	118
Figura 167: Relação de Placas .....	119
Figura 168: Relação de OS por Placas Relatório Analítico .....	120
Figura 169: Resumo de serviços executados.....	121
Figura 170: Impressão do Relatório de Resumo de serviços executados.....	121
Figura 171: Relatório de OS por Técnicos .....	122
Figura 172: Impressão do Relatório de OS por Técnicos.....	123
Figura 173: Impressão do Relatório de OS por Técnicos total geral .....	123
Figura 174: Previsão de Entrega de OS.....	123
Figura 175: Estrutura da Previsão de Entrega de OS .....	124
Figura 176: Tela Manutenção de OS .....	125
Figura 177: Menu Superior da tela Manutenção de OS .....	126
Figura 178: Filtros da Impressão da OS.....	126
Figura 179: Impressão da OS .....	127
Figura 180: Impressão do Relatório 1 .....	128

## LISTA DE SIGLAS

ABNT	Associação Brasileira de Normas Técnicas
CRUD	Create, Read, Update, Delete (criação, leitura, edição e deleção)
UML	Unified Modeling Language (linguagem de modelagem unificada)
OS	Ordem de Serviço
SAV	Sistema de Administração e Vendas
ERP	Enterprise Resource Planning (planejamento de recursos empresariais)

## SUMÁRIO

<b>HISTÓRICO .....</b>	<b>14</b>
<b>1 INTRODUÇÃO .....</b>	<b>15</b>
<b>1.2 Definição do Tema ou Problema .....</b>	<b>16</b>
<b>1.3 Objetivos .....</b>	<b>16</b>
<b>1.3.1 Objetivo Geral .....</b>	<b>16</b>
<b>1.3.2 Objetivos Específicos .....</b>	<b>16</b>
<b>1.4 justificativa.....</b>	<b>17</b>
<b>2 REVISÃO BIBLIOGRÁFICA.....</b>	<b>17</b>
<b>3 METODOLOGIA .....</b>	<b>17</b>
<b>3.1 Princípios Adotados para o Desenvolvimento do Módulo .....</b>	<b>18</b>
<b>3.1.1 Agilidade .....</b>	<b>19</b>
<b>3.1.3 Confiabilidade.....</b>	<b>19</b>
<b>3.1.4 Organização.....</b>	<b>19</b>
<b>3.1.5 Gestão .....</b>	<b>20</b>
<b>3.2 Delphi .....</b>	<b>20</b>
<b>3.3 Permissões para Usuários do Sistema.....</b>	<b>22</b>
<b>3.4 UML do Usuario Admin.....</b>	<b>22</b>
<b>3.5 Padrões do Sistema .....</b>	<b>23</b>
<b>3.5.1 Layout das Telas de CRUD: .....</b>	<b>23</b>
<b>3.5.2 Regras de Funcionamento das Telas de CRUD quando em modo de Navegação: .....</b>	<b>24</b>
<b>3.5.3 Regras de Funcionamento das Telas de cadastro e modificação quando em modo de edição:.....</b>	<b>24</b>
<b>3.5.4 Mensagens de Confirmação .....</b>	<b>26</b>
<b>3.5.5 Mensagens de erro e validação .....</b>	<b>27</b>
<b>28</b>	
<b>3.5.6 Diagrama de Exclusão de Registro .....</b>	<b>29</b>
<b>3.6 Atalhos do Teclado: .....</b>	<b>30</b>
<b>3.7 Ao abrir uma tela ligada a uma tabela do banco: .....</b>	<b>30</b>
<b>3.8 Ligações com outras tabelas dentro de uma tela: .....</b>	<b>31</b>
<b>3.9 Tabelas Planejadas, de acordo com avaliações prévias, para ter muitos registros:.....</b>	<b>31</b>

<b>3.10 Tabelas Planejadas, de acordo com avaliações prévias , para ter muitos registros:.....</b>	<b>31</b>
<b>3.11 Telas de Consulta.....</b>	<b>32</b>
<b>3.12 Ativo / Inativo.....</b>	<b>33</b>
<b>3.13 Padrões de Código usados no Módulo de OS.....</b>	<b>34</b>
<b>3.13.1 Prefixos para os componentes principais usados no Sistema.....</b>	<b>34</b>
<b>3.13.2 Funcionalidades dos Componentes Principais Usados no Sistema.....</b>	<b>34</b>
<b>3.13.3 Conexão com o banco.....</b>	<b>35</b>
<b>3.13.4 Habilitar ou Desabilitar Botões e Campos:.....</b>	<b>40</b>
<b>41</b>	
<b>3.13.5 Colocando a Tela em modo de Inserção (ao clicar em novo):.....</b>	<b>42</b>
<b>3.13.6 Colocando a Tela em modo de Inserção (ao clicar em novo) 1:.....</b>	<b>43</b>
<b>3.13.7 Quando a tela entra no modo de navegação.....</b>	<b>44</b>
<b>3.13.8 Botões de Navegação.....</b>	<b>45</b>
<b>3.13.9 Botão Fechar.....</b>	<b>45</b>
<b>3.14 Tela Principal do Sistema.....</b>	<b>46</b>
<b>3.15 Tela Cadastro de Serviços.....</b>	<b>48</b>
<b>3.16 Características e funcionalidades básicas da tela Grupos de Serviços..</b>	<b>48</b>
<b>3.17 Características específicas da tela.....</b>	<b>49</b>
<b>3.18 Itens de Serviço da OS.....</b>	<b>50</b>
<b>3.19 Características e funcionalidades básicas do grid de Itens de Serviços</b>	<b>50</b>
<b>3.20 Decisões relevantes tomadas durante o desenvolvimento da parte de itens de serviço.....</b>	<b>60</b>
<b>3.21 Características e funcionalidades básicas da tela.....</b>	<b>64</b>
<b>3.22 Motivos para o desenvolvimento da tela.....</b>	<b>64</b>
<b>3.23 Funcionalidades básicas da tela Manutenção de Eventos.....</b>	<b>66</b>
<b>3.24 Funcionamento da Tela Manutenção de Eventos.....</b>	<b>66</b>
<b>3.25 Motivos e decisões tomadas durante o desenvolvimento da tela Manutenção de Eventos.....</b>	<b>72</b>
<b>3.26 Características e funcionalidades básicas da tela Cadastro de Veículos</b>	
<b>75</b>	
<b>3.27 Particularidades da Tela Cadastro de Veiculos.....</b>	<b>75</b>
<b>3.28 Informando ou Alterando um Veiculo em uma OS.....</b>	<b>76</b>

3.29	Códigos da Parte de Veículos da OS .....	79
3.30	Características e funcionalidades básicas da tela Cadastro de Técnicos 86	
3.31	Escolhendo um técnico para um item de serviço em uma OS .....	86
3.32	Funcionalidades e características básicas da tela Cadastro de Seguradoras .....	90
3.33	Características e Funcionalidades da Tela Cadastro de Mensagem da OS 92	
3.34	Status e Situação da OS .....	92
3.34.1	Características básicas da situação e status da OS.....	92
3.35	Integração com a parte de Peças .....	99
3.35.1	Integração com a Parte de Contas a Receber .....	99
3.36	Relatórios de Modulo de Ordens de Serviço .....	100
3.36.1	Abertura dos Relatórios da Ordem de Serviços.....	100
3.37	Padrões de Codificação dos Relatórios .....	101
3.37.1	Relatórios de OS Geradas .....	105
3.37.2	Estrutura do Relatório .....	106
3.38	Relatorio Analítico de OS Geradas .....	107
3.39	Relatório Resumido de OS Geradas .....	110
3.40	Relatório de OS Abertas .....	111
3.41	Relatório de OS Fechadas .....	111
3.42	Carga Horária de OS.....	112
3.43	Filtros do Relatório.....	113
3.44	Estrutura do Relatório.....	113
3.45	Impressão do Relatório.....	113
3.46	Consulta do Relatório .....	114
3.47	Total de Carga Horaria /DIA .....	115
3.48	Filtros do Relatório.....	115
3.49	Estrutura do Relatório.....	115
3.50	Impressão do Relatório.....	116
3.51	Consulta do Relatório .....	116
3.52	Filtros do Relatório.....	117
3.53	Estrutura do Relatório Analítico.....	118
3.54	Estrutura do Relatório Fluxo de Placas.....	118

3.55	Relatório Analítico .....	118
3.56	Fluxo de Placas .....	118
3.57	Relação de OS por Placas .....	119
3.58	Filtros do Relatório.....	119
3.59	Estrutura do Relatório Analítico.....	119
3.60	Estrutura do Relatório.....	119
3.61	Relatório Analítico .....	120
3.62	Fluxo de Placas .....	120
3.63	Resumo de Serviços Executados .....	121
3.64	Filtros do Relatório.....	121
3.65	Estrutura.....	121
3.66	Relação de OS Técnicos .....	122
3.67	Filtros do Relatório.....	122
3.68	Estrutura.....	122
3.69	Impressão do Relatório.....	123
3.70	Filtros do Relatório.....	124
3.71	Estrutura.....	124
3.72	Impressão do Relatório.....	124
3.73	Tela Manutenção de OS .....	125
3.74	Características Principais da Tela .....	125
3.75	Exemplo de Geração de Uma OS .....	126
5	VALIDAÇÃO .....	129
6	CONCLUSÃO .....	129
7	CRONOGRAMA .....	130
8	REFERÊNCIAS BIBLIOGRÁFICA .....	131
	Lexos ERP omnichannel. Sistema de Ordem de Serviço Flexível. 2020. < <a href="https://www.lexos.com.br/sistema-ordem-de-servico/">https://www.lexos.com.br/sistema-ordem-de-servico/</a> >. Acessado em Acessado em 8 junho 2020. ....	131
	Bom Controle. O que contém uma ordem de serviço. 2020. < <a href="https://bomcontrole.com.br/ordem-de-servico/">https://bomcontrole.com.br/ordem-de-servico/</a> >. Acessado em Acessado em 8 junho 2020.....	131

## **HISTÓRICO**

A empresa Mba Sistema, que requisitou o desenvolvimento deste trabalho, é especializada a quase de 30 anos no setor automotivo.

A primeira versão do SAV (sistema de administração e vendas), foi desenvolvido com a linguagem cliper em uma plataforma 16 bits, já trabalhava com foco em lojas de autopeças a 23 anos e passou a trabalhar com ordens de serviço a 20 anos, tendo atualmente 55 clientes que trabalham com geração de ordens de serviço através do sistema antigo.

Devido as limitações da linguagem e necessidade de mais recursos, a 7 anos, foi desenvolvida uma nova versão, mais completa, em Delphi. Essa nova versão possuía mais recursos no que se refere a administração de lojas, restaurantes e mercados e conseguiu atender um perfil de cliente mais exigente mas não possuía um módulo de ordens de serviço.

A maioria dos clientes já foram migrados para o sistema novo mas, devido principalmente à falta desse módulo, ainda existem muitos clientes no sistema antigo que dependem deste para a geração de ordens de serviço e muitos outros clientes potenciais que trabalham com ordens de serviço e cujo sistema antigo não atende as necessidades.

A experiência adquirida ao longo dos anos pela empresa no setor automobilístico, o network com empresas do setor e o próprio sistema antigo, foram os principais suportes sobre os quais foi desenvolvido o módulo de OS para o sistema novo.

## **1 INTRODUÇÃO**

A proximidade da empresa Mba Sistemas com o setor automobilístico, permitiu que ela identificasse a existência de muitas oficinas, especializadas ou não em geração de ordens de serviço, que não possuem sistemas automatizados que contemplem a gestão de ordens de serviços automotivo e tampouco tem como prioridade investir em sistemas complexos e custosos. Contudo, à medida que essas empresas crescem, se torna cada vez mais difícil administrar sem tal recurso, e estas empresas tendem a buscar soluções mais especializadas e de baixo custo.

Foi visando esse nicho de mercado e aproveitando sua experiência e proximidade com o setor de serviços automotivos, que a empresa Mba Sistemas decidiu focar no desenvolvimento de um módulo especializado na geração e gestão de ordens de serviços automotivos para o SAV novo, um módulo completo e adaptado as demandas de um mercado cada vez mais exigente.

Antes e durante o desenvolvimento, foram estudados sistemas similares, materiais sobre o tema e realizadas pesquisas de campo com clientes potenciais e clientes do sistema antigo, visando integrar em um único ambiente as melhores características de vários sistemas e atender da melhor forma as expectativas e necessidades dos usuários. Buscou-se sempre tornar o módulo o mais abrangente em recursos e informações, mais intuitivo e a prova de erros de usuários que pudessem comprometer a confiabilidade do sistema.



## **1.2 Definição do Tema ou Problema**

Esse projeto tratará, da implantação de um módulo de geração de ordens de serviço para o sistema Sav, utilizando a linguagem Delphi e o banco de dados PostgreSQL.

## **1.3 Objetivos**

Realizar o desenvolvimento de toda a parte referente a ordens de serviço, relatórios relacionados e integrar com a parte de peças, com os clientes, com os vendedores com a parte de contas a pagar contas a receber.

### **1.3.1 Objetivo Geral**

Desenvolver um sistema simples e de baixo custo, que contemple a gestão de ordens de serviços automotivos, voltado a atender uma carência do setor, particularmente de pequenas oficinas especializadas ou não, que não possuem sistema de gestão.

### **1.3.2 Objetivos Específicos**

Pesquisar e reunir informações sobre oficinas de manutenção automotiva, referentes as operações realizadas nelas, a sistemas que contemplam a execução de seus processos e a geração de relatórios para sua gestão.

Definir estratégias para suprir de forma eficiente as necessidades do setor, ao mesmo tempo evitando tornar o sistema excessivamente complexo para o usuário.

Desenvolver um sistema para suprir as necessidades de oficinas de manutenção automotiva de porte pequeno e médio.

## **1.4 justificativa**

A motivação desse projeto foi uma demanda da empresa Mba Sistemas pelo desenvolvimento de um módulo de geração de ordens de serviço em seu sistema novo, o Sav.

## **2 REVISÃO BIBLIOGRÁFICA**

O presente projeto irá discorrer sobre as necessidades do setor de ordens de serviço automotivos, os padrões utilizados no layout das telas e sua programação, sobre as escolhas durante o desenvolvimento e como elas contribuem para atender as necessidades do setor e sobre algumas ferramentas do ambiente Delphi utilizadas durante a programação.

## **3 METODOLOGIA**

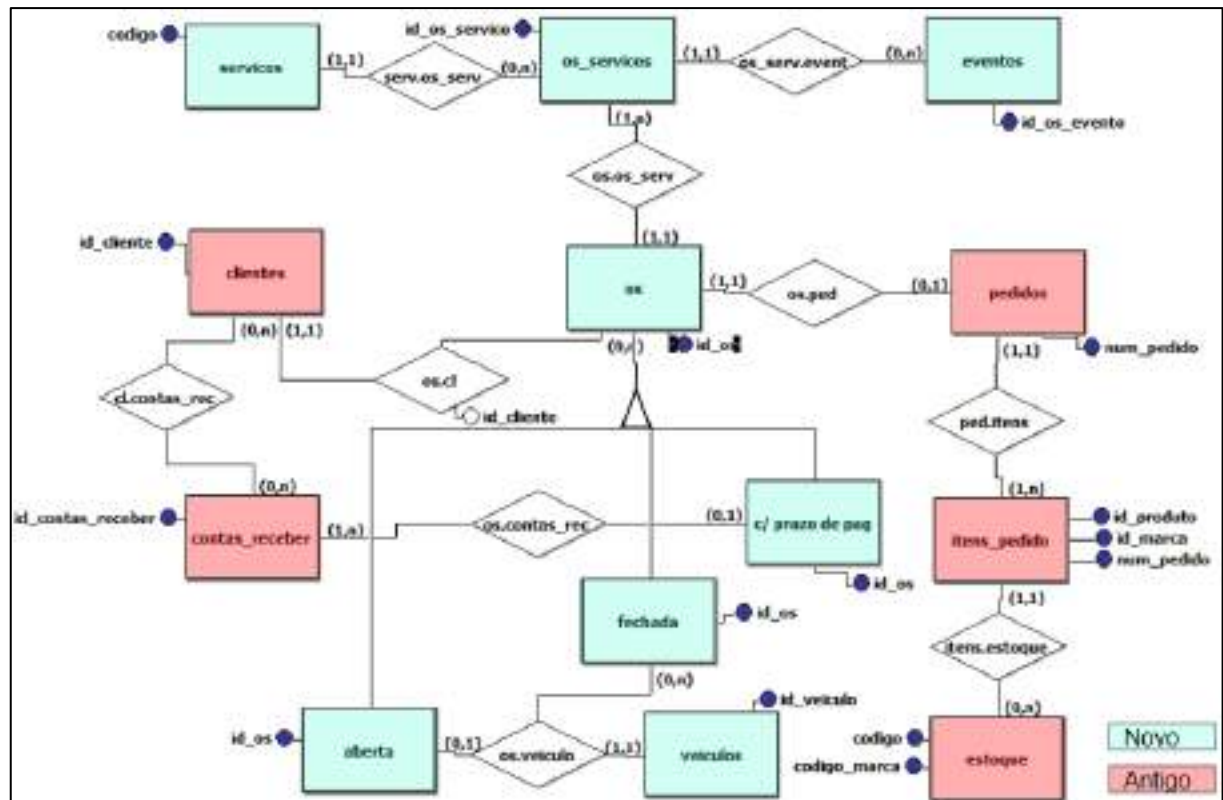
Primeiramente realizou-se um estudo e trabalho de desenvolvimento encima dos demais módulos do sistema, para serem definidos os padrões de layout, padrões de funcionamento das telas e padrões de codificação utilizados.

A segunda parte deste trabalho foi desenvolvida através da pesquisa exploratória em sistemas de geração de ordens de serviços automobilísticos.

Essa pesquisa serviu para compreender as estruturas de dados, regras de negócio, processos executados, interação com o usuário e relatórios para a gestão que podem existir em sistemas similares. Essas pesquisas foram realizadas antes e durante o desenvolvimento do módulo de ordens de serviço.

Todo o desenvolvimento do projeto foi realizado em Delphi, que é um ambiente de desenvolvimento integrado.

Figura 1: Br Modelo



Fonte: Produzida pelo autor

### 3.1 Princípios Adotados para o Desenvolvimento do Módulo

Os principais benefícios de um sistema de geração de ordens de serviço para uma oficina são:

- **Agilidade:** a sistematização das tarefas facilita a rotina de todos os envolvidos no processo, que ocorre de forma mais independente, não sendo necessário que a solicitação se torne um trâmite entre os diferentes cargos hierárquicos;
- **Organização:** Com um sistema organizado é muito mais fácil manter a empresa organizada;
- **Confiabilidade:** impermeabilização contra erros de usuários e maior segurança dos dados o sistema confiável;
- **Gestão:** saber as demandas do cliente de forma detalhada é um recurso poderoso para organizar prazos e equipes e melhorar a lucratividade.

O Módulo de OS do sistema SAV foi desenvolvido tendo como prioridade focar esses aspectos.

### **3.1.1 Agilidade**

O sistema foi desenvolvido com grande ênfase em tornar os processos mais rápidos para o usuário, mas sem comprometer a organização.

O contato com algumas oficinas mecânicas e o fato do sistema ser especializados nesse nicho facilitou o desenvolvimento de processos rápidos e organizados de cadastros e manutenções, sem excesso de informações irrelevantes ou falta de informações importantes.

Com um atendimento mais eficaz e rápido, a OS aumenta a produtividade.

### **3.1.3 Confiabilidade**

A própria empresa MBA Sistemas tem um longo histórico de confiabilidade no mercado, resultado de um cuidado permanente com a proteção dos dados.

No módulo de OS esse padrão foi mantido, sempre impedido a entrada de qualquer valor inválido, criando permissões especiais para cada operação e avisando o usuário quando algum valor está fora do normal.

### **3.1.4 Organização**

O módulo de OS permite um nível máximo de organização durante a realização de cada atividade no sistema.

É fornecido um treinamento para o uso do sistema para os usuários onde é ensinada a forma mais correta de utilizar o sistema.

Apesar disso o usuário tem liberdade para realizar alguns processos de forma mais rápida sem passar por todos os estágios remendáveis, dessa forma alguns relatórios não vão ficar imprecisos devido a falta de informações importantes.

O sistema fornece todas as ferramentas para uma gestão organizada da oficina, mas foi entendido que o usuário pode precisar realizar uma operação de rápida e o sistema não restringe isso.

### 3.1.5 Gestão

O módulo todo foi desenvolvido de forma a permitir o cruzamento de informações relevantes, de forma que foi possível gerar relatórios que mostram, de maneira geral ou específica, diversos ângulos da oficina.

Existem atualmente 13 telas de relatórios, cada uma podendo gerar 2 relatórios diferentes e com diversos filtros aplicados.

### 3.2 Delphi

A linguagem Delphi possui um mercado bem ativo, muitas empresas utilizam aplicações em Delphi, porém você pode aprender outras linguagens para aumentar o seu conhecimento e agregar mais de uma tecnologia para ampliar o seu espaço no mercado de trabalho. Outro ponto importante é verificar o índice da TIOBE (<https://www.tiobe.com/tiobe-index/>) que mede o impacto de uma tecnologia no mercado, e veja também o guia de carreiras aqui a Devmedia.

- O Delphi é uma combinação de linguagem de programação e Software Development Kit (SDK), que permite o desenvolvimento de aplicativos multiplataforma para desktop, mobile, web, nuvem e ampla conectividade IoT;
- O Delphi é uma linguagem de programação relativamente simples, com sintaxe de código bem “clean” e de fácil entendimento;
- O código escrito em Delphi é facilmente legível. Por exemplo, você pode concatenar textos (strings) utilizando o sinal “+” ao invés de ter que utilizar uma função;
- A documentação do Delphi é bastante ampla e bem organizada, o que ajuda muito quem está iniciando;
- A IDE do Delphi é bastante integrada e permite que você facilmente arraste e solte componentes para criar seus formulários;
- O Delphi suporta teste em tempo real, o que faz com que você facilmente encontre e corrija erros de programação;

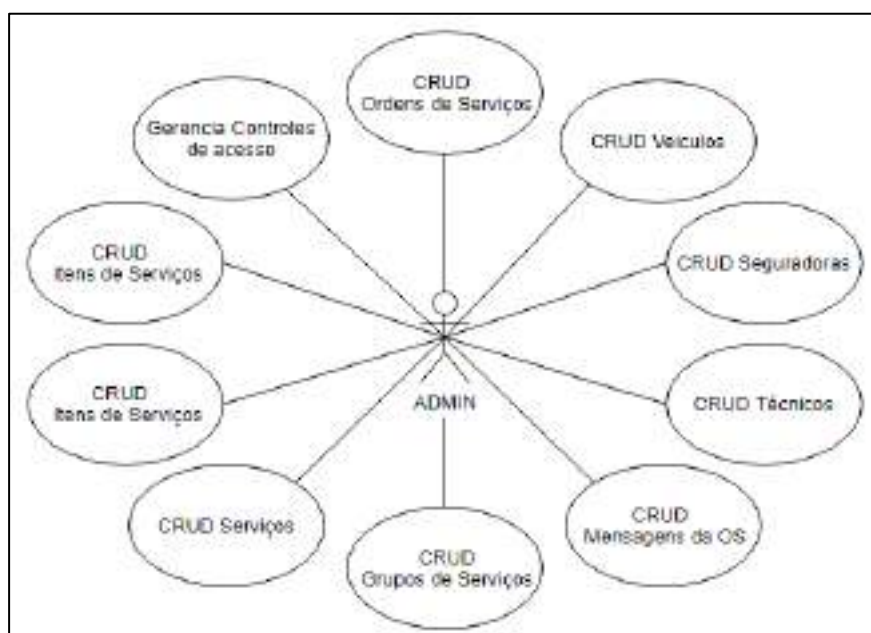
- O Delphi possui o modelo de Desenvolvimento Rápido de Aplicação (RAD – Rapid Application Development), o que facilita e agiliza muito no andamento do projeto;
- O Delphi suporta arquitetura cliente-servidor e bancos de dados SQL;
- O Delphi suporta API completa do Windows;
- O Delphi permite a criação de componentes de fácil integração no IDE.

### 3.3 Permissões para Usuários do Sistema

As permissões são individualizadas, ou seja, cada operação em cada tela que exige permissão é criada individualmente para cada usuário pelo Admin., que tem todas as permissões.

### 3.4 UML do Usuario Admin

Figura 2: Usuário Admin



Fonte: Produzida pelo autor

### 3.5 Padrões do Sistema

O módulo de OS que foi desenvolvido faz parte de um sistema maior, portando teve de se adaptar a alguns padrões que já eram adotados no sistema antigo, por exemplo.

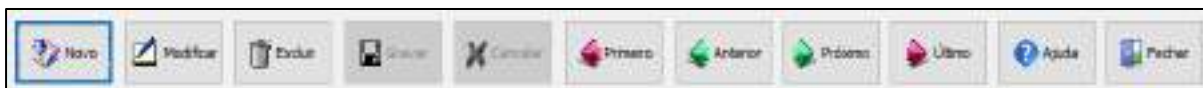
#### 3.5.1 Layout das Telas de CRUD:

Todos os campos das telas tem uma “label” em cima identificando o campo.

Contem uma ficha principal, onde se encontra pelo menos todos os campos da tabela a qual a tela está ligada, as vezes trazendo também informações de outras tabelas;

Contem pelo menos os botões novos, modificar, excluir, salvar, cancelar, primeiro, anterior, próximo, último e fechar nessa ordem, Conforme a figura abaixo.

Figura 3: Botões das Telas de CRUD



Fonte: Produzida pelo autor

Contem uma grade, na extrema direita das abas que contem os campos da tela, com um grid que mostra pelo menos todos os registros e todos os campos (com um título alterado para ser estético e compreensivo), exceto id, da tabela a qual a tela está ligada, as vezes trazendo também informações de outras tabelas;

Na grade, acima do grid, tem sempre um argumento de pesquisa que vai filtrando os registros mostrados no grid à medida que o usuário escreve nele;

As telas filtram os registros da grade por apenas por um campo, normalmente descrição ou nome;



Figura 4: Exemplo de grid das telas de CRUD

Descrição	Tipo	Data de Cadastro Horas	Minutos	Custo	Preço	Tabela ICM
▶ TROCA DE PNEU	PROPRIO	08/01/2020	1	45	12,00	50,00
TROCA DO FREIO	PROPRIO	08/01/2020	4	45	43,00	500,00

Fonte: Produzida pelo autor

### 3.5.2 Regras de Funcionamento das Telas de CRUD quando em modo de Navegação:

Todos os campos ligados a uma tabela do banco ficam desativados, de forma que o usuário não consegue clicar sobre eles. Conforme a figura abaixo.

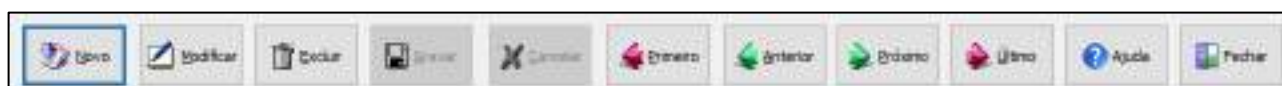
Figura 5: Exemplo de campos de uma tela de CRUD em modo de visualização

Placa	Fabricante	Modelo	Cor	Ano Fab/Mod
TTT2000	FORD	F5	PRETO	2010/2010
Combustível	Chassi	Potência	Cliente	
ALCOOL		HP	AGLAE T DE ABAUJO ME	
Ativo				
SIM				

Fonte: Produzida pelo autor

O botão “Novo”, “Modificar”, “Excluir”, “Fechar” e botões de navegação ficam ativos, assim como todas as páginas da tela. Já os botões “gravar” e “cancelar” ficam inativos, de forma que o usuário não pode clicar sobre eles; Conforme a figura abaixo.

Figura 6: Botões das telas de CRUD quando em modo de visualização



Fonte: Produzida pelo autor

### 3.5.3 Regras de Funcionamento das Telas de cadastro e modificação quando em modo de edição:

Quando entra em modo de edição o foco vai para a primeira página da tela e o primeiro campo ativo de cima para baixo e da esquerda para a direita;

A maioria dos campos ligados a tabela do banco ficam ativos, exceto aqueles que não podem ser modificados depois de criados, aqueles são preenchidos via sistema, aqueles que dependem do valor de outro campo e aqueles que trazem informações de outras tabelas;

Conforme a figura abaixo.

Figura 7: Exemplo de campos de uma tela de CRUD em modo de inserção ou edição

Placa KKK1234	Fabricante RENO	Modelo V	Cor PRETO	Ano Fab/Mod 2002/2005
Combustível DIESEL	Chassi	Potência HP	Cliente	
Ativo SIM				

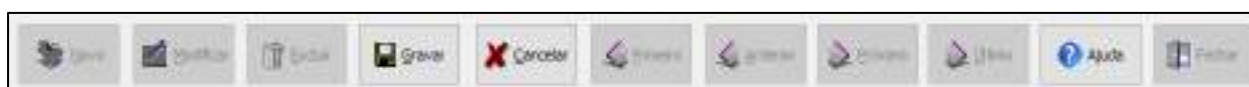
Fonte: Produzida pelo autor

Os botões “salvar” e “cancelar” ficam ativos e os botões “novo”, “modificar”, “excluir”, “fechar” e botões de navegação ficam inativos, de forma que o usuário não pode clicar sobre eles;

O usuário pode navegar entre os campos usando o “enter” ou “tab” para ir para o próximo campo ou usar o mouse para ir para qualquer campo ou botão;

Não é possível alterar registros pelo Grid. Conforme a figura abaixo.

Figura 8: Botões das tela de CRUD em modo de inserção ou edição



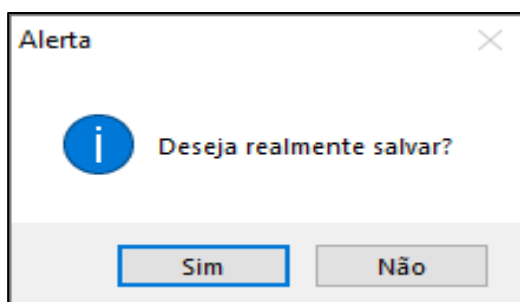
Fonte: Produzida pelo autor

### 3.5.4 Mensagens de Confirmação

Para as operações de salvar, cancelar e excluir, de todas as tabelas, é requisitado pelo sistema uma confirmação com botões de sim e não.

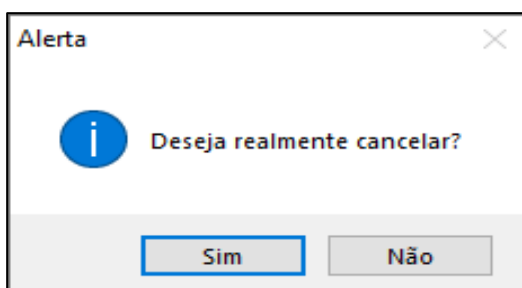
Conforme a figura abaixo.

Figura 9: Mensagem de confirmação para salvar



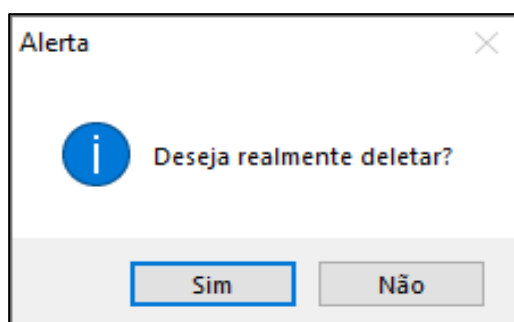
Fonte: Produzida pelo autor

Figura 10: Mensagem de confirmação para cancelar



Fonte: Produzida pelo autor

Figura 11: Mensagem de confirmação para excluir

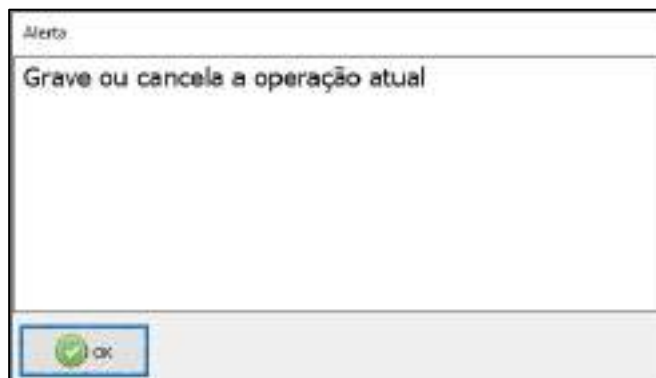


Fonte: Produzida pelo autor

### 3.5.5 Mensagens de erro e validação

Caso o usuário clique em “fechar” ou no “X” da tela e a tela estiver no modo de inserção ou navegação, aparece a mensagem de erro: “Grave ou cancele a operação antes de fechar a tela”. Conforme a figura abaixo.

Figura 12: Tentativa de fechar uma tela de CRUD quando em modo de inserção ou edição



Fonte: Produzida pelo autor

Alguns campos obrigatórios são validados no próprio campo, de forma que o usuário não consegue sair do campo sem preenchê-lo de forma correta;

Por exemplo o campo da figura abaixo.

Figura 13: Campo obrigatório

A imagem mostra um formulário com um campo de texto. O rótulo do campo é "Descricao" em negrito. O campo de texto contém o texto "ELETRICA".

Fonte: Produzida pelo autor

Caso um ou mais campos obrigatórios não tenham sido preenchidos, ao clicar em salvar, aparece uma mensagem indicando o primeiro dos campos obrigatórios não preenchidos e, ao clicar em “ok” o cursor foca nesse campo. Conforme exemplificado pelas figuras abaixo.

Figura 14: Mensagem de erro devido a não preenchimento de campo obrigatório



Fonte: Produzida pelo autor

Clicando em “ok” o foco vai para primeiro campo que não passou na validação. Conforme a figura abaixo.

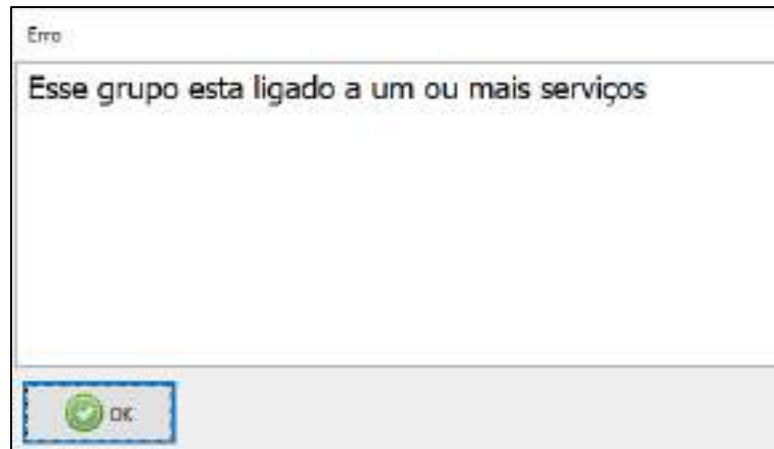
Figura 15: Campo obrigatório recebendo foco por não ter sido preenchido



Fonte: Produzida pelo autor

Salvo uma exceção, quando o usuário tenta excluir um registro de uma tabela pai que está ligado a um registro de uma tabela filho via, o sistema bloqueia e manda a mensagem de erro: “Esse (nome dos filhos da tabela) já é utilizado em outra tabela”. Conforme a figura abaixo.

Figura 16: Mensagem de erro por tentar excluir registro de tabela pai que está ligado a um registro de uma tabela filho

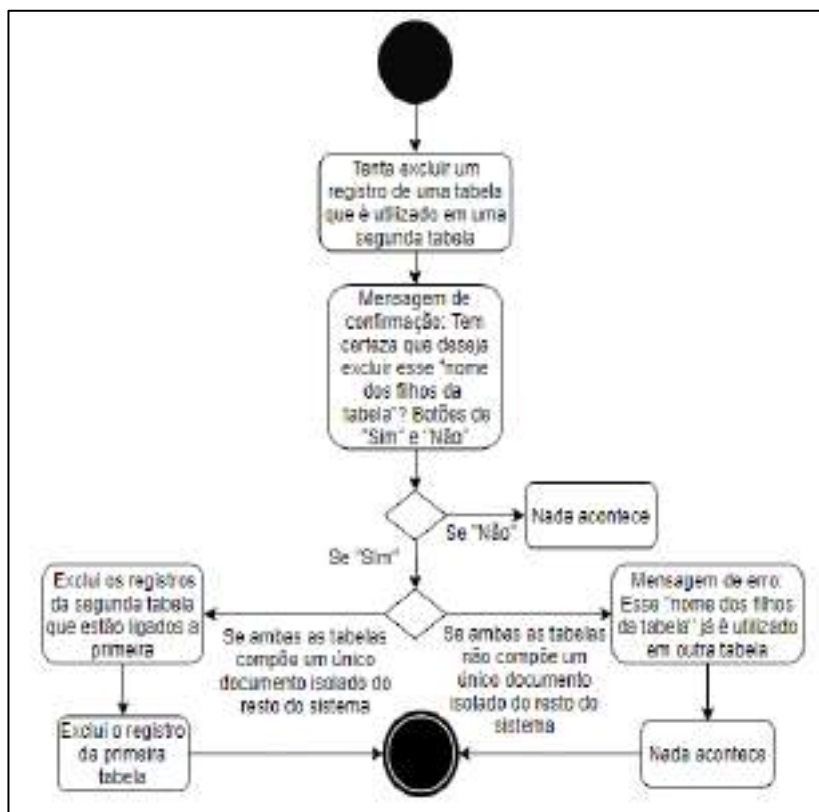


Fonte: Produzida pelo autor

### 3.5.6 Diagrama de Exclusão de Registro

Segue abaixo um diagrama de atividades para explicar o padrão e a exceção:  
Conforme a figura abaixo.

Figura 17: Diagrama de Atividades para exclusão de registros



Fonte: Produzida pelo autor

### 3.6 Atalhos do Teclado:

- ✓ Ao pressionar “alt”+”n”, se a tela estiver em modo de navegação, o botão “novo” será acionado;
- ✓ Ao pressionar “alt”+”m”, se a tela estiver em modo de navegação, o botão “modificar” será acionado;
- ✓ Ao pressionar “alt”+”e”, se a tela estiver em modo de navegação, o botão “excluir” será acionado;
- ✓ Ao pressionar “alt”+”p”, se a tela estiver em modo de navegação, o botão “primeiro” será acionado;
- ✓ Ao pressionar “alt”+”a”, se a tela estiver em modo de navegação, o botão “anterior” será acionado;
- ✓ Ao pressionar “alt”+”r”, se a tela estiver em modo de navegação, o botão “próximo” será acionado;
- ✓ Ao pressionar “alt”+”f”, se a tela estiver em modo de navegação, o botão “fechar” será acionado;
- ✓ Ao pressionar “alt”+”g”, se a tela estiver em modo de inserção ou edição, o botão “gravar” será acionado;
- ✓ Ao pressionar “alt”+”c”, se a tela estiver em modo de inserção ou edição, o botão “cancelar” será acionado;
- ✓ Caso a tela esteja em modo de inserção ou modificação, ao pressionar “Esc” o botão cancelar é acionado, caso contrário o botão fechar é acionado;

### 3.7 Ao abrir uma tela ligada a uma tabela do banco:

- ✓ A tela é aberta em modo de navegação;
- ✓ A tela mostra último registro cadastrado;

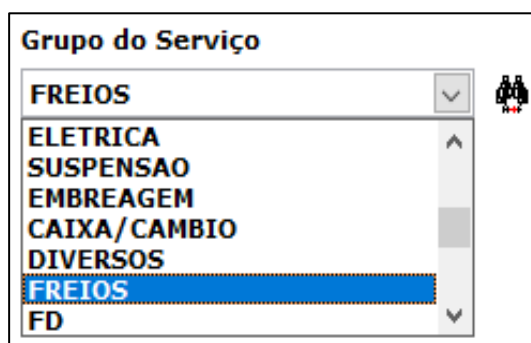
### 3.8 Ligações com outras tabelas dentro de uma tela:

Os campos que fazem a ligação são normalmente um campo com dropdown, ao clicar nele aparecem, em baixo dos campos todos os registros todos os registros ativos de um campo da outra tabela;

Os campos usados para armazenar a chave estrangeira de outra tabela nunca mostram a id do registro selecionado da outra tabela, e sim algum campo de texto, preferencialmente único, da outra tabela;

Os campos que fazem ligação com outra tabela tem sempre um botão ao lado. Conforme a figura abaixo.

Figura 18: Dropdown de um campo que faz ligação com uma tabela



Fonte: Produzida pelo autor

Se a outra tabela possui tela de consulta, o botão vai levar para a sua tela de consulta para possibilitar uma filtragem para a seleção do registro, se não vai levar direto para a tela de CRUD da outra tabela;

### 3.9 Tabelas Planejadas, de acordo com avaliações prévias, para ter muitos registros:

- ✓ Todas possuem tela de consulta.

### 3.10 Tabelas Planejadas, de acordo com avaliações prévias, para ter muitos registros:

- ✓ Nenhuma vai ter tela de consulta;



### 3.11 Telas de Consulta

Todas as telas de consulta tem um grid. O grid mostra pelo menos todos os registros e todos os campos (mudando as vezes o título do campo do banco para torná-lo mais compreensível e com inicial maiúscula), exceto id, da tabela a qual a tela está ligada, mostrando alguns casos dados de outras tabelas também;

Acima do grid tem sempre um argumento de pesquisa que vai filtrando os registros mostrados no grid a medida que o usuário escreve nele.

Essas telas são chamadas nas telas que utilizam à tabela a qual a tela de consulta está ligada, através de um botão a direita do campo que armazena a chave estrangeira ou através de um “enter” em no campo código de um grid;

Quando o usuário pressiona “enter” sobre um registro do grid da tela de consulta, ela é fechada e o campo que armazena sua chave estrangeira é preenchido com o id do registro selecionado;

As telas de consulta tem um componente dropdown para selecionar o campo que vai ser filtrado e uma barra de pesquisa para que vai filtrando os registros tendo o campo selecionado como parâmetro;

As telas de consulta tem um botão a direita da barra de pesquisa que leva para a sua respectiva tela de cadastro. Conforme a figura abaixo.

Figura 19: Botão a direita do campo levando para tela de consulta

Placa

Argumento para pesquisa: PLACA

Pesquisa: TTT

Todos [Acessar Cadastro](#)

Placa	Modelo	Ano	Cor	Chassi	Combustível	Potência Fabricante	Nome
TTT2000	F8	2010/2010	PRETO		ALCOOL	FORD	AGLAZ T DE ARAUJO ME

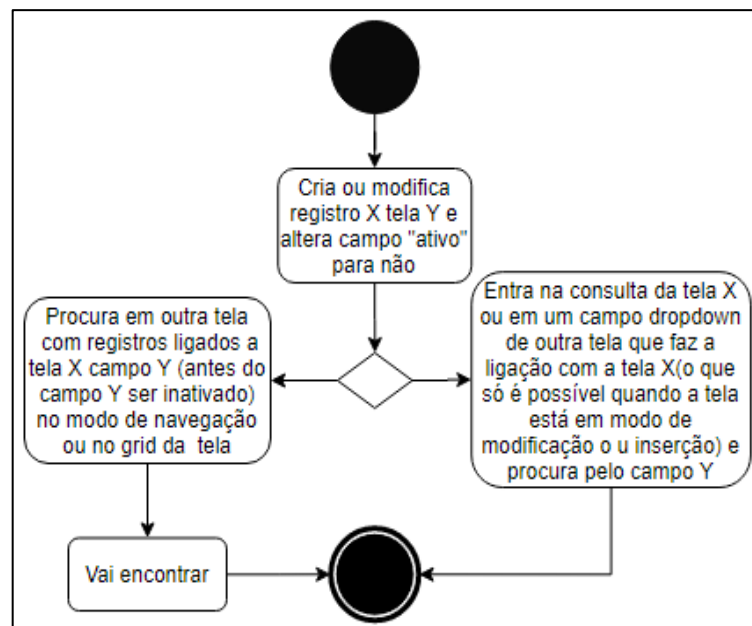
Fonte: Produzida pelo autor

### 3.12 Ativo / Inativo

Todos as tabelas cujo os registros tem ou vão ter suas chaves primarias utilizadas em outras tabelas (como chave estrangeira), vão ter o campo ativo. O campo ativo identifica, ao inserir ou modificar um registro na pai e alterar o campo com a chave estrangeira que faz a ligação com a tabela filho, se o registro aparecerá ou não como opção. Registros inativos não serão disponíveis nas inclusões ou modificações de registros da tela pai, mas, ao navegar na tela da tabela pai entre os registros ou olhar no grid da mesma, vai aparecer todos os dados que a tela da tabela pai puxa da tabela filho, mesmo quando os registros em questão estiverem inativos.

Segue abaixo um diagrama de atividades para exemplificar como funciona a atividade/inatividade dos registros:

Figura 20: Diagrama de Inativação



Fonte: Produzida pelo autor

### **3.13 Padrões de Código usados no Módulo de OS**

#### **3.13.1 Prefixos para os componentes principais usados no Sistema**

- ✓ Telas: frm;
- ✓ Botões: btn;
- ✓ Componentes de texto: edt;
- ✓ Componentes de texto ligados ao banco: dbedt;
- ✓ Grid ligado ao banco(dbGrid):dbgrd;
- ✓ dbLookupCombobox: dblkcb;
- ✓ Combobox: cbb;
- ✓ dbCombobox: dbcbb;

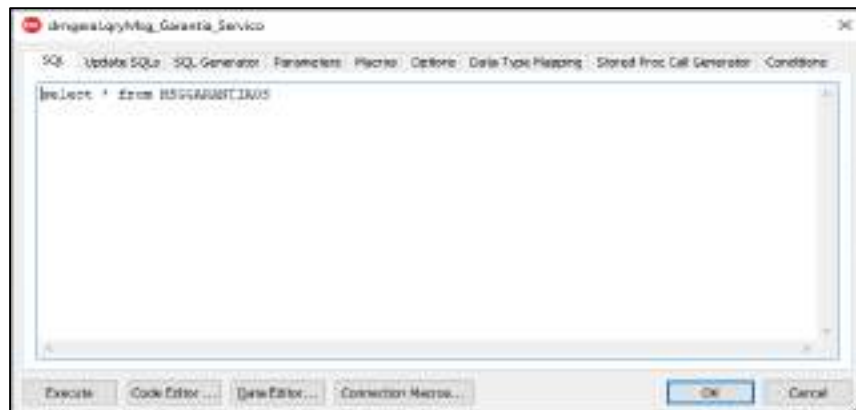
#### **3.13.2 Funcionalidades dos Componentes Principais Usados no Sistema**

- ✓ Botões: Realizar alguma atividade no evento do click;
- ✓ Componentes de texto: Mostrar alguma informação ao usuário;
- ✓ Componentes de texto ligados ao banco: Permitir o CRUD da tela;
- ✓ dbGrid: Mostrar informações de uma query(ligada diretamente ao banco) ou de um clientDataSet(ligado a um dataSetProvider que é ligado a uma query);
- ✓ dbLookupCombobox: Fazer conexão com uma tabela do banco que não é a tabela principal da Tela, ele permite gravar o valor da chave de um registro de qualquer tabela enquanto mostra pro usuário o valor de outro campo do mesmo registro;
- ✓ Combobox: É um campo com dropdown que permite selecionar um item de sua lista;
- ✓ dbCombobox: É um campo com dropdown que permite selecionar um item de sua lista ligado diretamente query(ligada ao banco diretamente ao banco) ou em um clientDataSet(ligado a um dataSetProvider que é ligado a uma query);

### 3.13.3 Conexão com o banco

- ✓ Tabelas que permitem alteração direta dos registros:
- ✓ Para essas telas é utilizado o componente TuniQuery para a conexão com o banco através de um SQL. Conforme a figura abaixo.

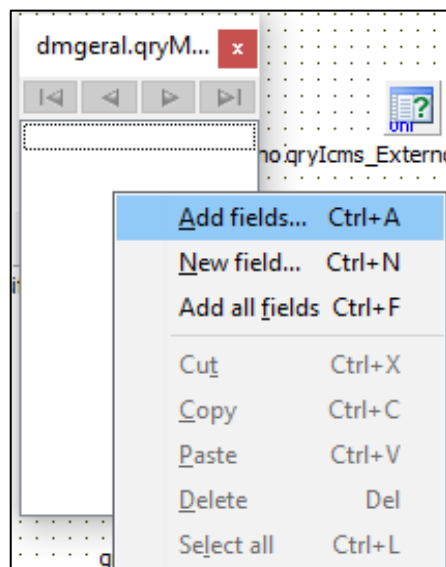
Figura 21: Consulta dentro de uma query



Fonte: Produzida pelo autor

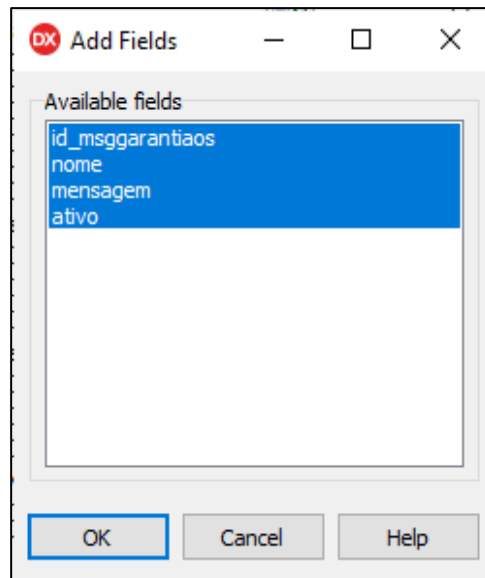
- ✓ Ao executar a consulta é possível inserir campos na query que virão a interagir com o banco através dos componentes da tela. Conforme a figura abaixo.

Figura 22: Query sem campos adicionados



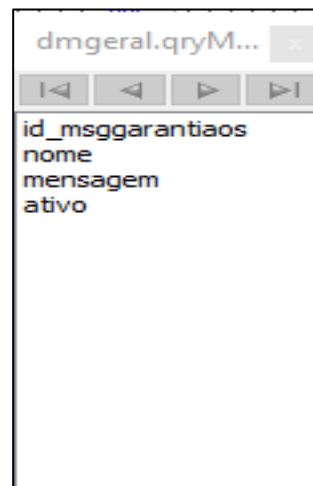
Fonte: Produzida pelo autor

Figura 23: Campos disponíveis para adicionar



Fonte: Produzida pelo autor

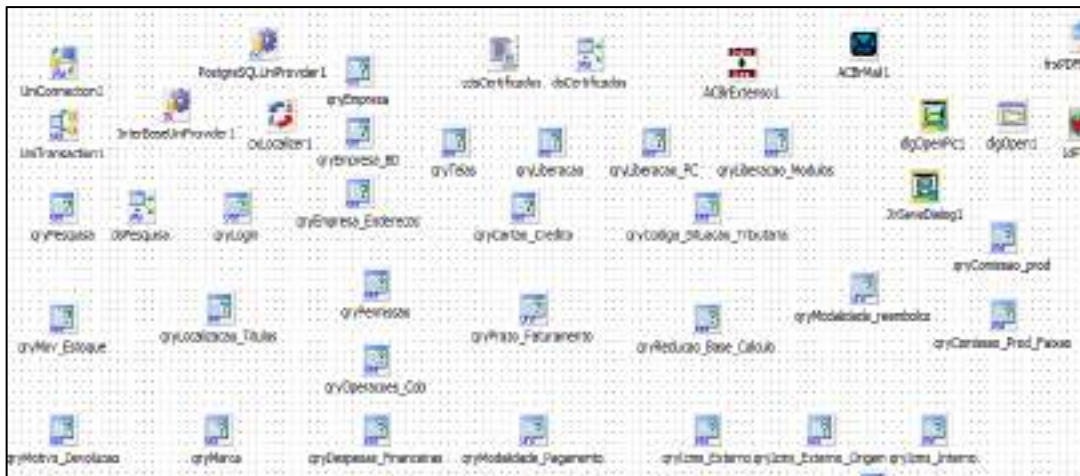
Figura 24: Query com os Campos Adicionados



Fonte: Produzida pelo autor

As queries (filhas do componente TuniQuery) são agrupadas na tela "dmGeral", lá estão armazenadas todas as queries que fazem conexão com o banco. Esse padrão é utilizado para permitir que qualquer tela acesse as informações de outra tela de forma simples e organizada, Conforme a figura abaixo.

Figura 25: dmGeral



Fonte: Produzida pelo autor

Para fazer uma tela utilizar informações dessa ou qualquer outra tela do sistema, ela tem que “chamar” ela dentro de “uses”, acrescentando o prefixo “u” e retirando o prefixo “frm” quando necessário por padrão do sistema; Conforme a figura abaixo.

Figura 26: Chamando recursos do dmGeral para uma Tela

```
implementation
uses uDMGeral, uDMExtra;
```

Fonte: Produzida pelo autor

No evento “onShow” da tela é chamada e populada a query que vai fazer a conexão com o banco; Conforme a figura abaixo.

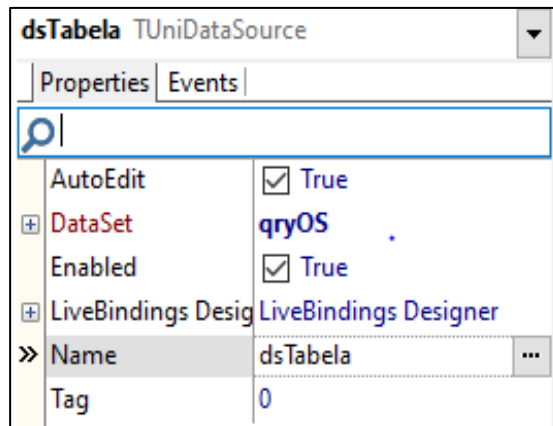
Figura 27: Código para populat uma query

```
DMGeral.qryMsg_Garantia_Servico.Close;
DMGeral.qryMsg_Garantia_Servico.SQL.Clear;
DMGeral.qryMsg_Garantia_Servico.SQL.Add('select * from MSGGARANTIAOS');
DMGeral.qryMsg_Garantia_Servico.Open;
DMGeral.qryMsg_Garantia_Servico.IndexFieldNames := 'nome';
```

Fonte: Produzida pelo autor

A query que conecta com a tabela principal do banco é sempre ligada ao componente “TuniDataSource” para permitir a conexão com o grid da tabela e, por padrão, o nome dado a esse datasource é dsTabela; Conforme a figura abaixo.

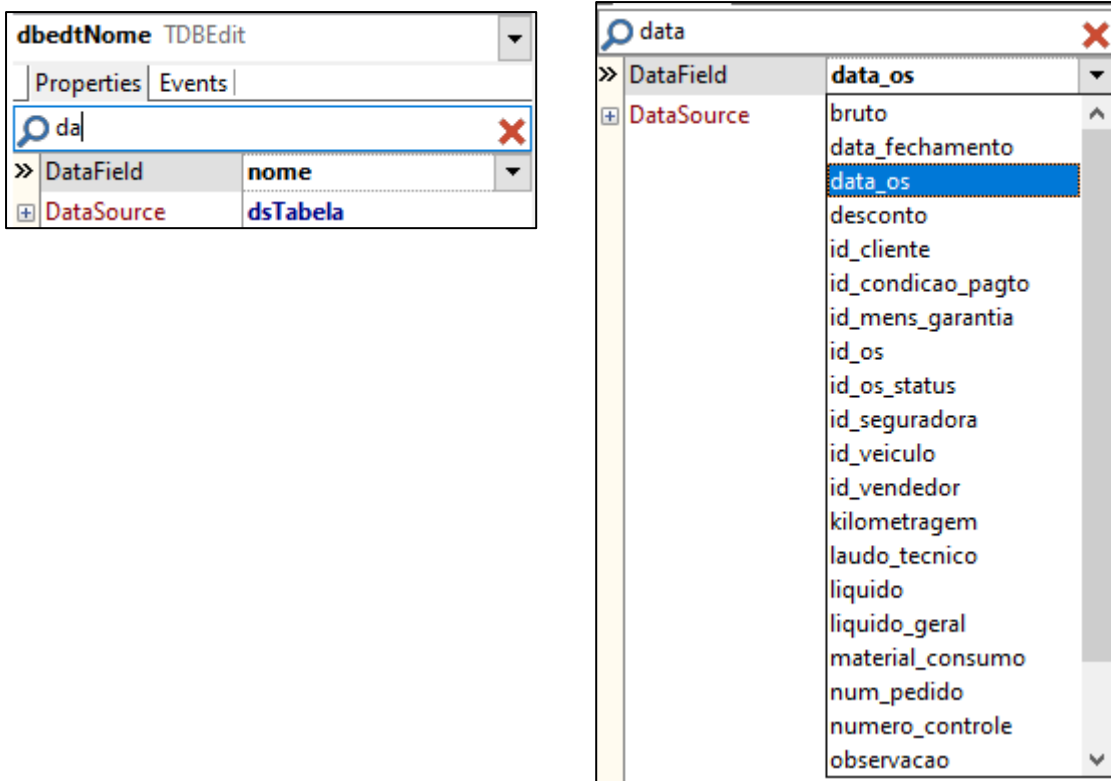
Figura 28: Conf datasource para conectar a uma query



Fonte: Produzida pelo autor

Os componentes de alteração, visualização e inclusão agora podem ser conectados campos do banco bastam conectá-los ao datasource e escolher um campo da query; Conforme a figura abaixo.

Figura 29: Conectando componente a um campo da query

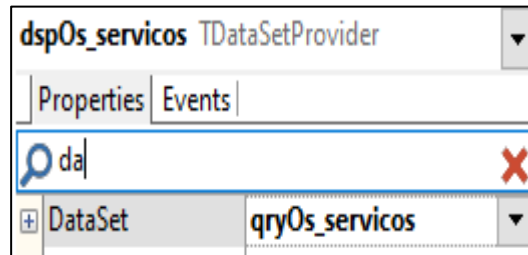


Fonte: Produzida pelo autor

Componentes ligados a tabelas do banco em tempo de execução;

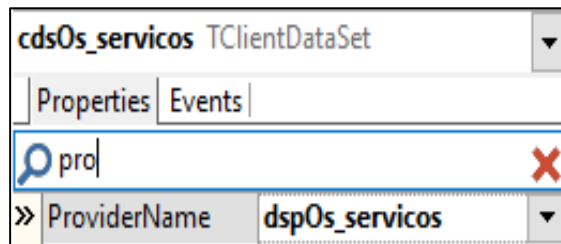
Para fazer o CRUD de registros apenas em tempo de execução, é utilizado componente TDataSetProvider, que pega as informações da query e transfere elas para um TClientDataSet; Conforme as figuras abaixo exemplificam.

Figura 30: Conectando um TDataSetProvider a uma query



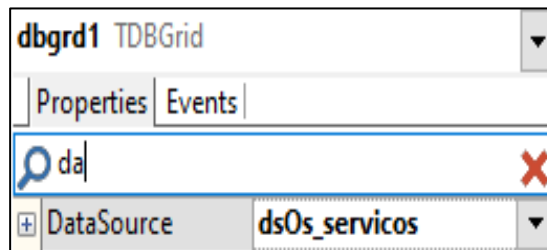
Fonte: Produzida pelo autor

Figura 31: Conectando um TClientDataSet a um TDataSetProvider



Fonte: Produzida pelo autor

Figura 32: Conectando um dbGrid ao ClienteDataSet



Fonte: Produzida pelo autor

Para gravar alterações feitas em um clientDataSet se usa comando `Nome_do_clientDataSet.ApplyUpdates(-1)`, salvando assim todas as operações realizadas no clientDataSet;



Para cancelar alterações feitas em um clientDataSet se usa o comando Nome\_do\_clientDataSet.CancelUpdates;

### 3.13.4 Habilitar ou Desabilitar Botões e Campos:

Para habilitar ou desabilitar campos, foi criada a função a função habilita, que vai ser executada sempre que a tabela sair do modo de edição, inserção ou navegação;

Essa receber “true” ou “false” como parâmetros, depende da tela estar no modo de navegação ou não; Conforme a figura abaixo.

Figura 33: Função habilitada da tela OS

```

procedure TfirmOS.habilita(valor :Boolean);
Var
i : Integer;
begin
  for i := 0 to ComponentCount -1 do
  begin
    if (Components[i] is TDBEdit) then
      TDBEdit(Components[i]).Enabled := valor;
    if (Components[i] is IDBCheckBox) then
      IDBCheckBox(Components[i]).Enabled := valor;
    if (Components[i] is IDBComboBox) then
      IDBComboBox(Components[i]).Enabled := valor;
    if (Components[i] is TDBRadioGroup) then
      TDBRadioGroup(Components[i]).Enabled := valor;
    if (Components[i] is TDBMemo) then
      TDBMemo(Components[i]).Enabled := valor;
    if (Components[i] is TDBLookupComboBox) then
      TDBLookupComboBox(Components[i]).Enabled := valor;
    if (Components[i] is IDBGrid) then
      IDBGrid(Components[i]).Enabled := valor;
    if (Components[i] is TjvDBDateEdit) then
      TjvDBDateEdit(Components[i]).Enabled := valor;
  end;
  dbedtid_os.Enabled:=false;
end;

```

Fonte: Produzida pelo autor

Ela percorre todos os componentes da tela e define que tipo de componente recebe o valor “true” ou “false” para o seu estado de ativação, dependo do tipo dele, os campos que sempre vão ter sempre um único valor são colocados abaixo do laço com um valor fixo.

Para habilitar ou desabilitar botões, foi criada a função a função `habilita_navegacao`, que vai ser executada sempre que a tabela sair do modo de edição, inserção ou navegação;

Essa função vai verificar as permissões do usuário, e permitir ou não os clicks nos botões dependem das permissões e do parâmetro que a função receber (“true” ou “false”);

Função `habilita_navegacao` da tela OS (pode mudar um pouco dependendo da tela), Verificando permissão, Conforme a figura abaixo.

Figura 34: Verificando permissão

```
DMGeral.qryPermissao.locate('programa','B0100',[]);
```

Fonte: Produzida pelo autor

Figura 35: Campos que podem ser alterados dependendo do valor do parâmetro da função

```
btnCancela.Enabled := not(valor);  
btnSalvar.Enabled := not(valor);  
btnexclui_item.Enabled := not(valor);  
btnFecha.Enabled := valor;  
btnRecupera.Enabled := valor;  
dbgrdl.Enabled := not(valor);
```

Fonte: Produzida pelo autor

Figura 36: Campos que nunca podem ser alterados

```
dbedtbruto.enabled:=false;  
dbedtLiquido.enabled:=false;  
dbedtData_fechamento.enabled:=false;  
dbedtSituacao.enabled:=false;
```

Fonte: Produzida pelo autor

### 3.13.5 Colocando a Tela em modo de Inserção (ao clicar em novo):

- ✓ Primeiro é validado se o usuário tem permissão para incluir registros; Conforme a figura abaixo.

Figura 37: Verificando permissão para criar uma nova OS

```

procedure TfrmOS.btnNovoClick(Sender: TObject);
begin
  DMGeral.qryPermissao.locate('programa','A0100',[]);
  if DMGeral.qryPermissao.FieldByName('inclui').Value <> 'SIM' then
  begin
    ShowMessage('Acesso negado para seu usuário!');
    exit;
  end;

```

Fonte: Produzida pelo autor

- ✓ Depois são acionadas as funções habilita e habilita\_navegacao. Conforme a figura abaixo.

Figura 38: Configurando os campos e botões para o modo de inserção

```

habilita(true);
habilita_navegacao(false);

```

Fonte: Produzida pelo autor

- ✓ Depois é utilizado o código dsTabela.append, para criar um registro vazio para inserir os dados. Conforme a figura abaixo:

Figura 39: dsTabela append

```

dstabela.DataSet.Append;

```

Fonte: Produzida pelo autor

- ✓ Depois é colocado o foco na primeira aba da tela e no primeiro campo ativo da primeira aba da tela. Conforme a figura abaixo :

Figura 40: Colocando foco no primeiro campo da tela

```
ts5.Show;  
dblkcbbid_cliente.SetFocus;
```

Fonte: Produzida pelo autor

- ✓ Depois são inseridos os valores default. Conforme a figura abaixo :

Figura 41: Valores default

```
qryOS.FieldByName('previsao_entrega').AsDateTime:=now;  
qryOS.FieldByName('data_os').AsDateTime:=now;  
qryOS.FieldByName('id_os_status').AsInteger:=16;  
qryOS.FieldByName('bruto').AsFloat:=0;  
qryOS.FieldByName('desconto').AsFloat:=0;  
qryOS.FieldByName('liquido').AsFloat:=0;  
qryOS.FieldByName('liquido_geral').AsFloat:=0;
```

Fonte: Produzida pelo autor

### 3.13.6 Colocando a Tela em modo de Inserção (ao clicar em novo) 1:

- ✓ Primeiro é validado se o usuário tem permissão para modificar registros;
- ✓ Depois são acionadas as funções habilita e habilita\_navegacao;
- ✓ Depois é utilizado o código dsTabela.edit, para colocar o registro no modo de edição;
- ✓ Depois é colocado o foco na primeira aba da tela;  
Conforme a figura abaixo:

Figura 42: Códigos que realizam e exemplificam as ações explicadas acima

```

procedure TfrmOS.btnModificarClick(Sender: TObject);
begin
  DMGeral.qryPermissao.locate('programa','A0100',[]);
  if DMGeral.qryPermissao.FieldByName('altera').Value <> 'SIM' then
  begin
    ShowMessage('Acesso negado para seu usuário!');
    exit; ↴
  end;

  habilita(true);
  habilita_navegacao(false);
  atualizagrid();

  cdsOs_Servicos.Edit;
  dsTabela.Edit;
  ts5.Show;
  dbedtData_fechamento.Enabled:=False;
  dblkcbbid_cliente.SetFocus;

```

Fonte: Produzida pelo autor

### 3.13.7 Quando a tela entra no modo de navegação

- ✓ Podem existir várias validações no salvar então vou mostrar o padrão apenas no cancelar.
- ✓ Ao clicar em cancelar:
- ✓ É requisitado confirmação, se sim as funções habilita e habilita\_navegacao são chamadas e as alterações são canceladas; Conforme a figura abaixo:

Figura 43: Códigos que realizam e exemplificam as ações explicadas acima 1

```

r := Application.MessageBox('Deseja realmente cancelar?', 'Alerta', MB_yesno+MB_iconInformation);
if r = 6 then
begin
  habilita(false);
  habilita_navegacao(true);
  dsTabela.DataSet.Cancel;

```

Fonte: Produzida pelo autor

### 3.13.8 Botões de Navegação

- ✓ Primeiro: dsTabela.DataSet.First;
- ✓ Anterior: dsTabela.DataSet.Prior;
- ✓ Próximo: dsTabela.DataSet.Next;
- ✓ Último: dsTabela.DataSet.Last;

### 3.13.9 Botão Fechar

- ✓ Quando clica no botão fechar o evento onClose é acionado;
- ✓ No evento onClose é verificado se a tela está no modo de inserção, em caso positivo a tela fecha, em caso negativo a tela não fecha e aparece a mensagem “Salve ou cancele a alteração antes de fechar a tela”;  
Conforme a figura abaixo:

Figura 44: Código do botão Fechar onClose e código do evento onClose da tela

Close;

Fonte: Produzida pelo autor

```
if qryOS.State in [dsedit,dsinsert] then
begin
  ShowMessage('Salve ou cancele a alteração antes de fechar a tela');
  abort;↑
end
else
frmOS:=nil;
```

### 3.14 Tela Principal do Sistema

Figura 45: Tela Principal

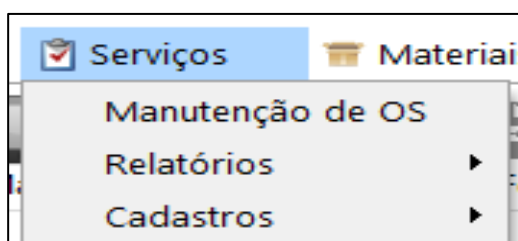


Fonte: Produzida pelo autor

Todas as telas do sistema, exceto as de consulta, podem ser acessadas através da tela Principal. As telas do módulo de ordens de serviço ficam dentro de serviços.

Conforme a figura abaixo:

Figura 46: Acesso ao módulo de ordens de serviço pela tela principal



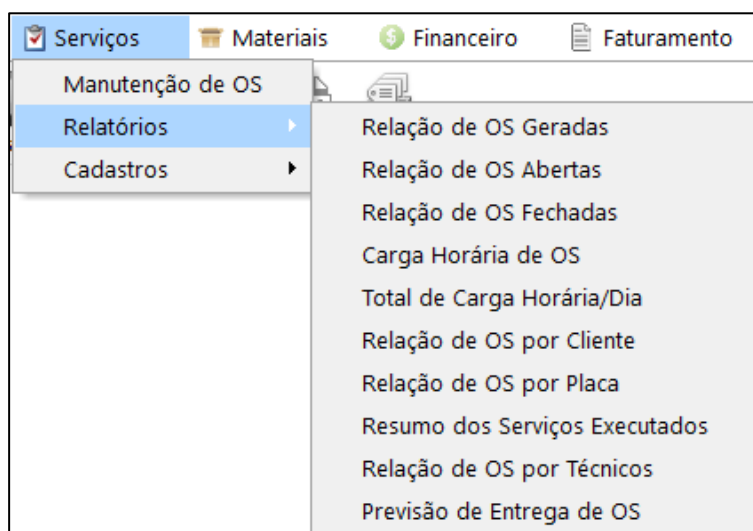
Fonte: Produzida pelo autor

O link "Manutenção de OS" leva para a tela principal do módulo de serviços, que é a tela Manutenção de OS.

Ao deixar o mouse sobre Relatórios aparecem os links para todos os relatórios do módulo de serviços.

Conforme a figura abaixo:

Figura 47: Acesso aos relatórios do módulo de serviços

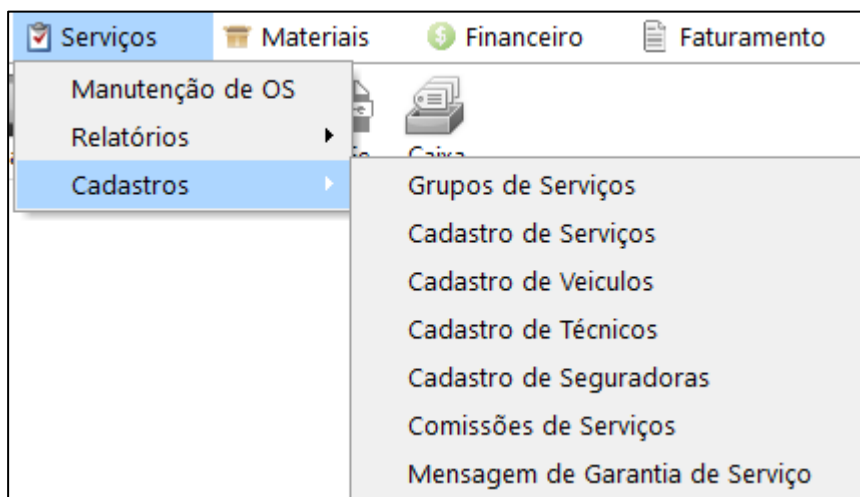


Fonte: Produzida pelo autor

Ao deixar o mouse sobre Cadastros aparecem os links para todas as demais telas de leitura, modificação, inserção e exclusão de dados do módulo de serviços.

Conforme a figura abaixo:

Figura 48: Telas do módulo de serviços



Fonte: Produzida pelo autor



### 3.15 Tela Cadastro de Serviços

Figura 49: Tela cadastro de serviços

Fonte: Produzida pelo autor

### 3.16 Características e funcionalidades básicas da tela Grupos de Serviços

Ela segue todos os padrões de layout, regras e desenvolvimento comentados nos tópicos Padrões do Sistema e Padrões de Desenvolvimento;

Ela utiliza um “dbLookupComboBox” para indicar a qual grupo de serviço vai pertencer o serviço que está sendo cadastrado ou modificado;

Ela possui ligação direta com a tabela do banco “serviços”, permitindo alteração leitura, inclusão, alteração e exclusão dos registros das mesma e indireta com a tabela “grupos\_serviços”, permitindo apenas selecionar um registro da mesma;

Motivos do desenvolvimento da tela Cadastro de Serviços:

Em qualquer oficina prestadora de serviço organizada, existe pelo menos uma tabela com descrição e preços para cada serviço.

A descrição de um serviço pode ser bastante específica, podendo chegar próximo dos 50 caracteres, de forma que, caso os itens de serviço sejam criados do zero para cada OS, se torna cada vez mais difícil manter qualquer tipo de padrão e controle sobre eles.

Portanto é natural que o módulo de OS que está sendo desenvolvido tenha, no mínimo, espaço para registrar modelos de serviços que possam ser usados na hora de adicionar um item de serviço em uma OS, de forma que o usuário não crie vários itens de serviço com nomes diferentes para a mesma coisa, não tenha que perder tempo procurando o valor do serviço e mantenha o banco de dados do sistema organizado.

Com isso em mente, e aproveitando o fato desse modelo (o serviço) permitir que um item de serviço permaneça ligado ao modelo que o criou, de forma que podem ser acrescentadas informações indiretamente aos itens de serviço, sem poluir o grid de itens de serviço, se iniciou uma análise de outros sistemas para definir quais informações poderiam ser adicionadas ao cadastro de serviço. Essas informações deveriam ser úteis em futuros relatórios para melhorar a gestão da OS ou facilitar a filtragem dos registros na hora de consultar serviços e, ao mesmo tempo, não tornar o processo de cadastro muito burocrático com um excesso de informações. Após pesquisas e avaliações foram definidos os seguintes campos: “descricao”, “preco”, “horas”, “minutos”, “custo”, “tipo”, “data\_cadastro”, “grupo de serviço” e tipo.

### **3.17 Características específicas da tela**

- ✓ A comissão é personalizada a cada cliente, não será detalhado;
- ✓ Os serviços cadastrados nessa tela servem como template para o cadastro de itens de serviços em uma ordem de serviços;
- ✓ O serviço ao qual um item de serviços está relacionado acrescenta as seguintes informações adicionais ao item de serviço:
  - ✓ Horas estimadas para conclusão do item de serviço;
  - ✓ Minutos estimados para conclusão do item de serviço;
  - ✓ Grupo ao qual o item de serviço pertence;

### 3.18 Itens de Serviço da OS

Figura 50: Grid de serviços da OS

It	Código	Descrição	Lado	Qtd	Pr Unitária	Box Técnico	Último Evento	Botão	Status
1	3	PINTURA PORTA DIANTEIRA ESQUERDA GRD		999	200,00	ALEXANDRE SILVA	FINALIZADO	Eventos	APROVADO
2	5	TROCA DE PNEU		1	50,00			Eventos	APROVADO

Condição de Pagamento	Val. Serviços	Desconto(%)	Val Líquido
30 DIAS DEPOIS A PRAZO	09.850,00	20,00	159.990,00

Imprimir	Excluir	Excluir Item	Gravar	Cancelar	Cancelar	Cancelar	Cancelar	Cancelar	Cancelar	Cancelar
----------	---------	--------------	--------	----------	----------	----------	----------	----------	----------	----------

Fonte: Produzida pelo autor

### 3.19 Características e funcionalidades básicas do grid de Itens de Serviços

Não existe tela CRUD de itens de serviço;

Os itens de serviço vão ser visualizados, modificados, inseridos e excluídos apenas dentro da OS através do grid da tela OS;

No grid de cada OS, só é possível ver, criar, modificar ou excluir os itens de serviço ligados aquela OS;

É possível escolher um serviço, através da tela Consulta de Serviços, para servir de modelo para a criação de um item de serviço;

Como podem ter vários usuários trabalhando e fazendo cadastros em uma mesma tabela ao mesmo tempo, foi utilizado um componente Tclientdataset (já foi comentado no tópico padrões do desenvolvimento), para possibilitar a alteração e criação de registros apenas em tempo de execução, de forma que, se dois usuários estão cadastrando uma OS ao mesmo tempo, o banco de dados não é corrompido;

O grid da OS possui ligação com a tabela “os\_servicos”, de forma que é possível visualizar, cadastrar, modificar e excluir itens de serviço através dele, só que as alterações feitas só serão salvas no banco quando a OS for salva.

Para incluir um novo item de serviço, a tela de OS tem que estar em modo de inserção ou modificação e a tela Consulta de Serviços (ligada a tabela do banco “servicos”), tem que ser chamada;

Para chamar a tela Consulta de Serviços, basta a tela de OS estar em modo de inserção ou alteração e o usuário pressionar “enter” abaixo do campo “código” no grid de itens de serviço;

Conforme a figura abaixo:

Figura 51: Grid de itens de serviços com foco no campo código

It	Código	Descrição	Lado	Qty	Pr Unitário	Box Técnico	Último Evento	Botão	Status
1	3	PINTURA PORTA COXETERA ESQUERDA GPD		2	200,00		FINALIZADO	Eventos	APROVADO
2	5	TROCA DE PNEU		1	50,00			Eventos	APROVADO

Fonte: Produzida pelo autor

- Pressiona “enter” acionar o evento onKeyPress do grid
- Trecho de código verifica se a tela está em modo de edição ou inserção, se o cursor está sobre o campo código e se a tecla “enter” foi pressionada.

Conforme a figura abaixo:

Figura 52: Código que chama a tela Consulta de Serviços

```

if qry09.State in [dsInsert,dsEdit] then
begin
  if DBGrid1.Columns.Grid.SelectedField.FieldName = 'id_servico' then
  begin
    if Key = 13 then
    begin
      if frmconsulta_servicos = nil then
        Application.CreateForm(Tfrmconsulta_servicos, frmconsulta_servicos);
        frmconsulta_servicos.showmodal;
      end;
    end;
  end;
end

```

Fonte: Produzida pelo autor, a figura acima apresenta um trecho de código que verifica se a tela está em modo de edição ou inserção, se o cursor está sobre o campo código e se a tecla “enter” foi pressionada

- Cria e abre a tela Consulta de Serviço, que permite filtrar serviços pelos campos mostrados apresentados na figura abaixo.

Figura 53: Tela Consulta de Serviços com dropdown das possibilidades de filtros aberto

Preço	Horas	Minutos	Tipo	Grupo
30,00	4	40	PROPRIO	PREÇOS
50,00	1	40	PROPRIO	SUSPENSÃO

Fonte: Produzida pelo autor

Com a tela de Consulta de Serviços aberta, para escolher o serviço que vai servir de template, o usuário pode usar os filtros (ou não), tem que dar um “enter” sobre o serviço escolhido para servir de template para o item. O usuário pode também, a partir da tela Consulta de Serviços, acessar a tela de Cadastro de Serviços, cadastrar um novo serviço e, finalmente, escolher o serviço recém-cadastrado para servir de template para o item de serviço;

Depois de confirmar a operação, o item vai receber, do serviço escolhido como template, seus campos “código”, “descrição” e “preço”, podendo alterar “descrição” e “preço”. O campo “It”(item) recebe o valor do item anterior+1, a quantidade recebe 1 e status recebe “APROVADO”;

O item de serviço fica permanentemente relacionado ao serviço que serviu de template pelo campo “codigo”, de forma que as informações do serviço escolhido como template podem ser acessados em seus respectivos itens em diversos relatórios;

Conforme as figuras abaixo demonstram:

Figura 54: Serviço selecionado da tela de consulta de serviços

LIMPEZA	30,00	0	45	PROPRIO	EMBREAGEM
▶ PINTURA PORTA DIANTEIRA ESQUERDA	200,00	2	0	PROPRIO	ELETRICA
RE	3.232,00	0	0	PROPRIO	EMBREAGEM

Fonte: Produzida pelo autor, essa figura demonstra as condições necessárias para que a tecla “enter” abra a consulta de serviços.

Seleciona um registro.

Pressiona “enter” com o registro selecionado para invocar o evento `onKeyPress` do grid.

Código verifica se a tela está em modo de edição ou inserção, se o cursor está sobre o campo código, se a tecla “enter” foi pressionada e se já existia um item de serviço ligado a essa OS.

Figura 55: Código para geração de novo item de serviço

```

procedure frmConsulta_Servicos.Dbgrid.KeyPress(Sender: TObject; var Key: Char);
var item: Integer;
begin
  if Key = #13 then
  begin
    if frmOS.Showing=True then
    begin
      Key := #0;

      frmOS.cdsOs_Servicos.Open;
      frmOS.cdsOs_Servicos.Last;

      if frmOS.cdsOs_Servicos.FieldByName('item_os').Value=null then
        item:=0
      else
        item:=frmOS.cdsOs_Servicos.FieldByName('item_os').Value;
      frmOS.cdsOs_Servicos.Append;
      frmOS.cdsOs_Servicos.FieldByName('item_os').Value := item+1;
      frmOS.cdsOs_Servicos.FieldByName('id_servico').Value := qryconsulta_servicos.FieldByName('codigo').Value;
      frmOS.cdsOs_Servicos.FieldByName('descricao').Value := qryconsulta_servicos.FieldByName('descricao').Value;
      frmOS.cdsOs_Servicos.FieldByName('preco').Value := qryconsulta_servicos.FieldByName('preco').Value;
      frmOS.cdsOs_Servicos.FieldByName('quantidade').Value := 1;
      frmOS.cdsOs_Servicos.FieldByName('status').Value := 'APROVADO';
      frmOS.cdsOs_Servicos.Post;

      frmOS.Dbgrid.SelectedIndex := frmOS.Dbgrid.SelectedIndex+1;
    end;
  end;
end;

```

Fonte: Produzida pelo autor, o código acima verifica se a tela está em modo de edição ou inserção, se o cursor está sobre o campo código, se a tecla “enter” foi pressionada e se já existia um item de serviço ligado a essa OS. Se sim para as três condições, fecha a tela de consulta, cria um serviço novo ligado a OS com campos default e manda o cursor para o campo descrição

Se sim para as três primeiras condições, fecha a tela de consulta, cria um serviço novo ligado a OS com campos default e manda o cursor para o campo descrição.

Recorte do grid de itens de serviço da tela OS depois de ter acabado de incluir um novo item .

Conforme a figura abaixo:

Figura 56: Recorte do grid de itens de serviço com um serviço recém adicionado

It	Código	Descrição	Lado	Qtd	Pr Unitário	Box Técnico	Último Evento	Bo/SO	Status
1	3	PINTURA PORTA DIANTEIRA ESQUERDA GPD		2	206,00		FINALIZADO	Eventos	APROVADO
2	3	PINTURA PORTA DIANTEIRA ESQUERDA		1	206,00			Eventos	APROVADO

Fonte: Produzida pelo autor

Campos dos itens:

- ✓ O campo “lado” tem 4 caracteres, pode ser alterado;

- ✓ O campo “quantidade” recebe 1 quando é criado, podendo ser alterado, ele permite apenas valores inteiros de 1 a 99999;
- ✓ O campo “box” não recebe nada quando criado, podendo ser alterado para valores inteiros entre 1 até 99;
- ✓ O campo “Técnico” será comentado no tópico “Informando um Técnico em um Item de Serviços”;
- ✓ Os campos “Último Evento” e campo “Botão” serão comentados no tópico “Cadastro de Eventos”;
- ✓ O campo “status” recebe “APROVADO” quando criado, mas pode ser alterado ao clicar sobre ele, abrindo assim um dropdown com as opções “APROVADO” e “REJEITADO”. Itens com status “REJEITADO” não são contabilizados na soma dos itens da OS e não podem mais ser alterados, apenas excluídos;

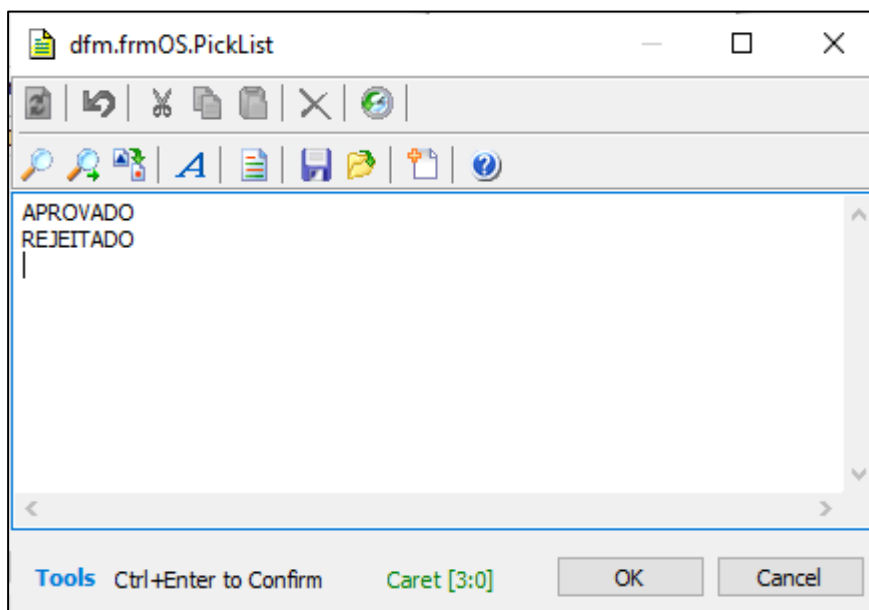
Conforme a figura abaixo:

Figura 57: Propriedade PickList do campo status do grid de itens de serviço



Fonte: Produzida pelo autor

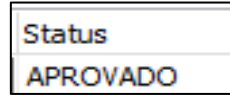
Figura 58: Adicionando os status possíveis no PickList



Fonte: Produzida pelo autor



Figura 59: Campo status já configurado recebendo “APROVADO”



Fonte: Produzida pelo autor, o campo acima mostra que já é possível inserir um status com a configuração realizada, mas são necessários 5 clicks.

- ✓ Como mostra a imagem acima, já é possível selecionar um status, mas são necessários 5 clicks.
- ✓ Para resolver esse problema é utilizado o evento onCellClick do grid.
- ✓ Ao clicar em um registro do campo Status o evento onCellClick é invocado.

Conforme a figura abaixo:

Figura 60: Trecho do código do evento onCellClick

```
if qryOS.State in [dsInsert,dsEdit] then
begin
```

Fonte: Produzida pelo autor, Trecho do código do evento onCellClick do grid que verifica o estado da tela

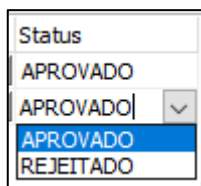
Figura 61: Trecho de código do evento onCellClick 1

```
if Column.PickList.Count > 0 then
begin
  keybd_event (VK_F2, 0, 0, 0);
  keybd_event (VK_F2, 0, KEYEVENTF_KEYUP, 0);
  keybd_event (VK_MENU, 0, 0, 0);
  keybd_event (VK_DOWN, 0, 0, 0);
  keybd_event (VK_DOWN, 0, KEYEVENTF_KEYUP, 0);
  keybd_event (VK_MENU, 0, KEYEVENTF_KEYUP, 0);
end;
```

Fonte:

Produzida pelo autor, cria atalhos para encurtar a ativação do dropdown do campo Status do grid para 1 click (sem esse trecho de código seriam necessários 5 clicks)

Figura 62: Dropdown do Status ativado com 1 click



Fonte: Produzida pelo autor

Para excluir um item de serviço, a OS deve estar em modo de inserção ou modificação e o usuário deve clicar no botão “Excluir Itens” e confirmar a operação. Dessa forma o item será excluído e será subtraída uma unidade do campo item de todos os itens de serviço da mesma OS cujo valor campo “It” era maior do que o do “It” de serviços excluído.

- ✓ Clicar em um item de serviços e em seguida no botão “Excluir Itens”.

Código do botão Excluir Itens:

Figura 63: Código de exclusão de itens de serviço

```

procedure IfrsOS.btnexclui_itensClick(Sender: TObject);
var r :Byte;
item:Integer;
id:Integer;
id_final:Integer;
pos:TBookmark;
begin
  OGeneral.qryPermissao.Locate('programa','AS100',[]);
  if OGeneral.qryPermissao.FieldByName('altera').Value <> 'SIM' then
  begin
    ShowMessage('Acesso negado para seu usuário!');
    Exit;
  end;
  if not cdsOs_Servicos.IsEmpty then
  begin
    r := Application.MessageBox('Deseja realmente excluir este item?', 'Alerta', MB_yesno+MB_iconInformation);
    if r = 6 then
    begin
      item:=cdsOs_Servicos.FieldByName('item_os').Value;
      if cdsOs_Servicos.FieldByName('id_os_servico').Value<>null then
        id:=cdsOs_Servicos.FieldByName('id_os_servico').Value
      else
        id:=0;
      pos := cdsOs_Servicos.GetBookmark;
      cdsOs_Servicos.Last;
    end;
  end;
end;

```

Fonte: Produzida pelo autor

Figura 64: Código de exclusão de itens de serviço 1

```

while (cdsOs_Servicos.FieldByName('id_os_servico').AsInteger<=0)
and (cdsOs_Servicos.FieldByName('item_os').AsInteger>item) do
begin
cdsOs_Servicos.Prior;
id_final:=cdsOs_Servicos.FieldByName('id_os_servico').AsInteger;
end;

cdsOs_Servicos.GotoBookmark(pos);

cdsOs_Servicos.Delete;

cdsOs_Servicos.First;
while not cdsOs_Servicos.Eof do
begin
if cdsOs_Servicos.FieldByName('item_os').Value>item then
begin
cdsOs_Servicos.Edit;
cdsOs_Servicos.FieldByName('item_os').Value:=cdsOs_Servicos.FieldByName('item_os').Value-1;
cdsOs_Servicos.Post;
end;
cdsOs_Servicos.Next;
end;

```

Fonte: Produzida pelo autor

Figura 65: Código de exclusão de itens de serviço 2

```

with cdsEventos do
begin
Filtered := False;
Filter := '((id_os_servico = 0) and (item_os =' + intostr(item) +')) or ((id_os_servico=' +
intostr(id) +') and (id_os_servico>0))';
Filtered := True;
end;
cdsEventos.First;
while not cdsEventos.eof do
begin
cdsEventos.Delete;
end;
cdsEventos.filtered:=False;
with cdsEventos do
begin
Filtered := False;
Filter := '((id_os_servico = 0) and (item_os >' + intostr(item) +')) or ((id_os_servico>' +
intostr(id) +') and (id_os_servico<=' + intostr(id_final) +'))';
Filtered := True;
end;

```

Fonte: Produzida pelo autor

Figura 66: Código de exclusão de itens de serviço 3

```

while not cdsEventos.eof do
begin
  cdsEventos.Edit;
  cdsEventos.FieldName('item_os').Value:=cdsEventos.FieldName('item_os').Value-1;
  cdsEventos.Post;
  cdsEventos.Next;
end;
cdsEventos.filtered:=False;
end;
cdsOs_Servicos.Edit;
cdsOs_Servicos.Post;
end;
end;

```

Fonte: Produzida pelo autor

Os códigos acima não serão detalhados neste documento .

Figura 67: Pede confirmação da Exclusão



Fonte: Produzida pelo autor

Caso o usuário siga os passos acima, o sistema verifica as seguintes situações e executa as seguintes ações:

- ✓ Quando se sabe o id da tabela OS e portanto os itens de serviço estão ligados a uma OS (OS sendo alterada);
- ✓ Quando não se sabe o id da tabela OS e portanto os itens de serviço não estão ligados ainda a uma OS, tendo que ser filtrados por item e por id\_os=0 (OS sendo cadastrada);
- ✓ Quando não se sabe o id do item de serviço e portanto os eventos desse item estão ligados por código a esse item);
- ✓ Quando não se sabe o id do item de serviço e portanto os eventos desse item ainda não estão ligados por código a esse item, tendo que ser filtrados pelo campo item dos itens de serviço e campo item do evento (item de serviço e eventos criados e excluído antes de salvar a OS)
- ✓ Exclui em qualquer uma delas primeiro os eventos ligados ao item de serviço e depois o próprio item de serviço;

Com o item excluído, o sistema reajusta os valores do campo It, subtraindo 1 do campo It dos itens cujo mesmo era maior que o do excluído;

Conforme a figura abaixo:

Figura 68: Itens de Serviço da Tela OS depois da exclusão do item 1

Serviços										
It	Código	Descrição	Lado	Qtd	Pr Unitário	Box	Técnico	Último Evento	Botão	Status
▶ 1	3	PINTURA PORTA DIANTEIRA ESQUERDA		1	200,00				Eventos	APROVADO

Fonte: Produzida pelo autor

Sempre que se navega entre os registros da OS, é executada a função atualizagrid que filtra os itens de serviço novamente, selecionando apenas os registros cujo valor do campo id\_os corresponde ao valor do mesmo na tela OS.

Figura 69: Código da função atualizagrid

```

procedure TfrmOS.atualizagrid();
begin
  cdsOs_Servicos.Close;
  qryOs_servicos.Close;
  qryOs_servicos.SQL.Clear;
  qryOs_servicos.SQL.Add('select * from os_servicos where id_os=:id_os order by item_os');
  qryOs_servicos.ParamByName('id_os').asinteger:=qryOS.FieldByName('id_os').asinteger;
  qryOs_servicos.Open;
  cdsOs_Servicos.Open;
end;

```

Fonte: Produzida pelo autor, esse código é executado quando se precisa recarregar a consulta que alimenta o grid, normalmente quando se navega entre as OS, para filtrar sempre os itens de serviço ligados a OS mostrada.

### 3.20 Decisões relevantes tomadas durante o desenvolvimento da parte de itens de serviço

Porque foi colocado o campo “Status”:

Mesmo sendo possível excluir um item, foi avaliado que era importante também manter o registro daqueles itens que, por algum motivo, não foram aprovados. Foi concluído que essa informação pode ser muito útil na gestão da oficina. Esses itens não podem mais ser editados e nem contabilizam no total da OS.

Como inserir o técnico em um item da OS:

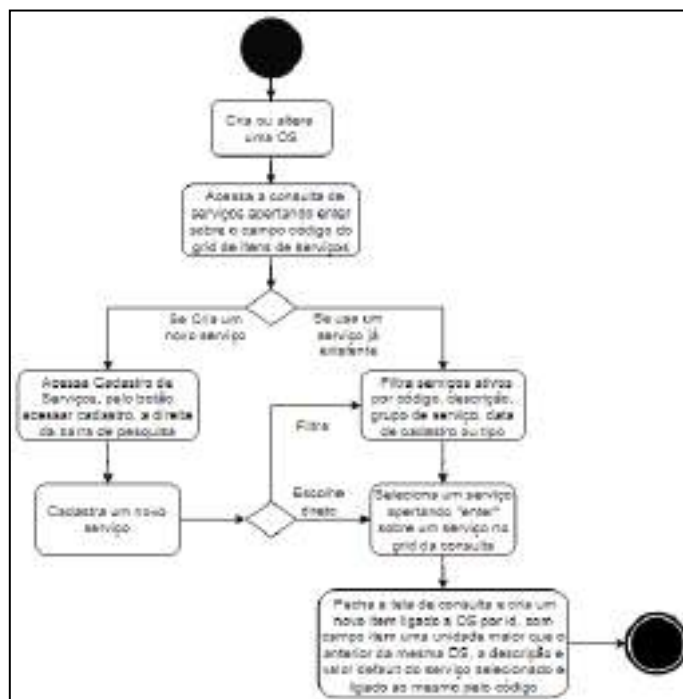
Informar o técnico em um item de serviço, sem tornar o cadastro lento, foi um desafio técnico complexo para o nível de conhecimento que eu tinha na época. Foram estudadas alternativas como abrir uma tela de consulta para selecionar ou permitir digitação do nome e depois validar se existia um técnico com aquele nome, mas foi encontrada uma forma de criar em um uma célula em um com um menu dropdown que mostra registros de uma tabela e permite sua seleção, método que acabou por ser adotado em função de ser mais rápido e confiável.

Muitos sistemas obrigam o usuário a selecionar um serviço já cadastrado na hora de adicionar um item de serviço em uma OS, contudo é entendido pela empresa onde trabalho que, via de regra, é melhor dar o máximo de possibilidades ao usuário, permitindo assim que adicione itens de serviço do zero sem puxar um modelo já cadastrado da tela Consulta Serviços.

Segue abaixo um UML do CRUD itens de serviço dentro de uma OS: Diagrama de Atividades dos Itens de Serviço.

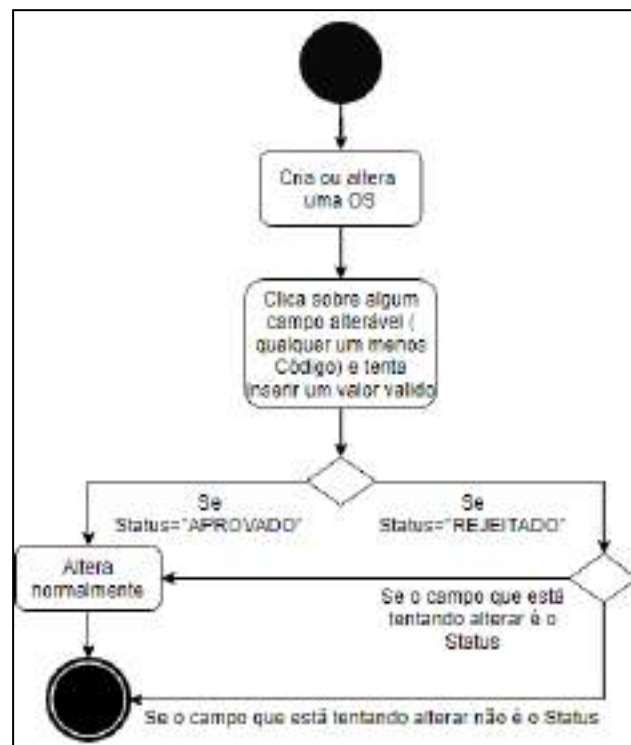
Conforme a figura abaixo:

Figura 70: Inclusão de um Novo Item de Serviço



Fonte: Produzida pelo autor

Figura 71: Alteração de um item de serviço



Fonte: Produzida pelo autor

Figura 72: Exclusão de Itens de serviços através do botão “ Excluir” da OS



Fonte: Produzida pelo autor

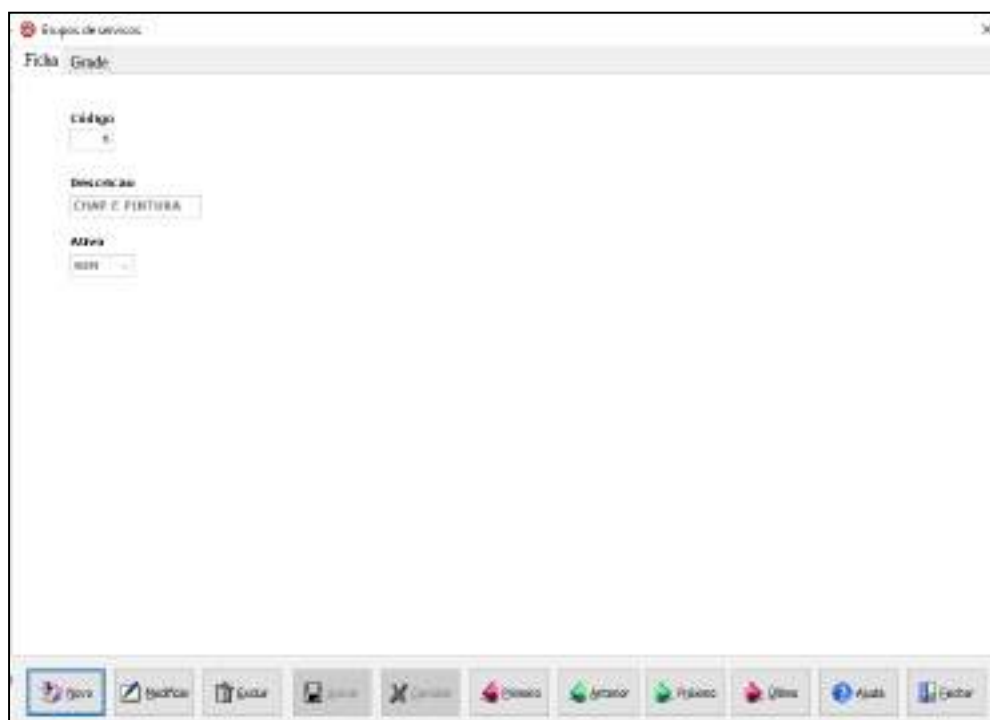
Figura 73: Exclusão de Itens de serviços através do botão “Excluir Itens” da OS



Fonte: Produzida pelo autor



Figura 74: Tela Grupos de Serviços



Fonte: Produzida pelo autor

### 3.21 Características e funcionalidades básicas da tela

Cadastrar grupos de serviços, que vão ser usados na tela Cadastro de Serviços para definir a qual grupo cada serviço pertence.

Segue todos os padrões previamente mencionados para telas de CRUD.

Relação com o banco:

É ligada diretamente a tabela "grupos\_servicos".

### 3.22 Motivos para o desenvolvimento da tela

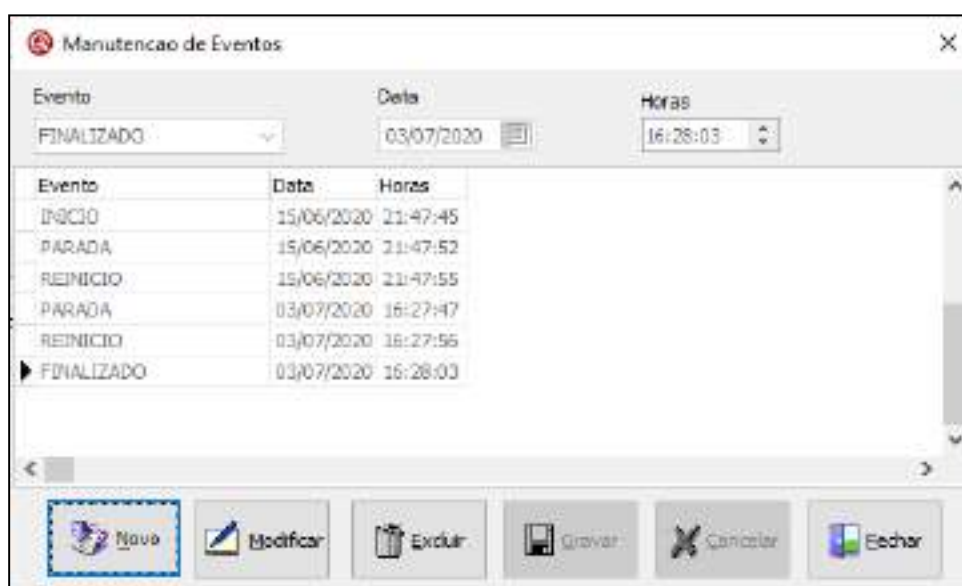
Quando o cliente cadastra um item de serviço em uma OS ele quer, antes de qualquer coisa, velocidade e facilidade na realização desta ação. Em banco de dados numa situação real, podem existir dezenas de tipos de serviço, e o usuário tem que achar o modelo de serviço requisitado dentro dessa enorme lista.

Claro que o usuário pode filtrar pela descrição do serviço requisitado na tela Consulta Serviços, mas nem sempre o serviço cadastrado terá o mesmo nome do serviço que foi requisitado, e o usuário muitas vezes não lembra da descrição com a qual o serviço foi cadastrado.

Para solucionar esse problema foi criada a tela Grupos de Serviços. Com ela é possível categorizar os serviços e, invés de procurar nas dezenas de registros da tela Consulta de Serviços ou perder tempo chutando nomes, o usuário pode simplesmente filtrar pelo grupo ao qual o serviço requisitado pertence e reduzir a lista enormemente.

Além da vantagem no aspecto operacional, a existência de uma divisão dos serviços em grupos auxilia na gestão da oficina, uma vez que permite a geração de relatórios que avaliam cada grupo individualmente.

Figura 75: Tela Manutenção de Eventos



Fonte: Produzida pelo autor

### 3.23 Funcionalidades básicas da tela Manutenção de Eventos

- ✓ Para tornar o processo de cadastro de OS mais dinâmico, essa tela não segue a maioria dos padrões do sistema, seguindo apenas os padrões: 1.1; 2;3;4;5; 6.1; 6.2; 6.3 e 6.7 até 6.10;
- ✓ Ela possui apenas os botões “novo”, “modificar”, “excluir”, “gravar”, “cancelar” e “fechar”;
- ✓ Ela é acessada apenas quando o usuário está criando ou alterando uma OS e pressiona “enter” sobre o código de algum item de serviços dessa OS;
- ✓ Ela vai mostrar sempre apenas os registros ligados ao item através do qual ela foi chamada, como mostram os recortes abaixo.
- ✓ A tela Manutenção de Eventos é uma tela ligada a tabela “os\_eventos” do banco;

### 3.24 Funcionamento da Tela Manutenção de Eventos

- Ela é chamada ao pressionar “enter” ou clicar sobre um registro da coluna “Botão” do grid;

Figura 76: Grid de Itens de Serviços da Tela OS

It	Código	Descrição	Lado	Qtd	Pr Unitário	Box	Técnico	Último Evento	Botão	Status
1	3	PINTURA PORTA DIANTEIRA ESQUERDA GFD		2	200,00			FINALIZADO	Eventos	APROVADO
2	3	PINTURA PORTA DIANTEIRA ESQUERDA		1	200,00			INICIO	Eventos	APROVADO

Fonte: Produzida pelo autor

- Clica no botão eventos de um item de serviços, invocando o evento onCellClick

Figura 77: Trecho de Código do onCellClick do grid de itens de serviço

```
if gryOS.State in [dsInsert,dsEdit] then
begin
```

Fonte: Produzida pelo autor, verdadeira se a tela está no modo de inserção ou edição

Figura 78: Código para abrir a tela Manutenção de Eventos do onClick

```

if cdsOs_servicos.FieldByName('status').asstring='APROVADO' then
begin
  if DBGrdl.SelectedField.FieldName = 'evento' then
  begin
    frmManutencao_evento.Showmodal;
  end;
end;
end;
end;

```

Fonte: Produzida pelo autor, esse código verifica se o campo clicado pertence a coluna botão e se, nesse registro, o status está aprovado antes de abrir a tela Manutenção de Eventos.

Antes de abrir qualquer tela o evento onShow da tela que está sendo aberta é invocado, no onShow da tela Manutenção de Eventos são filtrados os eventos do item de serviços que a chamou.

Conforme a figura abaixo:

Figura 79: Código onShow da tela Manutenção de Eventos

```

procedure TfrmManutencao_evento.FormShow(Sender: TObject);
begin
  frmOS.cdsOs_servicos.FieldByName('id_os_servico').AsInteger:=0;
begin
  with frmOS.cdsEventos do
  begin
    Filtered := False;
    Filter := '(id_os_servico = 0) and item_os='+inttostr(frmOS.cdsOs_Servicos.FieldByName('item_os').asinteger)+'';
    Filtered := True;
  end;
end;
begin
  with frmOS.cdsEventos do
  begin
    Filtered := False;
    Filter := 'id_os_servico = '+(QuotedStr(frmOS.cdsOs_servicos.FieldByName('id_os_servico').value));
    Filtered := True;
  end;
end;
end;

```

Fonte: Produzida pelo autor, essa figura mostra o código para abertura da tela de Manutenção da OS e filtragem dos eventos ligados ao item que à chamou.

Com as condições satisfeitas, a tela de Manutenção de Eventos é aberta

Conforme mostra a figura abaixo:

Figura 80: Tela Manutenção de Eventos de Serviços

It.	Código	Descrição	Evento	Data	Horas
1	3	PINTURA PORTA DIANTEIR	INICIO	05/07/2020	19:41:42
2	3	PINTURA PORTA DIANTEIR	INICIO	05/07/2020	19:43:42

Fonte: Produzida pelo autor

No primeiro evento cadastrado para um item de serviços, o campo “evento” recebe o valor "INICIO" (inalterável), “data” recebe a data de abertura da OS(alterável para 3 dias antes e n dias depois) e hora recebe hora atual.

Figura 81: Tela Manutenção de Eventos 1

Evento	Data	Horas
INICIO	05/07/2020	19:31:12

Fonte: Produzida pelo autor

Os eventos seguintes nunca podem começar antes do evento anterior, isso também é validado pelo sistema, e o campo “evento” nunca será de “INICIO” ou igual ao campo “Evento” do evento anterior, como mostram os recortes abaixo;

Figura 82: Tela Manutenção de Eventos 2

Evento	Data	Horas
FINALIZADO	05/07/2020	19:48:33
PARADA		
FINALIZADO		
INICIO	05/07/2020	19:41:42
PARADA	05/07/2020	19:47:59
REINICIO	05/07/2020	19:48:03
* FINALIZADO	05/07/2020	19:48:33

Fonte: Produzida pelo autor

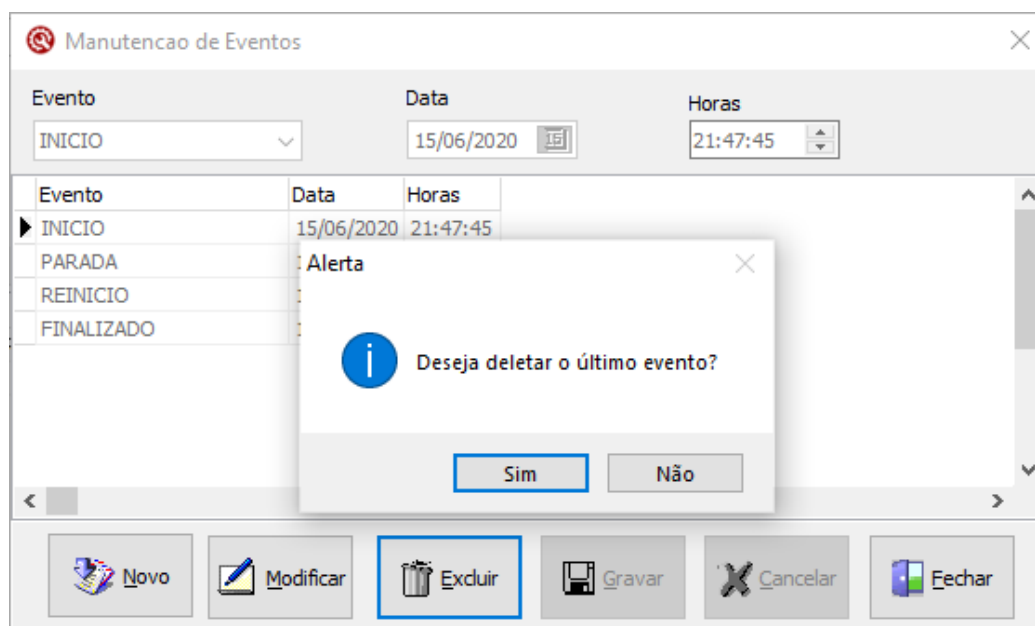
Pode haver infinitos eventos de “parada” e “reinício” mas o primeiro evento de um item de serviços é sempre de “início” e nunca vai haver outro evento depois do evento “finalizado” ligado a um mesmo item de serviços;

Só é possível excluir ou modificar o último evento ligado a um item de serviços, quando o usuário clicar em modificar ou excluir, o sistema já inicia o processo de exclusão ou modificação do último evento, como mostra o recorte abaixo;

Os códigos dos botões “ Novo” e “Modificar” não serão detalhados por serem muito extensos.

Ao clicar no botão Excluir da tela Manutenção de Eventos se inicia o processo de exclusão do último evento ligado a um item de serviços, como mostra a mensagem abaixo:

Figura 83: Tela Manutenção de Eventos 3



Fonte: Produzida pelo autor

Figura 84: Código para exclusão de um evento

```

procedure TfrmManutencao_evento.btnExcluirClick(Sender: TObject);
var  e :Byte;
begin
  e := Application.MessageBox('Deseja deletar o último evento?', 'Alerta', MB_yesno+MB_iconInformation);
  if e = 6 then
  begin
    begin
      frmOS.cdsEventos.Last;
      frmOS.cdsEventos.Delete;
      dbqxdL.Enabled:=false;

      dbcbEvento.Enabled:=false;
      dbedtdata.Enabled:=false;
      dbedthoras.Enabled:=false;

      data_max:=intyear(now,10);
    end;
  end;
end;

```

Fonte: Produzida pelo autor

Durante o processo de fechamento da tela de Manutenção de Eventos, o sistema faz com que o último evento cadastrado apareça ao lado esquerdo do botão que chamou a tela de eventos, no campo “último evento”, que é um campo calculado;

Figura 85: Código do campo calculado do cdsOs\_serviços

```

procedure TfrmOS.cdsOs_servicosCalcFields(DataSet: TDataSet);
begin
  if cdsEventos.Active=False then
    cdsEventos.Open;

  if frmOS.cdsOs_servicos.fieldbyname('id_os_servico').AsInteger=0 then
  begin
    with cdsEventos do
    begin
      Filtered := False;
      Filter := '(id_os_servico = 0) and (item_os='+inttostr(cdsOs_Serviços.FieldByName('item_os').asinteger)+' )';
      Filtered := True;
    end;
  end
  else
  begin
    with cdsEventos do
    begin
      Filtered := False;
      Filter := 'id_os_servico = ' + (QuotedStr(cdsOs_servicos.fieldbyname('id_os_servico').value));
      Filtered := True;
    end;
  end;
  cdsEventos.Last;
  cdsOs_servicos.FieldByName('evento_atual').value:=cdsEventos.FieldByName('evento').value;
end;

```

Fonte: Produzida pelo autor, o código mostrado na figura faz com que, no fechamento da tela Manutenção de Eventos, o campo calculado evento\_atual, do registro que chamou a tela, receba o valor da string do último evento ligado a ele.

Coloca o valor do último registro do campo “evento” dos eventos ligados ao item que o chamou.

Preenche o campo Último Evento e coloca o cursor no campo “Status”

Figura 86: Último Evento cadastrado em um item de serviço

Último Evento	Botão
FINALIZADO	Eventos

Fonte: Produzida pelo autor

As alterações feitas na tela Manutenção de Eventos só são salvas na tabela os\_eventos quando a OS, a qual os itens de serviço aos quais os eventos (que foram alterados, incluídos ou excluídos) estão ligados, é salva. Caso a operação (de inserção ou edição) na OS em questão seja cancelada, todas as alterações nos itens dessa OS e nos eventos desses itens são descartadas;



Figura 87: Código que salva as Alterações dos Eventos da OS no Banco

```

odsEventos.Filtered:=False;
odsEventos.First;
while not odsEventos.eof do
begin
  if odsEventos.FieldByName('id_os_servico').AsInteger=0 then
  begin
    dmxtra.qryConsultas.Close;
    dmxtra.qryConsultas.SQL.Clear;
    dmxtra.qryConsultas.SQL.Add('select * from os_servicos where id_os=id_os and item_os=item_os');
    dmxtra.qryConsultas.ParamByName('id_os').AsInteger:=qryOS.FieldByName('id_os').AsInteger;
    dmxtra.qryConsultas.ParamByName('item_os').AsInteger:=odsEventos.FieldByName('item_os').AsInteger;
    dmxtra.qryConsultas.Open;
    odsEventos.Edit;
    odsEventos.FieldByName('id_os_servico').Value:=dmxtra.qryConsultas.FieldByName('id_os_servico').Value;
    odsEventos.Post;
  end;
  odsEventos.Next;
end;
odsEventos.ApplyUpdates(-1);
odsEventos.Filtered := True;

```

Fonte: Produzida pelo autor

### 3.25 Motivos e decisões tomadas durante o desenvolvimento da tela

#### Manutenção de Eventos

Os motivos da criação da tela foram a necessidade da definição de estados de cada item de serviço (se está em andamento, suspenso ou finalizado) e a necessidade de informar a hora de início e conclusão de cada item de serviço. A solução encontrada com a tela Manutenção de Eventos contempla ambos os aspectos das seguintes formas:

O último evento cadastrado representa o estado atual do item de serviço, com a vantagem de mostrar todos os estados pelos quais ele passou;

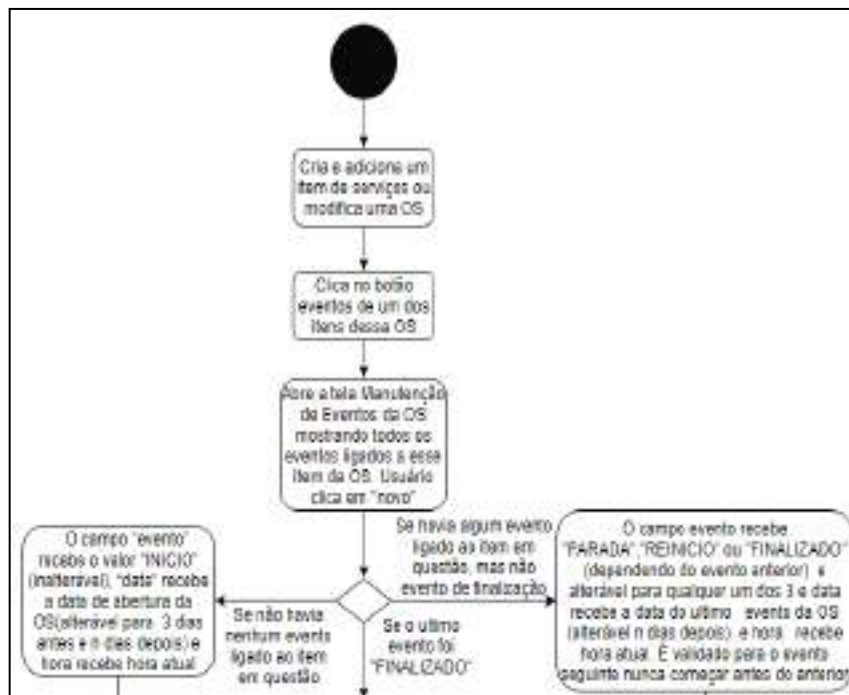
A data e hora do evento de início e data e hora do evento de finalização podem ser usadas como data e hora de início e data e hora de conclusão, com a vantagem de ter informações sobre o todas os momentos e intervalos de tempo em que ele ficou parado até ele ser reiniciado.

A tela de Manutenção de Eventos permite o controle do tempo exato utilizado para a conclusão de cada serviço (quando o mesmo foi finalizado);

Com a abordagem decidida, foi apenas uma questão de decidir a melhor forma de chamar a tela e a melhor solução técnica para o problema.

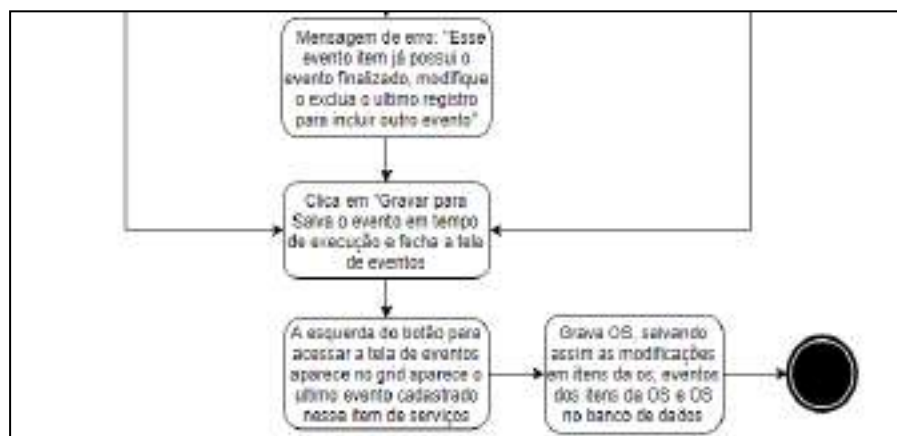
Conforme a figura abaixo: Diagrama de Atividades dos Eventos dos Itens da OS.

Figura 88: Inclusão de Eventos de Itens de Serviços



Fonte: Produzida pelo autor

Figura 89: Inclusão de Eventos de Itens de Serviços (continuação)



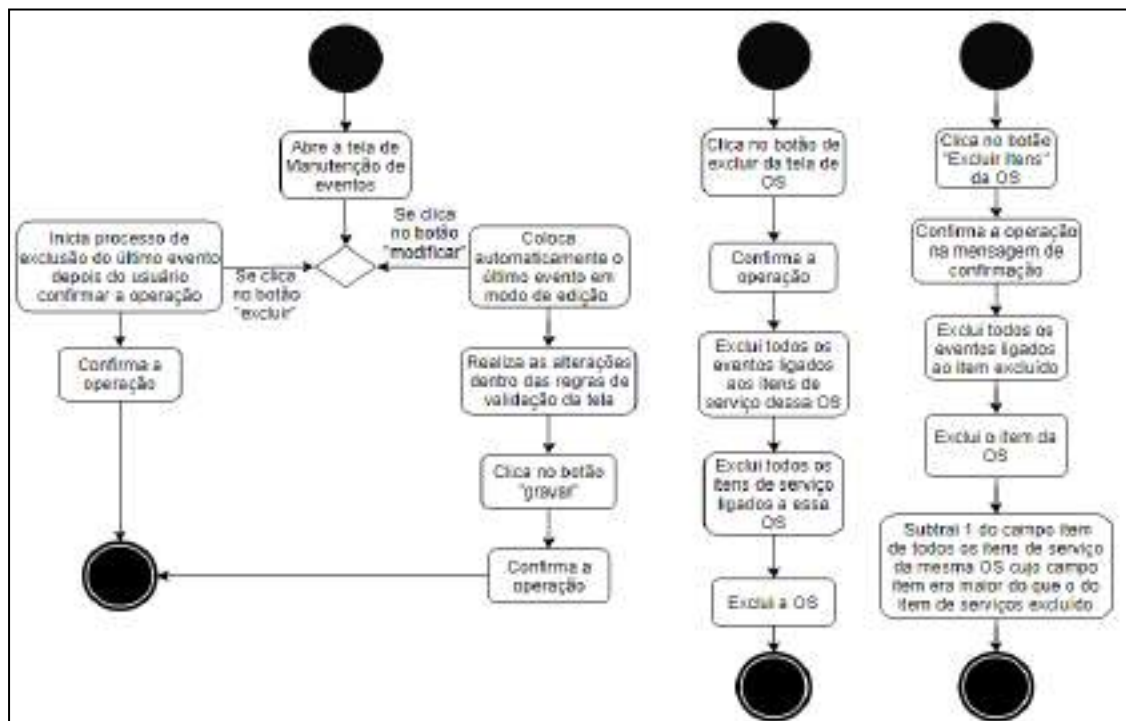
Fonte: Produzida pelo autor

Figura 90: Alteração de Eventos de Itens de Serviços



Fonte: Produzida pelo autor

Figura 91: Exclusão de Eventos de Itens de Serviços



Fonte: Produzida pelo autor

Figura 92: Tela Cadastro de Veículos



Fonte: Produzida pelo autor

### 3.26 Características e funcionalidades básicas da tela Cadastro de Veículos

Ela segue todos os padrões de layout, regras e desenvolvimento comentados nos tópicos Padrões do Sistema e Parões de Desenvolvimento.

A tela de Cadastro de Veículos é uma tela ligada a tabela do banco “veiculos” e segue todos os padrões previamente mencionados para telas de inserção e alteração.

Ela é utilizada na OS para identificar o veículo que está em manutenção;

Ela é utilizada na OS para identificar o veículo que está em manutenção;

### 3.27 Particularidades da Tela Cadastro de Veiculos

A tela Cadastro de Veículos tem duas particularidades:

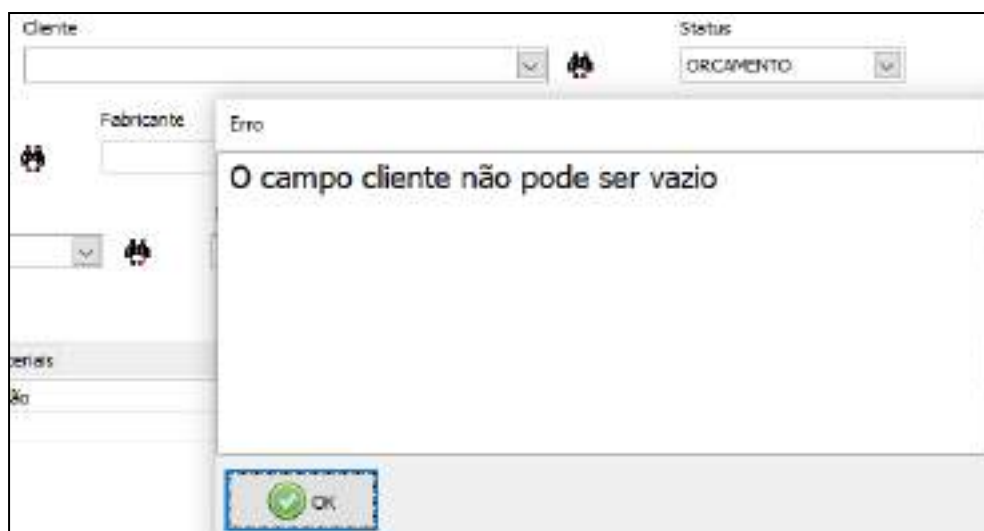
Seu campo cliente pode ser alterado na tela de OS sem passar pelo Cadastro de Veículos;

O campo “quilometragem” da OS não tem relação direta com o veículo, mas mostra uma mensagem de alerta caso exista uma OS com o mesmo veículo e quilometragem superior;

### 3.28 Informando ou Alterando um Veículo em uma OS

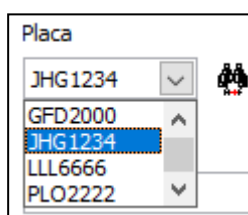
Quando entra em modo de inserção ou edição, o campo cliente recebe foco e só é possível sair do campo cliente quando este deixar de ser nulo. Isso é feito porque o campo “Placa” depende do valor do campo “Cliente”, já que ao abrir o dropdown ou a tela de consulta, serão mostrados apenas os veículos do cliente informado.

Figura 93: Mensagem de erro ao tentar sair do campo “Cliente” com ele vazio



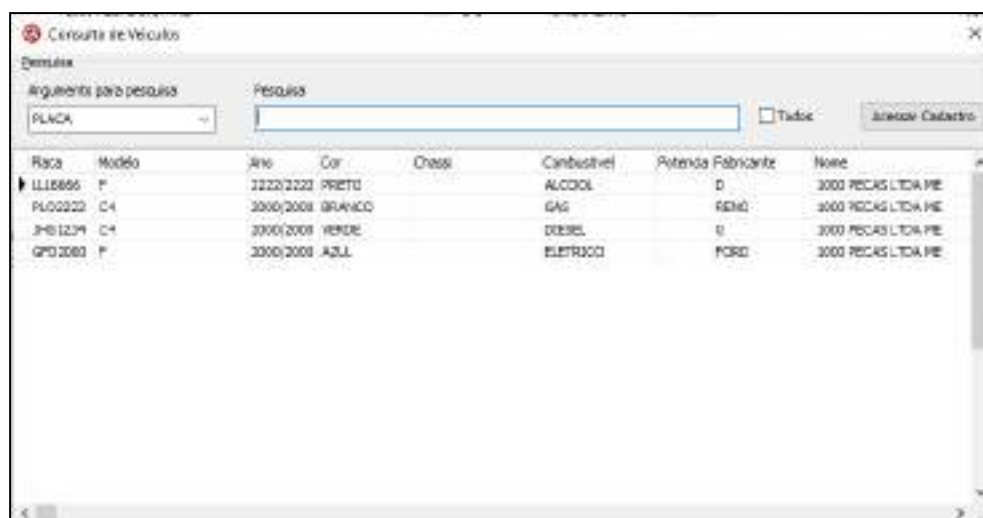
Fonte: Produzida pelo autor

Figura 94: Dropdown mostrando os veículos do cliente selecionado



Fonte: Produzida pelo autor

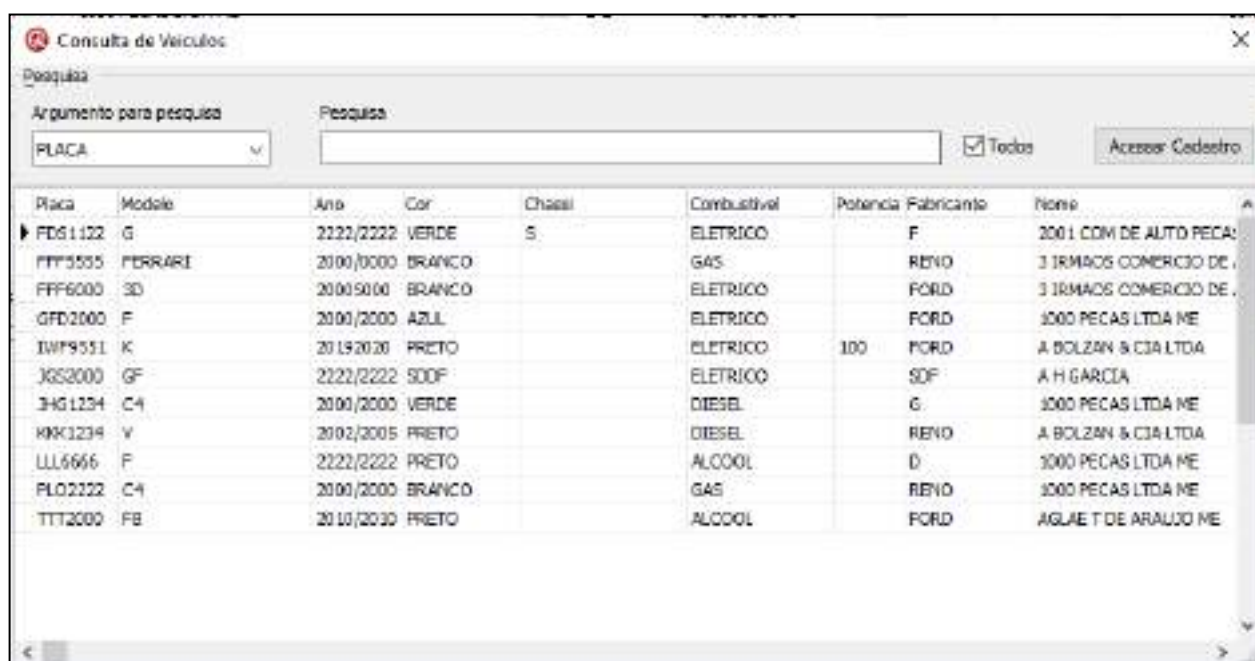
Figura 95: Tela Consulta de Veículos mostrando os veículos do cliente selecionado



Fonte: Produzida pelo autor

Se o veículo procurado já estiver cadastrado no sistema mas não estiver ligado ao cliente, é possível abrir a tela Consulta de Veículos e marcar o checkbox “Todos”, de forma a mostrar todos os veículos ativos cadastrados no sistema.

Figura 96: Tela Consulta de Veículos mostrando os veículos ativos do sistema

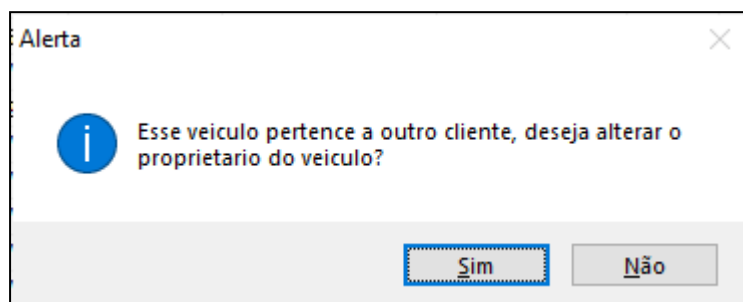


Fonte: Produzida pelo autor

Ao encontrar o veículo procurado (podendo filtrar por ano, combustível, cor, fabricante, modelo ou placa), é possível selecioná-lo pressionando “enter” sobre o registro onde ele se encontra.

Caso ele não pertença ao cliente informado na OS, o sistema pergunta se o usuário quer alterar o proprietário do veículo.

Figura 97: Mensagem de Alerta



Fonte: Produzida pelo autor

Se sim, o campo cliente do veículo em questão é alterado para o cliente previamente informado na OS, a tela Consulta de Veículos é fechada e o campo placa da OS será preenchido com o id do veículo selecionado, mostrando o campo placa para o usuário e os demais campos de veículo aparecem a direita só que de forma não editável.

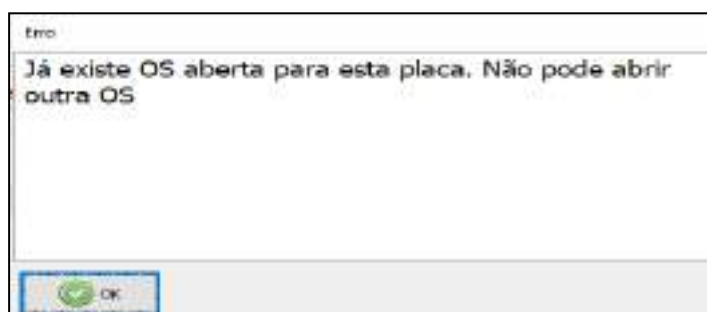
Figura 98: Informações do veículo mostradas na OS



Fonte: Produzida pelo autor

Caso já exista uma OS aberta com o veículo em questão, o sistema dispara uma mensagem de erro e não deixa sair do campo até ele ser validado corretamente.

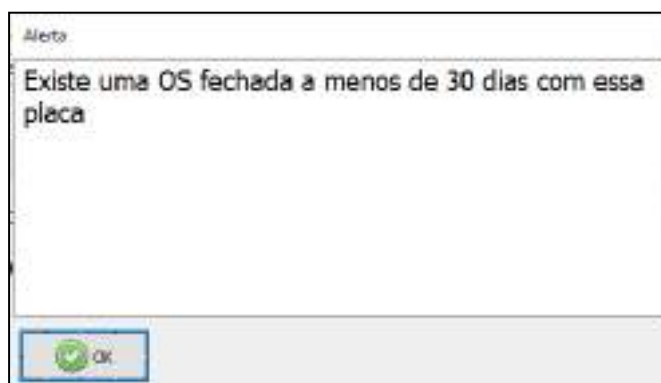
Figura 99: Mensagem de Erro



Fonte: Produzida pelo autor

Caso exista uma ou mais OS fechadas com o veículo em questão a menos de 30 dias, o sistema dispara uma mensagem de alerta mas permite sair do campo.

Figura 100: Tela de Alerta 1



Fonte: Produzida pelo autor

### 3.29 Códigos da Parte de Veículos da OS

Código acionado quando o dropdown de um campo é acionado:

Figura 101: Evento onDropDown do Campo Placa

```

procedura TfmOS.dbikcbbPlacaDropDown(Sender: TObject);
var id_veiculo:Integer;
begin
  if qryOS.FieldByName('id_veiculo').value<>null then
  begin
    id_veiculo:=qryOS.FieldByName('id_veiculo').value;
    DMGeral.qryveiculos.Close;
    DMGeral.qryveiculos.SQL.Clear;
    DMGeral.qryveiculos.sql.add('select * from veiculos where ativo=:ativo and '
    +' (id_cliente=:id_cliente or id_veiculo=:id_veiculo) order by placa');
    DMGeral.qryveiculos.parambyname('ativo').value:='SIM';
    DMGeral.qryveiculos.parambyname('id_cliente').value:=qryOS.FieldByName('id_cliente').value;
    DMGeral.qryveiculos.parambyname('id_veiculo').value:=qryOS.FieldByName('id_veiculo').value;
    DMGeral.qryveiculos.Open;
    dbikcbbPlaca.KeyValue:=id_veiculo;
    qryOS.FieldByName('id_veiculo').value:=id_veiculo;
  end
  else
  begin
    DMGeral.qryveiculos.Close;
    DMGeral.qryveiculos.SQL.Clear;
    DMGeral.qryveiculos.sql.add('select * from veiculos where ativo=:ativo and '
    +' (id_cliente=:id_cliente or id_veiculo=:id_veiculo) order by placa');
    DMGeral.qryveiculos.parambyname('ativo').value:='SIM';
    DMGeral.qryveiculos.parambyname('id_cliente').value:=qryOS.FieldByName('id_cliente').value;
    DMGeral.qryveiculos.parambyname('id_veiculo').value:=qryOS.FieldByName('id_veiculo').value;
    DMGeral.qryveiculos.Open;
  end;
end;

```

Fonte: Produzida pelo autor, o código acima verifica se o campo placa está vazio, caso esteja, simplesmente consulta os veículos do cliente informado, caso contrário consulta os veículos do cliente informado e ainda o veículo que está ocupando o campo (mesmo que ele não pertença ao cliente informado).



Código acionado quando uma tecla é pressionada com o cursor posicionado sobre o campo placa:

Figura 102: Evento onKeyDown do campo placa

```

procedure TfrmOS.dbkcbPlacaKeyPress(Sender: TObject; var Key: Char);
var x (Byte);
begin
  if dbkcbPlaca.KeyValue=nil then
  begin
    if key=#13 then
    begin
      if frmconsulta_veiculos = nil then
        Application.CreateForm(Tfrmconsulta_veiculos, frmconsulta_veiculos);

        if frmconsulta_veiculos.qryconsulta_veiculos.Active then
          frmconsulta_veiculos.qryconsulta_veiculos.Locate('id_veiculo', qryOS.FieldByName('id_veiculo').Value, []);
        frmConsulta_Veiculos.ShowModal;
      end;
    end;
  end;
end;

```

Fonte: Produzida pelo autor o código acima verifica se o campo “Placa” está vazio e verifica se a tecla “enter” foi pressionada, se sim abre a tela de consulta de veículos mostrando os veículos do cliente.

Código acionado quando a ação de “saída” do campo código é iniciada:

Figura 1033: Evento onExit do campo placa

```

DMExtra.qryConsultas.Close;
DMExtra.qryConsultas.SQL.Clear;
DMExtra.qryConsultas.SQL.Add('select * from os where id_veiculo=:id_veiculo');
DMExtra.qryConsultas.SQL.Add(' and situacao=:situacao and data_fechamento>:data_fechamento');
DMExtra.qryConsultas.parambyname('id_veiculo').value:=dbkcbPlaca.KeyValue;
DMExtra.qryConsultas.parambyname('situacao').value='FECHADA';
DMExtra.qryConsultas.parambyname('data_fechamento').AsDateTime:=IncDay(now, -30);
DMExtra.qryConsultas.Open;
if not DMExtra.qryConsultas.IsEmpty then
begin
  alerta('Alerta', 'Existe uma OS fechada a menos de 30 dias com essa placa');
end;

```

Fonte: Produzida pelo autor, o trecho de código acima verifica se existe uma OS fechada com o mesmo veículo com data de fechamento igual ou maior do que 30 dias antes da data atual, se sim dispara uma mensagem de alerta.

Figura 1044: Evento onExit do campo placa 1

```

if qryOS.State in [dsInsert] then
begin
  DMEExtra.qryConsultas.Close;
  DMEExtra.qryConsultas.SQL.Clear;
  DMEExtra.qryConsultas.SQL.Add('select * from os where id_veiculo=:id_veiculo and situacao=:situacao');
  DMEExtra.qryConsultas.parambyname('id_veiculo').value:=dblkcbbPlaca.KeyValue;
  DMEExtra.qryConsultas.parambyname('situacao').value:='ABERTA';
  DMEExtra.qryConsultas.Open;
  if not DMEExtra.qryConsultas.IsEmpty then
  begin
    alerta('Erro', 'Já existe OS aberta para esta placa. Não pode abrir outra OS');
    dblkcbbPlaca.SetFocus;
  end;
end;
if qryOS.State in [dsedit] then
begin
  DMEExtra.qryConsultas.Close;
  DMEExtra.qryConsultas.SQL.Clear;
  DMEExtra.qryConsultas.SQL.Add('select * from os where id_veiculo=:id_veiculo');
  DMEExtra.qryConsultas.SQL.Add(' and situacao=:situacao and id_os<=:id_os');
  DMEExtra.qryConsultas.parambyname('id_veiculo').value:=qryOS.FieldByName('id_veiculo').value;
  DMEExtra.qryConsultas.parambyname('situacao').value:='ABERTA';
  DMEExtra.qryConsultas.parambyname('id_os').value:=StrToInt(dbetid_os.text);
  DMEExtra.qryConsultas.Open;
  if not DMEExtra.qryConsultas.IsEmpty then
  begin
    alerta('Erro', 'Já existe OS aberta para esta placa. Não pode abrir outra OS');
    dblkcbbPlaca.SetFocus;
  end;
end;

```

Fonte: Produzida pelo autor, o trecho de código acima verifica se existe uma OS aberta com o mesmo veículo. É feita uma verificação caso a OS esteja em modo de inserção e para o modo de edição. Isso porque, modo de edição, é preciso desconsiderar, na busca por OS abertas com mesmo veículo, a própria OS que está sendo modificada

A função placa é executada sempre que é mudado o registro ou mudado o estado (navegação para inserção ou edição ou vice e versa) da tela OS. O código da função placa é mostrado na figura abaixo:

Figura 1055: Função placa

```

procedure TfrmOS.placa;
begin
  if databela.State in [dsEdit, dsInsert] then
  begin
    dgeneral.qryveiculos.Close;
    dgeneral.qryveiculos.SQL.Clear;
    dgeneral.qryveiculos.SQL.Add('select * from veiculos where id_veiculo=:veiculo');
    dgeneral.qryveiculos.ParamsByName('veiculo').value:=qryOS.FieldByName('id_veiculo').asinteger;
    dgeneral.qryveiculos.Open;
    cb1kcbbPlaca.KeyValue:=qryOS.FieldByName('id_veiculo').asinteger;
  end;
  if databela.State in [dsBrowse] then
  begin
    dgeneral.qryveiculos.Close;
    dgeneral.qryveiculos.SQL.Clear;
    dgeneral.qryveiculos.SQL.Add('select * from veiculos where id_veiculo=:veiculo');
    dgeneral.qryveiculos.ParamsByName('veiculo').value:=qryOS.FieldByName('id_veiculo').asinteger;
    dgeneral.qryveiculos.Open;
    cb1kcbbPlaca.KeyValue:=qryOS.FieldByName('id_veiculo').asinteger;
  end;

  edtmodelo.text:=dgeneral.qryveiculos.fieldbyname('modelo').asstring;
  edtfabricante.text:=dgeneral.qryveiculos.fieldbyname('fabricante').asstring;
  edtcor.text:=dgeneral.qryveiculos.fieldbyname('cor').asstring;
  edtano.text:=dgeneral.qryveiculos.fieldbyname('ano').asstring;
end;

```

Fonte: Produzida pelo autor, o trecho de código acima atualiza a query veículo para filtrar os veículos dependo do cliente e atualiza demais campos expositivos (não ligados ao banco) relacionados ao veículo

### Código de executado na abertura da tela Consulta de Veículos:

Figura 1066: Código onShow da tela Consulta de Veículos

```

procedure TfrmConsulta_veiculos.FormShow(Sender: TObject);
begin
  chkTodos.Checked:=False;
  edtPesquisa.Text:='';
  cbb1.ItemIndex:=5;
  if chkTodos.Checked then
  begin
    qryconsulta_veiculos.close;
    qryconsulta_veiculos.sql.clear;
    qryconsulta_veiculos.sql.add('select veiculos.*,clientes.nome from veiculos join clientes');
    qryconsulta_veiculos.sql.add('on veiculos.id_cliente=clientes.id_cliente');
    qryconsulta_veiculos.sql.add(' where veiculos.ativo=:ativo order by :cbb');
    qryconsulta_veiculos.parambyname('ativo').value:='SIM';
    qryconsulta_veiculos.parambyname('cbb').value:=cbb1.Text;
    qryconsulta_veiculos.open;
  end;
  edtPesquisa.SetFocus;
end;

```

Fonte: Produzida pelo autor, o código acima filtra os veículos que pertencem ao cliente da OS.

Código executado quando o usuário clica no checkbox da tela de consulta de veículos:

Figura 1077: Evento onClick do checkbox “Todos”

```

if chkTodos.Checked then
begin
  qryconsulta_veiculos.close;
  qryconsulta_veiculos.sql.clear;
  qryconsulta_veiculos.sql.add('select veiculos.*,clientes.nome from veiculos');
  qryconsulta_veiculos.sql.add('left join clientes on veiculos.id_cliente=clientes.id_cliente');
  qryconsulta_veiculos.sql.add('where veiculos.ativo=ativo and '+SQLPesq+' order by '+cbbl.text);
  qryconsulta_veiculos.paramsbyname('ativo').value:='SIM';
  qryconsulta_veiculos.open;
end
else
begin
  qryconsulta_veiculos.close;
  qryconsulta_veiculos.sql.clear;
  qryconsulta_veiculos.sql.add('select veiculos.*,clientes.nome from veiculos');
  qryconsulta_veiculos.sql.add('join clientes on veiculos.id_cliente=clientes.id_cliente');
  qryconsulta_veiculos.sql.add('where veiculos.ativo=ativo and ' + SQLPesq + ' and veiculos.id_cliente=cliente');
  qryconsulta_veiculos.sql.add('order by '+cbbl.text);
  qryconsulta_veiculos.paramsbyname('ativo').value:='SIM';
  qryconsulta_veiculos.paramsbyname('cliente').value:=frmOS.FieldByName('id_cliente').value;
  qryconsulta_veiculos.open;
end;
end;

```

Fonte: Produzida pelo autor, o código acima realiza toda a filtragem novamente sempre que é desmarcando ou marcado o checkbox, filtrando em todos os registros ativos quando está marcado e apenas os veículos pertencentes ao cliente quando desmarcado.

Código executado quando é pressionada uma tecla com o cursor sobre um registro do grid da tela de consulta de veículos:

Figura 1088: Evento onKeyPress do grid Consulta de veículos

```

procedure TfrmConsulta_veiculos.dbgrdDBGridKeyPress(Sender: TObject;
var Key: Char);
var r: Integer;
begin
  if Key = #13 then
  begin
    Key := #0;
    if frmOS.Showing=True then
    begin
      if qryconsulta_veiculos.FieldByName('id_cliente').Value<>frmOS.FieldByName('id_cliente').value then
      begin
        r := Application.MessageBox('Este veiculo pertence a outro cliente, deseja alterar o proprietario do veiculo?',
        'Alerta', MB_YesNo+MB_IconInformation);
        if r = 6 then
        begin
          frmOS.FieldByName('id_veiculo').asinteger := qryconsulta_veiculos.FieldByName('id_veiculo').asinteger;
          dmextra.qryconsultas3.close;
          dmextra.qryconsultas3.sql.clear;
          dmextra.qryconsultas3.sql.add('select * from veiculos where id_veiculo=id_veiculo');
          dmextra.qryconsultas3.paramsbyname('id_veiculo').value:=frmOS.FieldByName('id_veiculo').value;
          dmextra.qryconsultas3.open;
          dmextra.qryconsultas3.edit;
          frmOS.dbknobFlaca.SetFocus;
        end
      end
    end
  end;
end;

```

Fonte: Produzida pelo autor, o código acima verifica se o a tela OS está aberta, se o botão pressionado foi “enter”, se o veículo pertence ao cliente e ou não (caso em que o sistema vai pedir

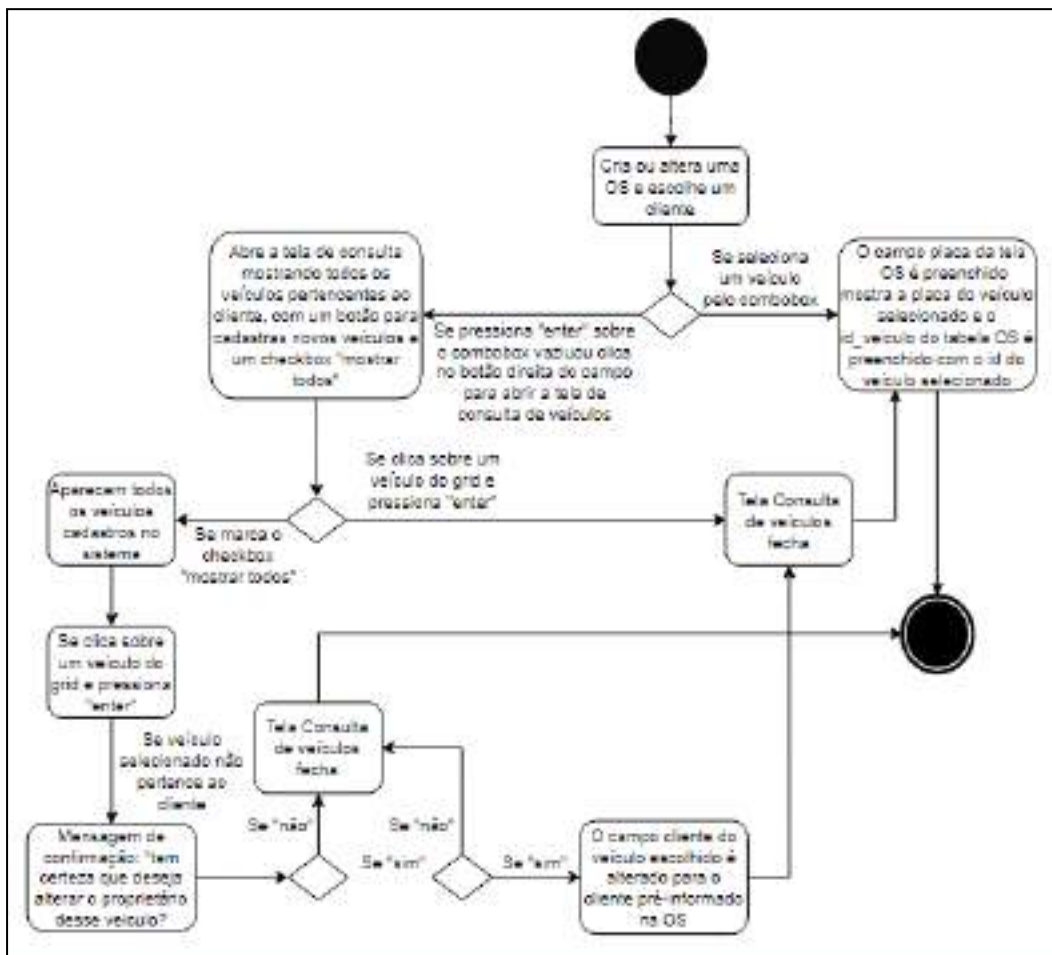
confirmação para mudar o proprietário do veículo) e executa a função placa para atualizar a consulta de veículo da tela OS para incluir o novo veículo que não pertencia ao cliente.

- Os demais códigos da tela seguem os padrões para telas de consulta explicadas no tópico Padrões de Código.

Segue abaixo um diagrama de atividades demonstrando o processo de definição do veículo que está em manutenção:

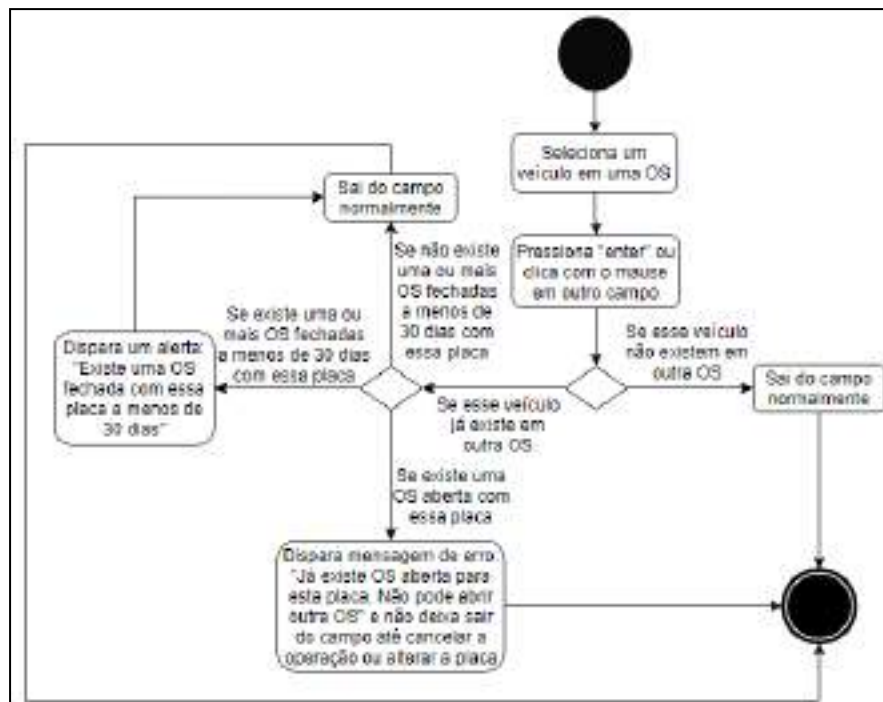
### Diagrama Atividades dos veículos da OS

Figura 1099: Informando um veículo em uma OS



Fonte: Produzida pelo autor

Figura 110: Validando um Veículo em uma OS



Fonte: Produzida pelo autor

Figura 111: Tela Cadastro de Técnicos

The screenshot shows a web application window titled 'Cadastro de Técnicos'. The form is for creating a new technician record. It includes the following fields and values:

- Código:** 3
- Nome:** ALEXANDRE SILVA
- Dt. Admissão:** 23/03/2009
- Dep:** 00001-000
- Endereço:** R. SERRA LARROSA
- BR:** 1
- Complemento:**
- Sistema:** CENTRAL
- Grupos:** INDIÚ ALGAR
- Id:** 89
- CPF:** 000.000.000-00
- Tel. Celular:**
- Telefone:** ( ) -
- Celular:** ( ) -
- e-Mail:**
- Assinatura:**
- Afiliado:** 000
- Data Cadastro:** 09/04/2016

At the bottom of the window, there is a toolbar with icons for 'Início', 'Atualizar', 'Excluir', 'Imprimir', 'Cancelar', 'Novo', 'Salvar', 'Excluir', 'Ajuda', and 'Fechar'.

Fonte: Produzida pelo autor

### 3.30 Características e funcionalidades básicas da tela Cadastro de Técnicos

A função da tela Cadastro de Técnicos é fazer o CRUD dos técnicos que, por sua vez, podem ser relacionados com itens de serviço.

A aba “ficha” possui duas abas dentro dela, uma para cadastrar todas as demais informações pertinentes ao técnico e outra para cadastrar uma observação sobre o técnico.

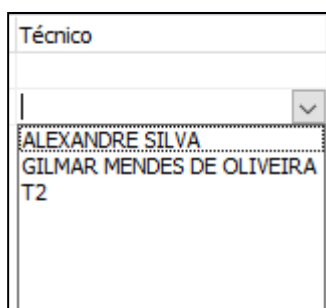
Em uma OS, são os técnicos que vão ser responsáveis por cada item de serviço, portanto, na tela de OS, é possível informar em cada item de serviço qual técnico realizara o serviço.

A tela de Cadastro de Técnicos é uma tela ligada a tabela do banco “tecnicos” e segue todos os padrões previamente mencionados, nos tópicos Padrões do Sistema e Padrões de Desenvolvimento para telas de CRUDS, portanto não será detalhado novamente.

### 3.31 Escolhendo um técnico para um item de serviço em uma OS

- ✓ O campo “Técnico” não recebe nada quando um item de serviço é criado
- ✓ Ele é alterável, ao clicar sobre o campo ele abre um dropdown mostrando o nome de todos os técnicos ativos cadastrados no sistema
- ✓ Ao clicar sobre um deles, o campo “Técnico” será preenchido;
- ✓ Abaixo a sequencia de passos para informar o técnico:

Figura 112: Dropdown do campo Técnico do grid de itens de serviços



Fonte: Produzida pelo autor

Figura 113: Técnico selecionado

Técnico
ALEXANDRE SILVA

Fonte: Produzida pelo autor

- Não é permitido digitar nem apagar textos dentro do campo;
- Configurações do sistema que permitiram esse efeito:
- Configuração do campo Técnico no clientDataSet para fazer mostrar registros da tabela técnico.

Figura 114: Configuração do campo técnico no clientDataSet

FieldKind	fkLookup
FieldName	tecnico
KeyFields	id_tecnico
LookupDataSet	dmgeral.qryTecnicos
LookupKeyFields	CODIGO
LookupResultField	NOME

Fonte: Produzida pelo autor, as imagens

acima mostram os campos que precisam ser configurados no clientDataSet para permitir a exibição dos registros de outra consulta em uma célula de um grid que funcione de ligação e seleção de permitindo também a seleção de um desses registros.

- Agora já é possível ver os técnicos filtrados pela qryTecnicos (apenas técnicos ativos) ao clicar 5 vezes sobre o campo Técnico do grid.

Para configurar o campo para listar os técnicos em 1 click, foi utilizado o evento onCellClick:

Figura 115: Verificação do estado da tela no evento de click no grid

```
if qryOS.State in [dsInsert,dsEdit] then
begin
```

Fonte: Produzida pelo autor

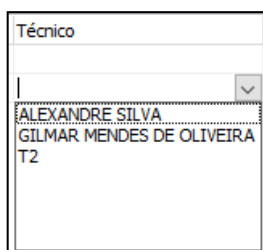


Figura 116: Atalho para ativação do dropdown

```
if DBGrdl.Columns.Grid.SelectedField.FieldName = 'tecnico' then
begin
  keybd_event (VK_F2, 0, 0, 0);
  keybd_event (VK_F2, 0, KEYEVENTF_KEYUP, 0);
  keybd_event (VK_MENU, 0, 0, 0);
  keybd_event (VK_DOWN, 0, 0, 0);
  keybd_event (VK_DOWN, 0, KEYEVENTF_KEYUP, 0);
  keybd_event (VK_MENU, 0, KEYEVENTF_KEYUP, 0);
end;
```

Fonte: Produzida pelo autor, o código acima verifica se o campo do grid clicado foi o técnico, se sim cria os atalhos para abrir o dropdown imediatamente

Figura 115: Dropdown do campo técnico aberto em 1 click



Fonte: Produzida pelo autor

Figura 116: Comissões de Serviços

Manutenção de Tabela Auxiliar - Comissão de Serviço

Tela: Grupos

Código: 1    Descrição: COMISSÃO 1    Emissão Mínima: 10,00 %    Data Cadastro: 27/04/2020

Faixas de Faturamento

Ord	Ato	DESCONTO1	COMISSAO1	DESCONTO2	COMISSAO2	DESCONTO3	COMISSAO3
1	100,00	100,00	5,00 %	2,00 %	3,00 %	1,00 %	2,00 %

Novo    Modificar    Excluir

Novo    Modificar    Excluir    Inserir    Remover    Atualizar    Voltar    Avançar    F12    F10

Fonte: Produzida pelo autor

Como foi desenvolvida apenas a tela e não foi ainda integrada com o resto do sistema, ela não será detalhada.

Figura 117: Faixas de Faturamento de uma comissão de Serviços

Manutenção de Tabela Auxiliar - Faixas de Faturamento

Tela: Grupos

Valor 1: 100,00    Valor 2: 100,00

Desconto 1: 5,00 %    Comissão 1: 2,00 %

Desconto 2: 2,00 %    Comissão 2: 3,00 %

Desconto 3: 0,50 %    Comissão 3: 1,00 %

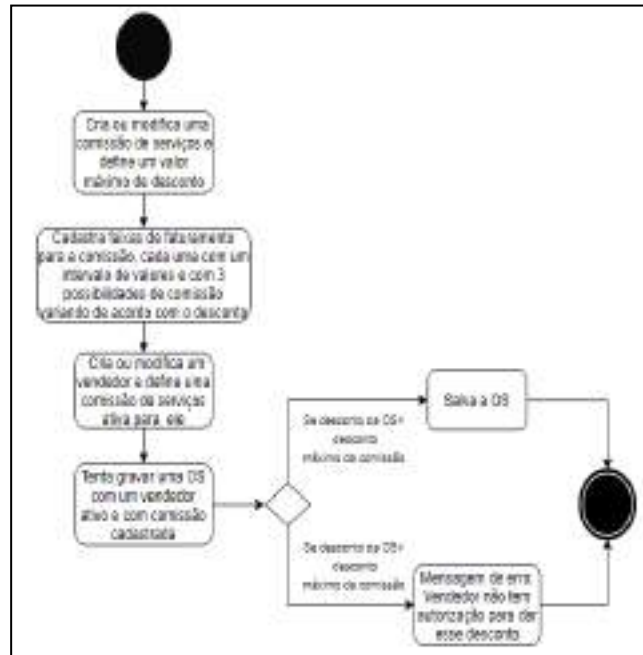
Gravar    Cancelar    Voltar    Avançar    F12    F10

Fonte: Produzida pelo autor

Como a parte de comissão de serviços não foi integrada com o resto do sistema, essa tela não será detalhada.

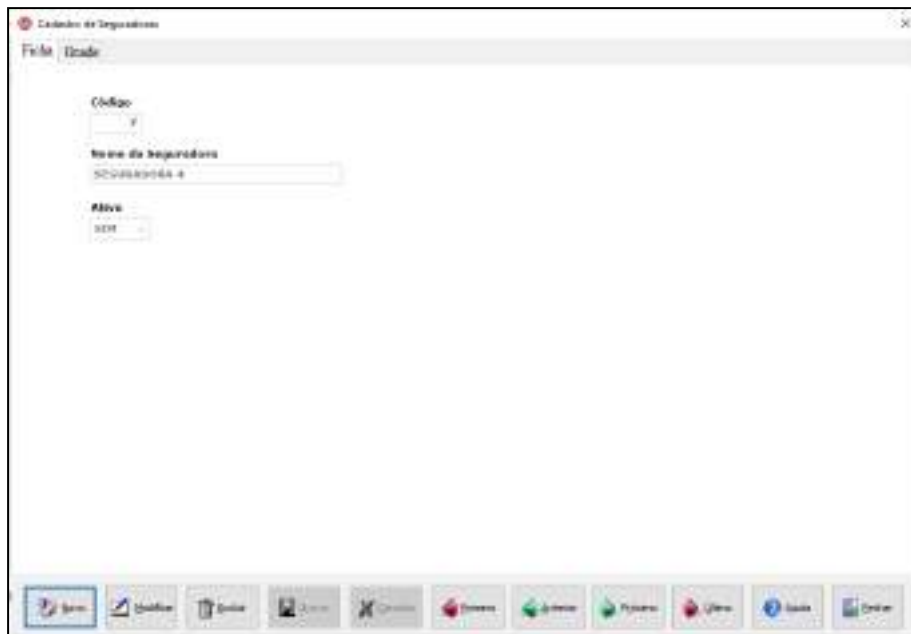
Será apenas demonstrado via diagrama de atividades como será essa integração.

Figura 118: Diagrama de Atividades da Tela Cadastro de Comissões de Serviços



Fonte: Produzida pelo autor

Figura 119: Tela Cadastro de Seguradoras



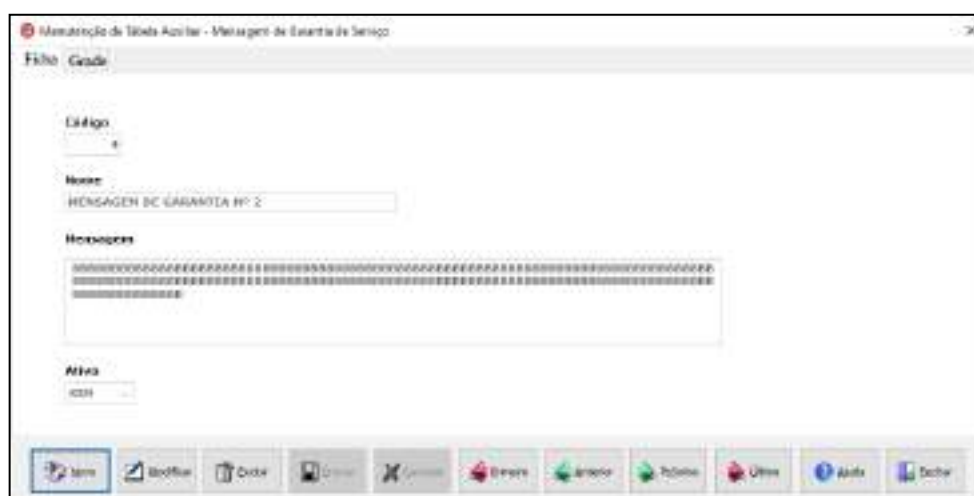
Fonte: Produzida pelo autor

### 3.32 Funcionalidades e características básicas da tela Cadastro de Seguradoras

- Seu objetivo é ler, cadastrar, modificar e excluir seguradoras.

- Essas seguradoras vão ser usadas, quando necessário, para informar a seguradora que vai cobrir os custos do concerto do veículo.
- É uma tela ligada diretamente a tabela “seguradoras” do banco de dados do sistema.
- Como essa tela segue todos os padrões de layout, regras de negócio e padrões de desenvolvimento comentados nos tópicos Padrões do Sistema e Padrões de desenvolvimento, ela não será detalhada.

Figura 120: Tela Mensagem de Garantia de Serviço



Fonte: Produzida pelo autor

### 3.33 Características e Funcionalidades da Tela Cadastro de Mensagem da OS

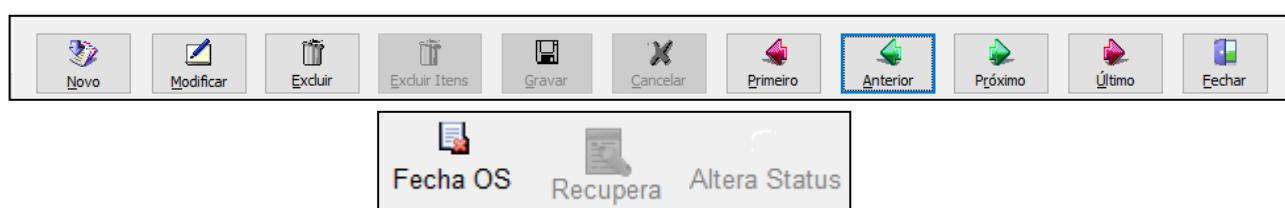
- A função da tela Cadastro de Mensagens da OS é fazer o CRUD das mensagens da OS que, por sua vez, podem ser relacionados com uma OS.
- A tela de Cadastro de Mensagens da OS é uma tela ligada a tabela do banco “msggarantias” e segue todos os padrões previamente mencionados, nos tópicos Padrões do Sistema e Padrões de Desenvolvimento para telas de CRUDS, portanto não será detalhada novamente.

### 3.34 Status e Situação da OS

#### 3.34.1 Características básicas da situação e status da OS

- ✓ Uma OS pode ter duas situações:
- ✓ Aberta: É possível modificar e excluir a OS;
- ✓ Botões de uma OS aberta no modo de visualização

Figura 121: Botões de uma OS aberta no modo de visualização



Fonte: Produzida pelo autor

Pode ter os seguintes status:

- ✓ “ORCAMENTO”, “AG APROVACAO”, “AG PARADA”, “BOX”, “EM SERVICIO”, “ENTREGUE”, “GARANTIA”, “LAVAGEM” e “NAO AUTORIZADO”.

Na inclusão de uma OS é possível adicionar apenas os status:

- ✓ “AG APROVACAO”, “EM SERVICIO”, “GARANTIA” e “ORCAMENTO”.
- ✓ Status disponíveis quando uma OS está sendo criada

Figura 122: Status disponíveis na criação de uma OS

The screenshot shows a form with two main sections: 'Status' and 'Situacao'. The 'Status' section has a dropdown menu currently open, showing the following options: 'ORCAMENTO' (selected), 'AG APROVACAO', 'GARANTIA', and 'EM SERVICO'. The 'Situacao' section has a text input field containing the word 'ABERTA'. Below the status dropdown, there are labels for 'Modelo' and 'Kilometros' with corresponding input fields.

Fonte: Produzida pelo autor

Na modificação de uma OS é possível alterar o status para:

- ✓ “ORCAMENTO”, “AG APROVACAO”, “AG PARADA”, “BOX”, “EM SERVICO”, “ENTREGUE”, “GARANTIA”, “LAVAGEM” e “NAO AUTORIZADO”.
- ✓ Status disponíveis quando uma OS está sendo modificada

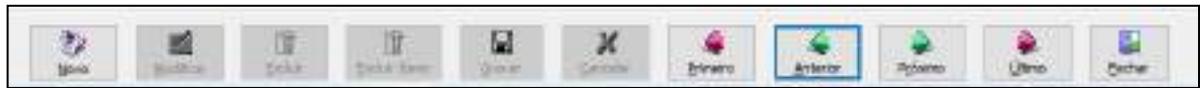
Figura 123: Status disponíveis na modificação de uma OS

The screenshot shows a form with two main sections: 'Status' and 'Situacao'. The 'Status' section has a dropdown menu currently open, showing the following options: 'AG APROVACAO', 'NAO AUTORIZADO', 'AG PARADA', 'LAVAGEM', 'BOX', 'GARANTIA', and 'EM SERVICO'. The 'Situacao' section has a text input field containing the word 'ABERTA'. Below the status dropdown, there are labels for 'Modelo', 'Kilometros', and 'mensagem de Garantia de OS' with corresponding input fields.

Fonte: Produzida pelo autor

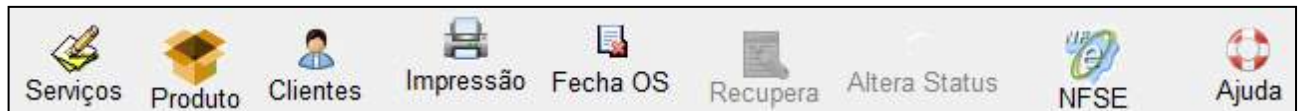
- Fechadas: Não é possível modificar nem excluir a OS;
- Botões de uma OS Fechada no modo de visualização.

Figura 1246: Botões da Parte Inferior



Fonte: Produzida pelo autor

Figura 1257: Botões da Parte Superior

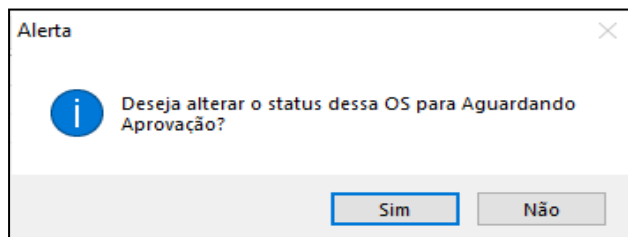


Fonte: Produzida pelo autor

Pode ter os seguintes status:

- ✓ “AG RETIRADA” e “ENTREGUE”;
- ✓ Ao fechar ela recebe o status “ENTREGUE”;
- ✓ O status pode ser alternado entre “AG RETIRADA” e “ENTREGUE” clicando no botão “Altera Status”, que só habilitado para OS com status “FECHADA” e é a única alteração que se pode fazer em uma OS fechada.
- ✓ Mensagem de confirmação disparada ao clicar no botão “Altera Status” quando o status atual é “Entregue” Conforme a figura abaixo.

Figura 1267: Mensagem de confirmação para alteração do status de uma OS fechada



Fonte: Produzida pelo autor

Figura 1278: Status Alterado

Status	Situação
AG RETIRADA	FECHADA

Fonte: Produzida pelo autor

O status pode ser alterado novamente como mostra a figura abaixo:

Figura 1289: Tela de Status

Status	Situação
ENTREGUE <input type="button" value="v"/>	<input type="button" value="FECHADA"/>

Fonte: Produzida pelo autor

Para fechar uma OS o usuário tem que ter uma permissão específica, clicar no botão “Fecha OS” e confirmar a operação. Como é uma operação delicada, a data e hora da operação, nome da operação e usuário que realizou a operação ficam salvos na tabela “log”.

Código do botão “Fecha OS”:

Verifica se o usuário tem permissão para fechar uma OS, caso contrário mostra uma. Conforme mostram as figuras abaixo:

Figura 12930: Código que verifica permissão para fechar uma OS

```
procedure TfrmOS.btnFechaClick(Sender: TObject);
var r :Byte;
begin
DMGeral.qryPermissao.locate('programa','A0200',[]);
if DMGeral.qryPermissao.FieldByName('acesso').Value <> 'SIM' then
begin
ShowMessage('Acesso negado para seu usuário!');
exit;
end;
end;
```

Fonte: Produzida pelo autor

Figura 13031: Código que dispara mensagem de confirmação para fechar a OS

```
r := Application.MessageBox('Deseja realmente fechar essa OS?','Alerta',MB_yesno+MB_iconInformation);
if r = 6 then
begin
```

Fonte: Produzida pelo autor

Altera a situação da OS para fechada, filtra novamente os status da tabela qryos\_status com a função atualiza\_status e altera o status da OS para “ENTREGUE” e faz data de fechamento, que era nula, receber a data.



Figura 13132: Código que altera a situação para fechada e o status para entregue

```

qryOS.Edit;
qryOS.FieldByName('situacao').value:='FECHADA';
atualiza_status();
qryOS.FieldByName('id_os_status').value:=25;
qryOS.FieldByName('data_fechamento').asdatetime:=Now;

```

Fonte: Produzida pelo autor

Figura 13233: Código que desabilita os botões depois do fechamento

```

btnModificar.Enabled:=False;
btnExcluir.Enabled:=False;
btnRecupera.Enabled:=true;
btnFecha.Enabled:=false;
btnAlterar_status.Enabled:=True;
qryOS.Post;

```

Fonte: Produzida pelo autor

Cria um registro da operação na tabela “log” do banco de dados do sistema. O campo “tipo” recebe a string “ALERTA”, o campo “tabela” recebe “OS”, o campo “descricao” recebe “OS x “FECHADA” e o campo “motivo” recebe “FECHAMENTO DE OS”. As.

Figura 13334: Código que registra o fechamento na tabela log

```

DMExtra.qrylog.close;
DMExtra.qrylog.sql.Clear;
DMExtra.qrylog.SQL.Add('select * from logs where l=0');
DMExtra.qrylog.Open;

DMExtra.qrylog.Append;
DMExtra.qryLogdata.Value := pega_data;
DMExtra.qryLoghora.Value := pega_hora;
DMExtra.qryLogusuario.AsString := DMGeral.qryLoginNOME.AsString;
DMExtra.qryLogtipo.Value := 'ALERTA';
DMExtra.qryLogtabela.Value := 'OS';
DMExtra.qryLogdescricao.Value := 'OS ' + intostr(qryos.FieldByName('id_os').value) + ' FECHADA';
DMExtra.qryLogMOTIVO.AsString := 'FECHAMENTO DE OS';
DMExtra.qrylog.Post;
DMExtra.qrylog.Refresh;
Exit;

```

Fonte: Produzida pelo autor

Quando uma OS é fechada, caso a modalidade de pagamento seja a prazo, serão gerados títulos com o nome do cliente em contas a receber, mas essa parte foi desenvolvida por outro programador portanto não será detalhada.

Para abrir uma OS fechada o usuário tem que ter uma permissão específica, clicar no botão “Recupera” e confirmar a operação. Como é uma operação delicada,

a data e hora da operação, nome da operação e usuário que realizou a operação ficam salvos na tabela “log”.

Código do botão “Recupera”:

Verifica se o usuário tem permissão para recuperar uma OS, caso contrário mostra uma mensagem de erro e aborta a operação.

Figura 13435: Código que verifica a permissão para recuperar uma OS fechada

```

procedure TfirmOS.btnRecuperaClick(Sender: TObject);
var r :Byte;
begin
  DMGeral.qryPermissao.locate('programa','A0100',[]);
  if DMGeral.qryPermissao.FieldByName('MASTER').Value <> 'SIM' then
  begin
    ShowMessage('Acesso negado para seu usuário!');
    exit;
  end;
end;

```

Fonte: Produzida pelo autor

Verifica se o usuário confirmou a operação. Se confirmou é verificado se já existe uma OS aberta com o mesmo veículo. Caso exista mostra mensagem de erro e aborta a operação.

Figura 13536: Código que verifica se já existe uma OS aberta com o mesmo veículo

```

r := Application.MessageBox('Deseja realmente abrir essa OS?', 'Alerta', MB_yesno+MB_iconInformation);
DMExtra.qryConsultas.Close;
DMExtra.qryConsultas.SQL.Clear;
DMExtra.qryConsultas.SQL.Add('select * from os where id_veiculo=:id_veiculo and situacao=:situacao');
DMExtra.qryConsultas.parambyname('id_veiculo').value:=qryOS.FieldByName('id_veiculo').value;
DMExtra.qryConsultas.parambyname('situacao').value:='ABERTA';
DMExtra.qryConsultas.Open;
if r = 6 then
begin
  if not DMExtra.qryConsultas.IsEmpty then
  begin
    alerta('Erro', 'Já existe OS aberta para esta placa. Não pode abrir outra OS');
  end
end

```

Fonte: Produzida pelo autor

Abre a OS, atualiza a qryos\_status, muda o status da OS para “EM SERVIÇO” e faz a data de fechamento ficar nula.

Figura 13637: Código que atualiza qryos\_status

```

qryOS.Edit;
qryOS.FieldName('situacao').value:='ABERTA';
atualiza_status();
qryOS.FieldName('id_os_status').asinteger:=20;
dbedtData_fechamento.Clear;
btnModificar.Enabled:=true;
btnExcluir.Enabled:=true;
btnRecupera.Enabled:=false;
btnFecha.Enabled:=true;
btnAltera_status.Enabled:=false;
qryOS.Post;
qryOS.Refresh;

```

Fonte: Produzida pelo autor

Caso a OS tenha sido aberta com sucesso, a operação é salva na tabela “log” do banco de dados do sistema. O campo “tipo” recebe a string “ALERTA”, o campo “tabela” recebe “OS”, o campo “descrição” recebe “OS x “RECUPERADA” e o campo “motivo” recebe “RECUPERAÇÃO DE OS”. As alterações na tabela “log” são salvas e a operação é finalizada. Conforme a figura abaixo:

Figura 13738: Código que registra na tabela log a recuperação da OS e quem recuperou

```

DMExtra.qrylog.close;
DMExtra.qrylog.sql.Clear;
DMExtra.qrylog.SQL.Add('select * from logs where l=0');
DMExtra.qrylog.Open;

DMExtra.qrylog.Append;
DMExtra.qryLogdata.Value := pega_data;
DMExtra.qryLoghora.Value := pega_hora;
DMExtra.qryLogusuario.AsString := DMGeral.qryLoginNOME.AsString;
DMExtra.qryLogtipo.Value := 'ALERTA';
DMExtra.qryLogtabela.Value := 'OS';
DMExtra.qryLogdescricao.Value := 'OS ' + inttostr(qryos.FieldName('id_os').value) + ' RECUPERADA';
DMExtra.qryLogMOTIVO.AsString := 'RECUPERAÇÃO DE OS';
DMExtra.qrylog.Post;
DMExtra.qrylog.Refresh;

```

Fonte: Produzida pelo autor

Foi criada apenas uma tabela no banco para os status da OS, “os\_status”, mas não uma tela para fazer os CRUDS dela dos mesmos. O motivo se deve aos status da OS serem um padrão do sistema e existe inclusive algumas validações em cima dos status.

A “os\_status” é importada nos bancos dos clientes de forma padronizada, mas os clientes podem solicitar a retirada de alguns e é feita a operação.

### 3.35 Integração com a parte de Peças

A parte de peças já existe no sistema e, atualmente, dois membros da equipe da empresa MBA Sistemas estão fazendo alterações nessa parte do sistema. A integração completa com a parte de peças depende da conclusão dessas alterações, portanto foi adiada.

Apesar dessa parte de peças ainda não ter sido integrada, já foram criados relatórios envolvendo com peças, com uma ligação via banco.

#### 3.35.1 Integração com a Parte de Contas a Receber

Essa integração, com a parte de contas a receber, foi realizada por outro programador, portanto será abordada superficialmente.

No fechamento da OS, caso a modalidade de pagamento seja a prazo, são gerados títulos em contas a receber com datas de vencimento para cada parcela. A operação é bloqueada caso o valor do serviço ultrapasse o limite de crédito do cliente.

Figura 13838-A: Código de verificação de limite de Credito do Cliente

```

DMGgeral.qryPrazo_Faturamento.Locate('codigo', cdsPedidosCODIGO_COND_PAG.AsInteger, []);
if DMGgeral.qryPrazo_Faturamento.FieldByName('PARCELAS').Value <> 0 then
begin
DMGgeral.qryClientes_Valor_Credito.Close;
DMGgeral.qryClientes_Valor_Credito.SQL.Clear;
DMGgeral.qryClientes_Valor_Credito.SQL.Add('select * from clientes_valor_credito where id_cliente= :id');
DMGgeral.qryClientes_Valor_Credito.ParamByName('id').Value := cdsPedidosCODIGO_CLIENTE.Value;
DMGgeral.qryClientes_Valor_Credito.Open;
//verifica inatividade do cliente
if DMGgeral.qryClientes_Valor_Credito.PRAZO_INATIVIDADE.Value <> 0 then
begin
IMEXtra.qryConsultas.Close;
IMEXtra.qryConsultas.SQL.Clear;
IMEXtra.qryConsultas.SQL.Add('select * from os where codigo_cliente= :codigo order by data_pedido);
IMEXtra.qryConsultas.SQL.Add(' desc limit 1');
IMEXtra.qryConsultas.ParamByName('codigo').Value := cdsPedidosCODIGO_CLIENTE.Value;
IMEXtra.qryConsultas.Open;
if not IMEXtra.qryConsultas.IsEmpty then
begin
if DaysBetween(pega_data, JvDBDatePickerEdit2.Date) >
DMGgeral.qryClientes_Valor_Credito.PRAZO_INATIVIDADE.AsInteger then
begin
alerta('Erro', 'Cliente com cadastro defasado');
cb1kbbCODIGO_VENDEDOR.SetFocus;
EXIT;
end;
end;
end;
end;

```

Fonte: Produzida pelo autor

Figura 13938-B: Código onde são Gerados os títulos

```

while c <= frmOS.dsPrazo_Fat.DataSet.FieldByName('PARCELAS').Value do
begin
    DMGeral.qryClientes_Valor_Credito.Locate('id_cliente', frmOS.cdsOScodigo_cliente.asinteger, []);
    contas_receber := Superstring('0', 9 - Length(frmTela_Atendimento.cdsPedidosNUM_PEDIDO.AsString))
    + frmOS.cdsOSNUM_PEDIDO.AsString + '-' + FormatFloat('000', c) + '/';
    + FormatFloat('000', DMGeral.qryPrazo_FaturamentoPARCELAS.AsInteger),
    frmOS.cdsPedidoscodigo_cliente.asinteger,
    frmOS.cdsPedidosNUM_PEDIDO.Value,
    data,
    valor + dif,
    DMGeral.qryClientes_Valor_CreditoLOCALIZACAO.Value,
    DMGeral.qryClientes_Valor_Creditoid_operacao_cobranca.Value,
    DMGeral.qryEmpresaTIPO_TITULO_DEFAULT.Value);

    dif := 0;
    c := c + 1;
    data := IncDay(data, DMGeral.qryPrazo_FaturamentoDIASDENAISPARC.AsInteger);
end;

```

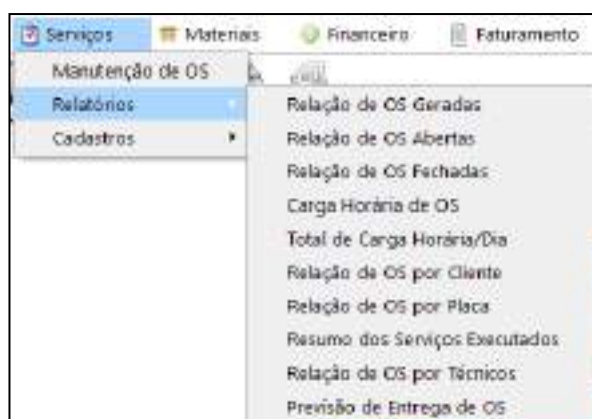
Fonte: Produzida pelo autor

### 3.36 Relatórios de Módulo de Ordens de Serviço

#### 3.36.1 Abertura dos Relatórios da Ordem de Serviços

- ✓ Os relatórios do módulo de OS precisam de permissão para serem abertos.
- ✓ Eles são acessados apenas pelo menu principal do sistema, no módulo de Ordens de Serviço.
- ✓ Menu onde estão os links para todos relatórios da parte de OS do sistema.

Figura 14039: Menu do módulo de OS



Fonte: Produzida pelo autor

Figura 14140: Tela Codigo para abetura de um Relatorio do modulo de ordens de serviços

```

procedure TfrmPrincipal.btn2Click(Sender: TObject);
begin
if DMGeral.qryPermissao.FieldByName('acesso').Value <> 'SIM' then
begin
  ShowMessage('Acesso negado para seu usuário!');
  exit;
end;

if frmRel_os_horas = nil then
  Application.CreateForm(TfrmRel_os_horas, frmRel_os_horas);
  frmRel_os_horas.ShowModal;
end;

```

Fonte: Produzida pelo autor

- Verificado se o usuário tem permissão, para acessar o relatório, se sim cria o relatório e mostra ele

### 3.37 Padrões de Codificação dos Relatórios

- ✓ A ferramenta de geração de relatórios utilizada é TrfxReport.
- ✓ Todos os campos, sejam de identificação ou campos que mostram um valor do banco, na ferramenta TrfxReport, são do tipo Memo.
- ✓ Os Memos, na ferramenta TrfxReport, são agrupados dentro de componentes de grupo que, por sua vez tem regras específicas.
- ✓ O primeiro componente de grupo utilizado na ferramenta TrfxReport é sempre o PageHeader, que aparece a cada nova página do relatório e é utilizado com o mesmo layout para todos os relatórios.

Figura 14241: Page Header

PageHeader: PageHeader1		Página:	IPAGE#
Empresa:		Data:	IDATE
SAV - SISTEMA DE ADMINISTRAÇÃO DE LOJAS		Hora:	ITIME

Fonte: Produzida pelo autor

Figura 14342: Código do Page Header

```
(frxrprt1.FindObject('memo4') as TfrxMemoView).Text := UpperCase(DMGeral.qryempresafantasia.value);
(frxrprt1.FindObject('memo20') as TfrxMemoView).Text := 'Relação de OS Geradas      Período: '+dtp1.text
+' a '+dtp2.text;
```

Fonte: Produzida pelo autor

O próximo componente de grupo utilizado na ferramenta TrfxReport é o MasterData, que é utilizado para mostrar os dados de nível mais alto do relatório. Ele recebe em seu dataSet, portanto, a consulta de nível mais alto.

Ex: Caso o relatório imprima os dados principais de uma OS e abaixo todos os itens dessa OS, no MasterData vai constar os dados da OS e os itens vão aparecer em detailData que vai ser dependente do MasterData.

O próximo componente de grupo utilizado na ferramenta TrfxReport (caso o relatório precise) é o detailData, que é dependente do MasterData e imprime dados relacionados de alguma forma a um registro impresso do MasterData a cada impressão do mesmo. Podem ter vários componentes detailData em um relatório.

O próximo componente de grupo utilizado na ferramenta TrfxReport (caso o relatório precise) é o subDetailData, que é dependente do detailData e imprime dados relacionados de alguma forma a um registro impresso do detailData a cada impressão do mesmo. Podem ter vários componentes subDetailData em um relatório, mas não existe nenhum caso no módulo de ordens de serviço.

O último componente de grupo utilizado na ferramenta TrfxReport é o ReportSummary. Nele são realizadas as totalizações de alguns valores pré-definidos somando esses valores de todas as instâncias em que eles aparecem.

- ✓ Consultas para os componentes de grupo da ferramenta TrfxReport:
- ✓ Consultas que vão ser usadas em um componente MasterData com filhos do TrfxReport:
- ✓ É criada uma query(já foi explicada no tópico padrões de código) com a consulta;
- ✓ É criado um componente TrfxDataSet que recebe a query e vai abastecer o MasterData no TrfxReport;
- ✓ É criado um dataSource que recebe a query e vai servir como parâmetro para as queries “filhas”.
- ✓ Consultas que vão ser usadas em um componente detailData com filhos do TrfxReport:

- ✓ É criada uma query com a consulta;
- ✓ Na propriedade MasterField da query é colocado o campo, da query com a consulta utilizada no MasterData do TrfxReport, que vai fazer a ligação com o a consulta que abastece o MasterData;
- ✓ Na propriedade MasterSource da query é colocado o dataSource, que tem a consulta utilizada no MasterData do TrfxReport;
- ✓ É criado um componente TrfxDataSet que recebe a query e vai abastecer o detailData no TrfxReport;
- ✓ É criado um dataSource que recebe a query e vai servir como parâmetro para as queries “filhas”.
- ✓ Consultas que vão ser usadas em um componente subDetailData do TrfxReport:

É criada uma query com a consulta;

Na propriedade MasterField da query é colocado o campo, da query com a consulta utilizada no MasterData do TrfxReport, que vai fazer a ligação com a consulta que abastece o detailData;

Na propriedade MasterSource da query é colocado o dataSource, que tem a consulta utilizada no detailData do TrfxReport;

É criado um componente TrfxDataSet que recebe a query e vai abastecer o subDetailData no TrfxReport.

Os eventos de cada item de serviço poderiam ser apresentados em um subDetailData, mas dessa forma ficaria um espaço em branco nos campos onde ficam os eventos e eles iam aparecer só na linha de baixo, portanto foi criado uma consulta que filtra os itens de serviço e os eventos de cada item de serviço, só que quando o evento é diferente de nulo ou de “INICIO”, os campos da tabela item de serviço não são impressos, criando esse efeito:



Figura 14441-A: Itens e eventos da OS

Itens da OS									
IT	Descrição	Lado	Técnico	Qty	Pr Unit	Evento	Data	Hora	Status
1	PINTURA PORTA DIANTEIRA ESQUERDA		GILMAR MENDES	1	200,00	INICIO	13/03/2020	17:58	APROVADO
						FINALIZADO	13/03/2020	18:30	
2	CHAPEACAO PORTA DIREITA		GILMAR MENDES	1	350,00				APROVADO
3	TROCA DO FREIO			1	500,00	INICIO	17/03/2020	12:33	APROVADO
						PARADA	17/03/2020	12:33	
						REINICIO	17/03/2020	12:33	
4	REVISAO COMPLETA		T2	1	100,00				APROVADO

Fonte: Produzida pelo autor

Os filtros dos relatórios são, quase sempre, aplicados apenas sobre a consulta utilizada apenas para o MasterData do TrfxReport, uma vez que, dessa forma, como as outras consultas são “filhas”, acabam sendo filtradas de acordos com os registros da consulta “pai”. A exceção é quando o campo filtrado é impresso na tabela filho. Exemplo da exceção:

Se existe um filtro por serviço, são filtrados no MasterData, todas as OS que possuem esse um item de serviços ligado a esse serviço e, no detailData, apenas os itens de serviço de cada uma dessas OS que tem ligação com esse serviço.

Figura 14541-A: Exemplo do Código de Aplicação dos Filtros em uma Consulta

```

if xbl.Checked then
begin
  consulta:='where os.data_os>='+quotedstr(dtp1.Text)+' and os.data_os<='+quotedstr(dtp2.Text);
end;
else
begin
  consulta:='where os.data_fechamento>='+quotedstr(dtp1.Text)+' and os.data_fechamento<='+quotedstr(dtp2.Text);
end;
if cbb4.ItemIndex<>0 then
begin
  if consulta='' then
  begin
    consulta:='where ';
  end;
  else
  begin
    consulta:=consulta+' and ';
  end;
  consulta:=consulta+'os_status,descricao'+quotedstr(cbb4.Text);
end;

```

Fonte: Produzida pelo autor

### 3.37.1 Relatórios de OS Geradas

Figura 14643:Relatório de OS Geradas

A imagem mostra a interface de usuário para gerar um relatório de OS. A janela é intitulada 'Relatório de OS Geradas'. No topo, há um campo 'Período de Apuração' com dois campos de data: '01/07/2020' e '11/07/2020', separados por 'até'. Abaixo, há dois radiobuttons: 'Abertura OS' (selecionado) e 'Fechamento OS'. Segue um menu suspenso 'Status da OS' com 'ORÇAMENTO' selecionado. Abaixo, há um menu suspenso 'Tipo do Layout' com 'ANALITICO' selecionado. Em seguida, há dois menus suspensos: 'Materiais de Consumo' com 'SM' selecionado e 'Imp. Eventos' com 'SM' selecionado. Abaixo, há dois campos de texto: 'Cliente' e 'Consultor', ambos com ícones de lupa à direita. Na base, há dois botões: 'Visualizar' e 'Cancelar'.

Fonte: Produzida pelo autor

Filtros do relatório:

Período de apuração:

- Os radiobuttons indicam se as datas filtradas vão ser referentes ao campo “data\_os” (radiobutton da esquerda) ou “data\_fechamento” (radiobutton da direita) da tabela “OS” do banco de dados do sistema;
- A data da esquerda filtra por data mínima e recebe na abertura da tela a data referente ao dia 1 do mês atual;
- A data da direita filtra por data máxima e recebe na abertura da tela a data do dia atual;

Filtro Status da OS:

- Permite filtrar por qualquer status que uma OS pode ter;
- Quando o radiobutton “Abertura OS” está marcado, só é possível filtrar pelos status que uma OS aberta pode possuir: “ORÇAMENTO”, “AG APROVACAO”, “AG PARADA”, “BOX”, “EM SERVICO”, “ENTREGUE”, “GARANTIA”, “LAVAGEM” e “NAO AUTORIZADO”;
- Quando o radiobutton “Fechamento OS” está marcado, só é possível filtrar pelos status que uma OS fechada pode possuir: “ENTREGUE” e “AG RETIRADA”;

- Quando o valor do filtro é “TODOS”, não é realizado nenhum filtro por status;
- O valor padrão do filtro é “TODOS”.

Filtro “Cliente”:

- Permite filtrar as OS relacionadas apenas a um cliente específico;
- Além de escolher o cliente pelo dropdown do filtro, é possível acessar a consulta de clientes pelo botão a direita dele e escolher um cliente pela tela de consulta;
- Quando nulo, não é realizado nenhum filtro de clientes;
- O valor padrão do filtro é nulo.

Filtro “Consultor”:

- Permite filtrar as OS relacionadas apenas a um vendedor específico;
- Além de escolher o vendedor pelo dropdown do filtro, é possível acessar a consulta de vendedores pelo botão a direita dele e escolher um vendedor pela tela de consulta;
- Quando nulo, não é realizado nenhum filtro de vendedores;
- O valor padrão do filtro é nulo.
- O filtro é identificado como “Consultor”, apesar acessar a tabela de vendedores, porque em uma oficina o termo consultor é mais usado.

O filtro “Materiais de Consumo” identifica se serão ou não imprimidos os materiais de consumo ligados a cada OS, ele só pode receber sim ou não;

O filtro “Imp. Eventos” identifica se serão imprimidos os eventos de cada item de serviço da OS ou não, ele só pode receber sim ou não;

Filtro “Tipo de Layout”:

- Identifica se vai ser imprimido um relatório analítico ou resumido;
- Botão “Visualizar”: Imprime o relatório.
- Botão “Cancelar”: Fecha o relatório.

### 3.37.2 Estrutura do Relatório

É um relatório em 2 níveis.

- 1º nível: Contém as informações da OS;
- 2º nível:

- Contém o laudo técnico, ocorrência e observação (caso existam);
- Contém os itens da OS e seus eventos em uma única consulta, seguindo o padrão de relatório 5.3.5;
- Contém a totalização do valor dos itens de serviço;
- Contém as peças ligadas a cada OS, caso exista;
- Contém a totalização do valor das peças;
- Contém os materiais de consumo utilizados, caso exista e caso o filtro Imprime Materiais de consumo tenha valor “Sim”;

### 3.38 Relatório Analítico de OS Geradas

Para cada página nova, imprime um cabeçalho com o nome da empresa que está executando o sistema, o nome do sistema, o período que está sendo filtrado, o nome do relatório, a página do relatório, a data em que ele está sendo gerado e a hora em que ele está sendo gerado.

Figura 14744: Header do relatório de OS Geradas

GOVONI	Página:	4
SAV - SISTEMA DE ADMINISTRAÇÃO DE LOJAS	Data:	11/07/2020
Relação de OS Geradas    Período: 01/07/2010 a 11/07/2020	Hora:	23:54:48

Fonte: Produzida pelo autor

As primeiras informações em cada OS do período e com o status filtrado (ou não) a serem imprimidas são referentes ao cliente, ao veículo, seguradora, forma de pagamento, desconto, datas, vendedor, nº de controle, nº da OS e status da OS.



Figura 15047: Relatório Analítico Relatório de OS Geradas itens de serviço

Itens da OS									
IT	Descrição	Lado	Técnico	Qty	Pr Unit	Evento	Data	Hora	Status
1	PINTURA DO ARRANHAO DA D PORTA DIANTEIRA		GILMAR MENDES	2	200,00	INICIO	31/03/2020	13:29	APROVADO
2	PINTURA PORTA DIANTEIRA ESQUERDA			1	200,00	INICIO	12/05/2020	12:32	APROVADO
						PARADA	12/05/2020	12:42	
						FINALIZADO	12/05/2020	12:50	
3	LIMPEZA			1	30,00				REJEITADO
Total Liquido Serviços:									480,00

Fonte: Produzida pelo autor

Caso exista alguma (ou várias) peça(s) relacionada(s) a OS, elas são impressas.

Figura 15148: Relatório Analítico Relatório de OS Geradas peças

Pedidos Peças						
IT	Código	Descrição do Produto	Qty	Pr Unit	Desc(%)	Val Total
1	MR1927	ABRACADEIRA MINOR 19X27	1,000	0,77	0,00	0,77
Total Liquido Peças:						0,77
Total Liquido Mat Consumo:						0,00
Total Liquido OS:						3.732,77

Fonte: Produzida pelo autor

Caso exista algum (ou vários) material(s) de consumo relacionado(s) a OS e se o filtro materiais de consumo estiver com o valor SIM, eles são impressos abaixo das

Figura 15249: Relatório Analítico Relatório de OS Geradas materiais de consumo

Fonte: Produzida pelo autor

Pedidos Mat Consumo						
IT	Código	Descrição do Produto	Qty	Pr Unit	Desc(%)	Val Total
1	5912	RET RODA DT TOWNER	1,006	8,15	0,0008	8,20
2	6446	RET VIRAB DT TOPIC/BESTA 2.2	1,080	8,30	0,0000	8,96
Total Liquido Mat Consumo:						17,16

Caso tenha algum material de consumo sendo impresso ou algum, a totalização da OS, que é a soma do valor dos serviços mais a soma do valor das peças, é impressa embaixo do total líquido dos materiais de consumo. Caso contrário vai ser impressa abaixo do total de peças, que, por sua vez, será impresso sempre, mesmo quando não houver peças ligadas a OS, caso em que o total de peças será 0,00.

Figura 15350: Relatório Analítico Relatório de OS Geradas total da OS

Total Liquido OS:	360,00
-------------------	--------

Fonte: Produzida pelo autor

No final do relatório, abaixo de todas as OS, é impressa a soma do total bruto de todas as OS filtradas (serviços+peças), a soma do valor total de desconto sobre os serviços de todas as OS filtradas, a soma do valor total de desconto sobre as peças

relacionadas as OS filtradas e a soma do total líquido de todas as OS filtradas (serviços+peças).

Figura 15451: Relatório Analítico Relatório de OS Geradas totais gerais da OS

Total Bruto:	3.058.092,94
Total Desconto Serviços:	511.254,75
Total Desconto Peças:	0,00
Total Líquido:	2.546.838,19

Fonte: Produzida pelo autor

### 3.39 Relatório Resumido de OS Geradas

Esse relatório mostra apenas o nº da OS, data de abertura, data de fechamento (quando tiver), nome do cliente, placa do veículo, status da OS e valor Líquido.

O cabeçalho é o mesmo do relatório Analítico.

O final do relatório com as totalizações das OS filtradas é também é igual ao do relatório analítico

Figura 15552: Relatório de OS Geradas Relatório Resumido

GOVONI						Página:	1
SAV - SISTEMA DE ADMINISTRAÇÃO DE LOJAS						Data:	12/07/2020
Relação de OS Geradas Período: 01/05/2020 a 12/07/2020						Hora:	01:06:41
OS Nº	Dt Abertura	Dt Fecham.	Nome do Cliente	Placa	Status da OS	Valor Líquido	
167	03/06/2020		A BOLZAN & CIA LTDA	IWF9551	ORCAMENTO	3.732,77	
168	03/06/2020	03/06/2020	2001 COM DE AUTO PECAS LTDA	FDS1122	AG RETIRADA	344,00	
175	15/06/2020		AGLAE T DE ARAUJO ME	TTT2000	EM SERVICO	360,00	
						Total Bruto:	4.612,77
						Total Desconto Serviços:	176,00
						Total Desconto Peças:	0,00
						Total Líquido:	4.436,77

Fonte: Produzida pelo autor

### 3.40 Relatório de OS Abertas

- É quase igual ao relatório de OS Geradas, com duas diferenças:
- Não existe a opção de filtrar o período por data de fechamento, ele é filtrado sempre por data de abertura da OS e não filtra OS fechadas.

Figura 15653: Filtro por data do relatório de OS Abertas

Fonte: Produzida pelo autor

Figura 15754: Relatório OS Abertas Relatório Resumido

OS Nº	Dt Abertura	Nome do Cliente	Placa	Status da OS	Valor Liquido
167	03/06/2020	A BOLZAN & CIA LTDA	IWF9551	ORCAMENTO	3.732,77
175	15/06/2020	AGLAE T DE ARAUJO ME	TTT2000	EM SERVICO	360,00

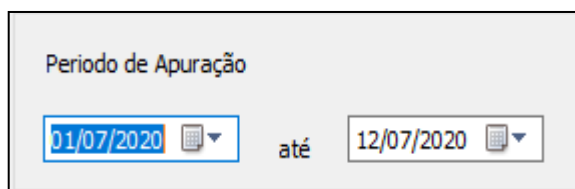
Fonte: Produzida pelo autor

### 3.41 Relatório de OS Fechadas

- É igual ao relatório de OS Geradas em todos os aspectos, com a diferença de que não existe a opção de filtrar o período por data de abertura, ele é filtrado sempre pela data de fechamento, de forma que pega apenas OS fechadas, conforme a figura abaixo:



Figura 15855: Filtro por data do Relatório de OS Fechadas



Período de Apuração

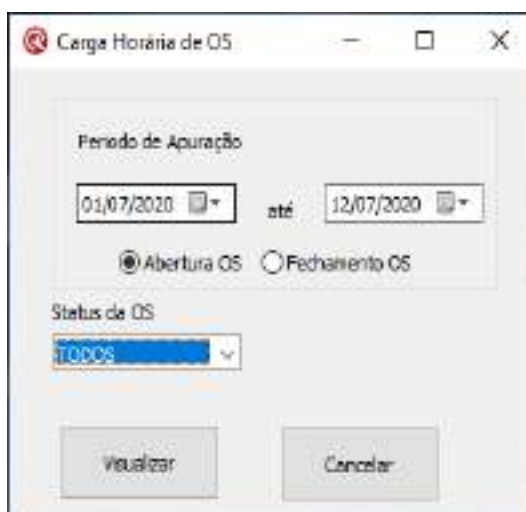
01/07/2020 até 12/07/2020

The image shows a dialog box titled "Período de Apuração" (Reporting Period). It contains two date input fields. The first field is set to "01/07/2020" and the second field is set to "12/07/2020". The word "até" (until) is placed between the two date fields. Both date fields have a small calendar icon to their right, indicating they are date pickers.

Fonte: Produzida pelo autor

### 3.42 Carga Horária de OS

Figura 15956: Relatório de Carga Horária de OS



Carga Horária de OS

Período de Apuração

01/07/2020 até 12/07/2020

Abertura OS  Fechamento OS

Status de OS

TODOS

Visualizar Cancelar

The image shows a dialog box titled "Carga Horária de OS" (OS Hourly Load). It has a standard Windows window title bar with minimize, maximize, and close buttons. The dialog contains a "Período de Apuração" (Reporting Period) section with two date pickers set to "01/07/2020" and "12/07/2020", separated by the word "até". Below this, there are two radio buttons: "Abertura OS" (selected) and "Fechamento OS". Underneath is a "Status de OS" (OS Status) dropdown menu currently showing "TODOS". At the bottom of the dialog are two buttons: "Visualizar" (View) and "Cancelar" (Cancel).

Fonte: Produzida pelo autor

### 3.43 Filtros do Relatório

Os filtros e botões que a tela contém funcionam da mesma forma que os da tela Relatório de OS Geradas.

Ele gera apenas 1 tipo de relatório, aonde consta o tempo total empregado para a realização de todos os serviços de cada OS.

### 3.44 Estrutura do Relatório

É um relatório em 2 níveis.

- 1º nível: Contem as informações da OS;
- 2º nível: Contem o tempo total gasto para realizar os serviços da OS;

### 3.45 Impressão do Relatório

Figura 16057: Impressão do relatório de Carga Horária de OS

Fonte: Produzida pelo autor

OS N <sup>o</sup> : 175		Status da OS: EM SERVICIO	
Nome.....:	AGLAE T DE ARAUJO ME	Código: 966	
Placa.....:	TTT2000    Modelo: F8	Ano: 2010/2010	
Cor.....:	PRETO                      Fabricante: FORD	Km: 0	
Data Abertura:	15/06/202	Previsão de Entrega:	15/06/2020    Data Fechamento:
Seguradora.....:	SEGURADORA 4	N <sup>o</sup> Ctr/Sin:	Consultor: DOUGLAS PEIXOTO
Mod Pag.....:	30 DIAS DEPOIS A PRAZO	Desconto: 20,00 %	
Tempo Total.:	1 horas    45 minutos		

- No final do relatório é impresso a soma dos tempos totais de todas as OS filtradas no relatório

Figura 16158: Impressão do relatório de Carga Horária de OS totalização de horas do período

Tempo Total do Relatório:	147 horas    46 minutos
---------------------------	-------------------------

Fonte: Produzida pelo autor

### 3.46 Consulta do Relatório

A consulta do tempo total funciona da seguinte forma:

Para cada item de serviço de cada OS com evento “FINALIZADO”, é calculado o tempo total em minutos decorrido entre o evento “INICIO” e o evento “FINALIZADO” e é descontado desse total os intervalos de tempo entre os eventos “PARADA” e “REINICIO” (podem ter várias paradas e reinícios para item de serviço) em minutos.

Esse resultado de cada item de serviço é agrupado em uma OS, somando os valores resultantes de todos os itens de serviço dessa OS.

Esse valor resultante é convertido de minutos para horas e minutos no próprio relatório com as seguintes funções:

Função para obter as horas:  
 $[(\text{Trunc}(\langle \text{frxDBRelatorio\_OS\_Servicos} \rangle \cdot \text{"minutos"}) / 60)]$  .

Dividindo os minutos por 60, e com a função “Trunc” que arredonda para baixo, tirando a parte fracionaria da divisão, é obtido as horas gastas para a conclusão do serviço.

Função para obter os minutos:  
 $[(\langle \text{frxDBRelatorio\_OS\_Servicos} \rangle \cdot \text{"minutos"} - \text{Trunc}(\langle \text{frxDBRelatorio\_OS\_Servicos} \rangle \cdot \text{"minutos"}) / 60) * 60]$  .

É descontado, do total de minutos, o valor do arredondamento da divisão do minuto por 60 (que são as horas) multiplicado por 60, de forma que só o que resta são os minutos.

Caso um item não tenha o evento “FINALIZADO”, o tempo dele é assumido como sendo o tempo previsto para a conclusão que é informado no serviço ao qual o item de serviço está ligado. Para esses

### 3.47 Total de Carga Horaria /DIA

Figura 16259:Relatório Carga Horária DIA

Fonte: Produzida pelo autor

### 3.48 Filtros do Relatório

Figura 16360: Definindo item por item qual data será filtrada

```
consulta:='where (I.data>='+quotedstr(dtp1.Text)+' and I.data<='+quotedstr(dtp2.Text)+')';
consulta:=consulta+'or (I.data is null and os.data_os>='+quotedstr(dtp1.Text);
consulta:=consulta+' and data_os<='+quotedstr(dtp2.Text)+')';
```

Fonte: Produzida pelo autor

- ✓ É filtrado apenas por período
- ✓ Trecho de código onde é criada a string de filtragem
- ✓ O filtro de período trabalha com duas situações:
- ✓ Se existe o evento “INICIO”, os itens de serviço são filtrados pela data do evento “INICIO”;
- ✓ Se não existe o evento “INICIO”, os itens de serviço são filtrados pela data de abertura da OS a qual cada item de serviço está ligado.

### 3.49 Estrutura do Relatório

- É um relatório em 1 nível com os campos “Dia”, “Horas” e “Minutos”.

### 3.50 Impressão do Relatório

Figura 16461: Impressão do relatório de Carga Horária/Dia

GOVONI	Página:	1
SAV - SISTEMA DE ADMINISTRAÇÃO DE LOJAS	Data:	12/07/2020
Total de Carga Horária/Dia Período: 01/04/2020 a 12/07/2020	Hora:	22:47:33
Dia: 12/05/2020	Tempo Total:	0 horas 17 minutos
Dia: 03/06/2020	Tempo Total:	144 horas 1 minutos
Dia: 06/06/2020	Tempo Total:	2 horas 0 minutos
Dia: 15/06/2020	Tempo Total:	1 horas 45 minutos
Tempo Total do Relatório:		148 horas 3 minutos

Fonte: Produzida pelo autor

### 3.51 Consulta do Relatório

Os registros da consulta são agrupados pela data, que pode ser data do evento “INICIO”, ou data da abertura OS caso este não exista para algum item de serviço.

Trecho da consulta que calcula faz os joins das tabelas, Conforme a figura abaixo:

Figura 16562: Recuperando os eventos de cada item de serviço

```

from os_servicos
left join os on os_servicos.id_os=os.id_os
left join (select * from os_eventos where evento='INICIO') as I on os_servicos.id_os_servico=I.id_os_servico
left join (select * from os_eventos where evento='FINALIZADO') as F on os_servicos.id_os_servico=F.id_os_servico
left join (select * from os_eventos where evento='PARADA') as Pa on os_servicos.id_os_servico=Pa.id_os_servico
left join (select * from os_eventos where evento='REINICIO') as R on os_servicos.id_os_servico=R.id_os_servico
join servicos on os_servicos.id_servico=servicos.codigo

```

Fonte: Produzida pelo autor

Figura 16663: Trecho da Consulta que Calcula os Minutos

```

case when I.evento is not null then I.data
else cast(os.data_os as date) end as dat,
sum(case when F.horas=null then servicos.horas
else
coalesce((F.data-I.data)*24*60+DATE_PART('hour', F.horas - I.horas)*60
+DATE_PART('minute', F.horas - I.horas),servicos.horas*60+servicos.minutos-
(R.data-Pa.data)*24*60+DATE_PART('hour', R.horas - Pa.horas)*60
+DATE_PART('minute', R.horas - Pa.horas),servicos.horas*60+servicos.minutos)
end) as minutos

```

Fonte: Produzida pelo autor

Figura 16764:Trecho da Consulta que agrupa data

```
group by dat
```

Fonte: Produzida pelo autor

A conversão dos minutos para horas e minutos funciona da mesma forma que no relatório Carga Horária por OS.

Figura 16865:Relatório de OS por Cliente

A imagem mostra uma janela de software intitulada "Relação de OS por Cliente". O formulário contém os seguintes campos e controles:

- Período de Apuração:** Duas caixas de texto com calendários para selecionar datas. O primeiro campo contém "01/01/2020" e o segundo "12/07/2020", separados pela palavra "até".
- Abertura OS / Fechamento OS:** Dois botões de opção. O primeiro, "Abertura OS", está selecionado com um círculo preenchido. O segundo, "Fechamento OS", não está selecionado.
- Status da OS:** Um menu suspenso com o valor "TODOS" selecionado.
- Tipo do Layout:** Um menu suspenso com o valor "FLUXO DE PLACAS" selecionado.
- Materiais de Consumo:** Um menu suspenso com o valor "NÃO" selecionado.
- Imp. Eventos:** Um menu suspenso com o valor "SDM" selecionado.
- Cliente:** Um menu suspenso com o valor "1000 PECAS LTDA ME" selecionado.
- Botões:** "Visualizar" (destacado com um retângulo azul) e "Cancelar".

Fonte: Produzida pelo autor

### 3.52 Filtros do Relatório

São iguais aos filtros do Relatório de OS Geradas, só que não tem o filtro "Consultor" e obriga a filtragem por cliente.

### 3.53 Estrutura do Relatório Analítico

A estrutura é igual do relatório Analítico de Relatórios de OS Geradas.

### 3.54 Estrutura do Relatório Fluxo de Placas

É um relatório em 2 níveis:

- 1º Nível: Mostra o nome do cliente filtrado;
- 2º Nível: Mostra a placa, modelo e cor do veículo e data de abertura e fechamento da OS.

### 3.55 Relatório Analítico

- É igual ao relatório analítico de Relatório de OS Geradas

### 3.56 Fluxo de Placas

Figura 16966: Fluxo de Placas

GOVONI						Página:	1
SAV - SISTEMA DE ADMINISTRAÇÃO DE LOJAS						Data:	12/07/2020
Relação de OS Por Cliente Período: 01/01/2020 a 12/07/2020						Hora:	23:21:49
Nome do Cliente: 3 IRMAOS COMERCIO DE AUTO PECAS							
Placa	Modelo	Cor	Dt da OS	Dt Fechamento	Status	Nº OS	
FFF6666	FERRARI	BRANCO	06/03/2020	02/07/2020	ENTREGUE	152	
KKK1234	V	PRETO	19/02/2020		EM SERVICIO	147	
FFF6000	3D	BRANCO	27/03/2020	02/07/2020	ENTREGUE	160	

Fonte: Produzida pelo autor

Mostra todos os veículos do cliente que estão em concerto ainda.

### 3.57 Relação de OS por Placas

Figura 17067: Relação de Placas



A imagem mostra a interface de uma janela de software intitulada "Relação de OS por Placas". A janela contém os seguintes elementos de controle:

- Período de Apuração:** Campos de data com ícones de calendário. O primeiro campo contém "01/04/2020" e o segundo, "12/07/2020", separados pela palavra "até".
- Abertura OS:** Um grupo de botões com o primeiro selecionado (radio button preenchido) e o segundo desativado.
- Situação de OS:** Um menu suspenso com o valor "TODOS".
- Tipo do Layout:** Um menu suspenso com o valor "ANALITICO".
- Materiais de Consumo:** Um menu suspenso com o valor "NÃO".
- Imp. Eventos:** Um menu suspenso com o valor "SIM".
- Placa:** Um campo de texto com o valor "DNF9551" e um ícone de lupa à direita.
- Botões:** "Visualizar" (destacado com um retângulo azul) e "Cancelar".

Fonte: Produzida pelo autor

### 3.58 Filtros do Relatório

São iguais aos filtros do Relatório de OS Geradas, só que não tem o filtro "Consultor" e nem o filtro "Cliente" e tem o filtro "Placa".

### 3.59 Estrutura do Relatório Analítico

A estrutura é igual a do relatório Analítico de Relatórios de OS Geradas.

### 3.60 Estrutura do Relatório

É um relatório em 2 níveis:

- 1º nível: Contem as informações da OS e do veículo;
- 2º nível: Contem as informações de serviços e de peças;



### 3.61 Relatório Analítico

É igual ao relatório analítico de Relatório de OS Geradas.

Figura 17168: Relação de OS por Placas Relatório Analítico

GOVONI					Página:	1
SAV - SISTEMA DE ADMINISTRAÇÃO DE LOJAS					Data:	12/07/2020
Relação de OS por Placas Período: 01/01/2020 a 12/07/2020					Hora:	23:27:26
Placa: JGS2000		Nome do Cliente: A BOLZAN & CIA LTDA				
Data	Nº OS	Status	Serviço	Qtd		
02/03/2020	149	ENTREGUE	TROCA DE PNEU GE	1 1		
Data	Nº OS	Status	Serviço	Qtd	Peças	Qtd
06/03/2020	151	EM SERVIÇO	LIMPEZA	1	CORREIA COM	3,020
					VELA MOTOR GM 2.4 8V	1,000
					ATUADOR HIDR.	1,000
Data	Nº OS	Status	Serviço	Qtd		
27/03/2020	157	ENTREGUE	REVISAO COMPLETA	1		

Fonte: Produzida pelo autor

### 3.62 Fluxo de Placas

Para o veículo filtrado, mostra o último proprietário, as vezes que ele passou pela oficina e, para cada vez que ele deu entrada na oficina, os serviços realizados sobre ele (a esquerda) e as peças utilizadas para o concerto(a direita).

### 3.63 Resumo de Serviços Executados

Figura 17269: Resumo de serviços executados

Resumo de Serviços Executados

Período de Apuração

01/02/2020 até 12/07/2020

Abertura OS  Fechamento OS

Status da OS

TODOS

Serviço

CHAFECAO PORTA DIREITA

Visualizar Cancelar

Fonte: Produzida pelo autor

### 3.64 Filtros do Relatório

Os que existem funcionam da mesma forma que os filtros do relatório de OS Geradas, só que tem um filtro a mais, “Serviço”, que permite filtrar por um serviço específico.

### 3.65 Estrutura

- É um relatório em 1 nível.

Figura 17370: Impressão do Relatório de Resumo de serviços executados

GOVONI			Página:	1
SAV - SISTEMA DE ADMINISTRAÇÃO DE LOJAS			Data:	12/07/2020
Resumo de Serviços Executados	Período:	01/02/2020 a 12/07/2020	Hora:	23:39:24
Descrição	Qtd	Valor		
LIMPEZA	2	R\$ 53,40		
Total Líquido:		R\$ 53,40		

Fonte: Produzida pelo autor

Agrupar cada serviço (ou um só caso tenha sido filtrado) e soma a quantidade e valor cada vez que ele é agrupado.

Na totalização são somados os valores de todos os serviços que aparecem na consulta.

### 3.66 Relação de OS Técnicos

Figura 17471: Relatório de OS por Técnicos

Relação OS por Técnicos

Período de Apuração

01/03/2020 até 31/07/2020

Abertura OS  Fechamento OS

Status da OS

ENTREGUE

Inp. Eventos

SPM

Serviço

Técnico

Visualizar Cancelar

Fonte: Produzida pelo autor

### 3.67 Filtros do Relatório

Até a parte o filtro “Serviço”, filtra da mesma forma que o relatório Relação de OS Geradas.

Permite também filtrar por um serviço e por em técnico.

### 3.68 Estrutura

É um relatório em 3 níveis.

- 1º nível: Contem o nome do Técnico;
- 2º nível: Contem as informações referentes as várias OS em que cada técnico atuou;
- 3º nível: Contem as informações dos itens de serviço e seus eventos, da OS que está no nível 2, e que foram executados pelo técnico que está no nível 1;

### 3.69 Impressão do Relatório

Figura 17572: Impressão do Relatório de OS por Técnicos

Técnico: ALEXANDRE SILVA							
OS Nº	Dt Abertura	Dt Fechamento	Cliente	Placa	Modelo	Status da OS	
167	03/06/2020		A BOLZAN & CIA LTDA	MF9551	K	ORÇAMENTO	
Itens da OS							
IT	Descrição	Lado	Qtd	Pt. Lq. Unit.	Evento	Data	Status
1	RTHRTYT	GRE	1	500,00	INICIO	03/06/2020	APROVADO
					PARADA	03/06/2020	
					REINICIO	03/06/2020	
					FINALIZADO	03/06/2020	
OS Nº	Dt Abertura	Dt Fechamento	Cliente	Placa	Modelo	Status da OS	
168	03/06/2020	03/06/2020	2601 COM DE AUTO PECAS LTDA	FDS1122	G	AG RETRADA	
Itens da OS							
IT	Descrição	Lado	Qtd	Pt. Lq. Unit.	Evento	Data	Status
1	PINTURA PORTA DIANTEIRA ESQUERDA DSF		2	160,00	INICIO	06/06/2020	APROVADO
					REINICIO	06/06/2020	
<b>Valor Total Técnico:</b>				620,00			

Fonte: Produzida pelo autor

São impressas todas as OS na qual cada técnico teve participação, cada serviço realizado em cada OS e por cada técnico, bem como seus eventos. Abaixo, o valor desses serviços é totalizado.

No final da OS é totalizado o valor dos serviços realizados por todos os técnicos filtrados do período:

Figura 17673: Impressão do Relatório de OS por Técnicos total geral

<b>Total Geral:</b>	<b>702.833,50</b>
---------------------	-------------------

Fonte: Produzida pelo autor

Figura 17774: Previsão de Entrega de OS

Fonte: Produzida pelo autor

### 3.70 Filtros do Relatório

Os que existem funcionam da mesma forma que os filtros do Relatório de OS Geradas.

### 3.71 Estrutura

- ✓ É um relatório 1 nível.

Figura 17875: Estrutura da Previsão de Entrega de OS

GOVONI				Página:	1
SAV - SISTEMA DE ADMINISTRAÇÃO DE LOJAS				Data:	13/07/2020
Previsão de Entrega de OS		Período: 01/07/2010 a 13/07/2020		Hora:	00:27:58
Dt Entrega	Placa	Carro	Status	Cliente	
19/02/2020	KKK1234	V	EM SERVICO	3 IRMAOS COMERCIO DE AUTO	
06/03/2020	JGS2000	GF	EM SERVICO	A BOLZAN & CIA LTDA	
16/03/2020	LLL6666	F	EM SERVICO	1000 PECAS LTDA ME	
27/03/2020	PLO2222	C4	EM SERVICO	1000 PECAS LTDA ME	
31/03/2020	JHG1234	C4	EM SERVICO	1000 PECAS LTDA ME	
03/06/2020	IWF9551	K	ORCAMENTO	A BOLZAN & CIA LTDA	
15/06/2020	TTT2000	F8	EM SERVICO	AGLAE T DE ARAUJO ME	

Fonte: Produzida pelo autor

### 3.72 Impressão do Relatório

Mostra os a previsão de entrega de cada veículo que ainda está em manutenção.

### 3.73 Tela Manutenção de OS

Figura 17976: Tela Manutenção de OS

Manutenção de OS

Serviços Postado Casos Impressão Fecha OS Relatório Ativa Grupos NFSE Ajuda

Página Ocorrência Linha Técnica Observação

Nº OS: 070 Cliente: AGLNET DE ARAUJO NE Status: EM SERVIÇO Seção: REDPTA Data OS: 15/06/2020

Rack: 1112000 Polígonos: F080 Modelo: P6 Car: P600 Ass.Medida: 0000000 Banco/Tag: 4 Preço de Entrada: 1500000

Causa: 2004000000 Segredaria: 0000000000 Nº de controle/tecnico: Mensagem de Garantia de OS: Data Fichamento: / /

id	código	descricao	Lado	qtd	Nº Usados	Box	Técnico	Última Invenção	Estado	Status
1	1	PINTURA PORTA DA FERRA ESQUERDA C/D		1	200,00			FINALIZADO	Exatidão	APROVADO
2	1	TROCA DE FIBRA		1	90,00				Exatidão	APROVADO

Condição de Pagamento: 30 DIAS DEPOIS A PRATO Val. Serviços: 400,00 DESCONTO(%): 20,00 Val. Liquidar: 320,00

Novo Atualizar Excluir Gravar Imprimir Cancelar Zerar Atualizar Fórmula Limpar Fechar

Fonte: Produzida pelo autor

### 3.74 Características Principais da Tela

- É a tela principal da parte de OS;
- Possui links para todas as outras telas da módulo de OS, exceto os relatórios;
- Interage com todas as outras telas da módulo de OS;
- As interações e eventos da tela, inclusive seus campos, já foram detalhados nos tópicos anteriores.

### 3.75 Exemplo de Geração de Uma OS

Primeiramente o usuário vai fazer um orçamento.

Com o orçamento salvo, o usuário imprime a OS clicando no botão do menu superior “Imprimir”.

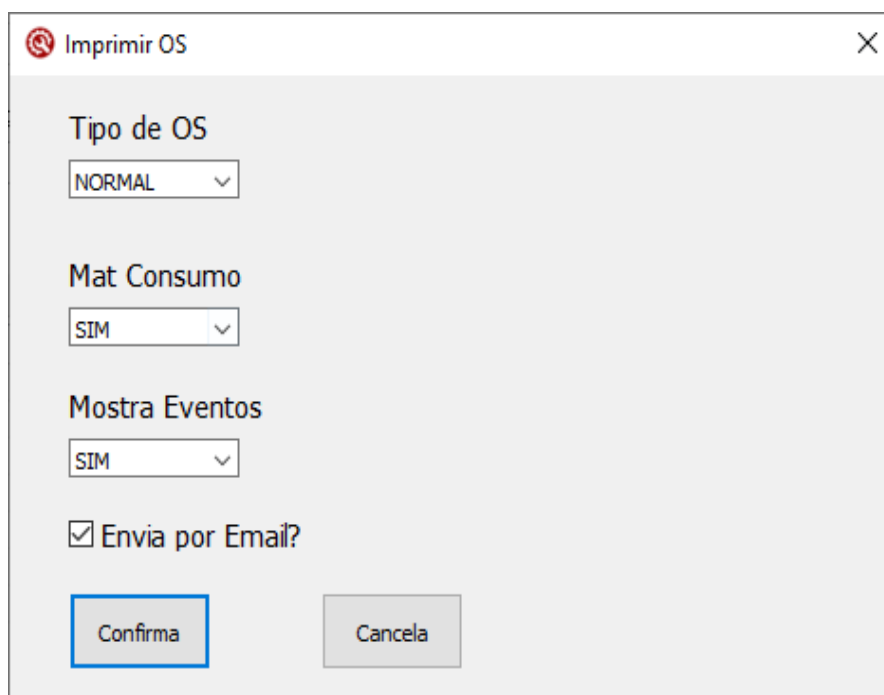
Figura 18077: Menu Superior da tela Manutenção de OS



Fonte: Produzida pelo autor

Ao clicar em “Imprimir” será aberta a tela abaixo com filtros e possibilidade de enviar por e-mail:

Figura 18178: Filtros da Impressão da OS

A imagem mostra uma janela de diálogo intitulada "Imprimir OS" com um ícone de lupa e um botão de fechar (X) no canto superior direito. O conteúdo da janela é organizado em seções: "Tipo de OS" com um menu suspenso selecionando "NORMAL"; "Mat Consumo" com um menu suspenso selecionando "SIM"; "Mostra Eventos" com um menu suspenso selecionando "SIM"; e uma caixa de seleção marcada "Envia por Email?". Na base da janela, há dois botões: "Confirma" (destacado com um contorno azul) e "Cancela".

Fonte: Produzida pelo autor





- Depois vão aparecer os itens para que o cliente aprove ou desaprove

Figura 18380: Impressão do Relatório 1

Itens da OS										
IT	Descrição do Serviço	Leite	Técnico	Box	Qtd	Valor Total	Evento	Data	Hora	Status
1	OSFSD				1	R\$ 1.231,00				APROVADO
2	OSFSD		T2		1	R\$ 1.231,00				APROVADO
3	T43T43				1	R\$ 43,00				APROVADO
4	OSFSD				1	R\$ 1.231,00				APROVADO
5	OSFSD				1	R\$ 1.231,00				APROVADO
6	T3T4				1	R\$ 32,00				APROVADO
7	OSFSD				1	R\$ 12.301,00				APROVADO
Total Líquido Serviços:										R\$ 8.650,00
Total Líquido Peças:										R\$ 0,00
Total Líquido Mat Consumido:										R\$ 0,00
PORTO ALEGRE, 13 de JULHO de 2021								Valor Bruto:	R\$ 17.300,00	
								Valor do Desconto Serviços:	R\$ 8.650,00	
								Valor do Desconto Peças:	R\$ 0,00	
								Valor Líquido:	R\$ 8.650,00	
3 IRMÃOS COMERCIO DE AUTO PEÇAS										
Garantia de peças 90 dias, garantia de mão de obra 30 dias. Não nos responsabilizamos pela origem dos equipamentos recebidos no carro. Mau uso, ou eventos de natureza, não são cobertos por essa garantia.										

Fonte: Produzida pelo autor

Caso o documento seja aprovado, o usuário modifica o status da OS para “em serviço”, cria o evento cria o evento “INICIO”, com data e hora, para os itens de serviço que já vão ser iniciados e indica, para cada um deles, o técnico responsável (opcional).

Ao final do expediente diário, ou devido a qualquer outro motivo que leve o técnico a interromper a execução de um serviço, é criado o status “PARADA”, com data e hora, para que esse tempo não seja contabilizado nos relatórios de tempo total para realização do serviço. Quando o serviço é retomado o usuário cria o status “REINICIO”, com data e hora. Para cada serviço concluído da OS é criado o evento “FINALIZADO”, com data e hora. Essas ações não são obrigatórias no sistema, mas permitem um controle preciso sobre cada ação realizada na oficina através da geração de relatórios diversos que mostram cada setor da oficina por vários ângulos.

Normalmente não é aconselhável excluir itens de serviço, a menos que tenha havido um engano, porque o registro da rejeição de um item protege a oficina em caso de algum problema vim a acontecer em decorrência de um serviço que deveria ter sido realizado sobre o veículo mas que foi rejeitado pelo cliente. Por isso

o sistema permite rejeitar algum item de serviço caso ele não tenha tido evento de início ainda, mesmo que o status da OS não seja de orçamento.

Quando efetuada a quitação do serviço, a OS pode ser fechada. A partir desse momento, a OS não pode mais ser alterada.

Para o operador alterar uma OS já fechada, se faz necessário à intervenção de um supervisor com permissão especial. Essa operação é registrada no log do banco do sistema e se trata de uma exceção, não do funcionamento normal do sistema.

## **5 VALIDAÇÃO**

Durante o projeto proposto o desenvolvimento de um módulo de ordens de serviço automotivo em Delphi, que foi concluído no dia 14/07/2020.

## **6 CONCLUSÃO**

Nesse trabalho foi desenvolvido um módulo completo de manutenção de ordens de serviço, como complemento para o sistema de gestão SAV, Sistema de Administração e Vendas. O desenvolvimento desse trabalho me forçou diversas vezes a achar soluções para problemas técnicos, quer seja por conta própria ou com os responsáveis pelo sistema, para criar soluções que atendessem as necessidades dos usuários. Por conta disso, desenvolvi minhas habilidades como programador, aprendi bastante a respeito do mercado de serviços automobilísticos e a desenvolver sistemas para esse nicho, desenvolvi novos conhecimentos sobre sistemas de forma geral e, mais importante, aprendi a encontrar soluções para problemas que aparecem durante o desenvolvimento de qualquer projeto. Foi observado que existe uma resistência inicial por parte das empresas do ramo que ainda não possuem sistema, essa resistência tende a diminuir depois do primeiro contato e exposição do sistema. Foi observado também que o primeiro contato tem uma receptividade muito melhor por parte das oficinas quando elas já têm referências do sistema por parte de alguma loja de autopeças que usa o sistema.



## 8 REFERÊNCIAS BIBLIOGRÁFICA

AGENCIA, Icone. **Qual a importância da Ordem de Serviço?** 2020. Disponível em: <<https://www.tecinco.com.br/como-tirar-ordem-de-servico/>>. Acessado em 12 junho 2020.

SANCES. **O Rumo do sucesso.** 2018. Disponível em: <<https://sances.com.br/blog/sistema-para-concessionaria/>>. Acessado em 2 junho 2020.

SCHULTZ, Felix. **O que contém uma ordem de serviço.** 2018. Disponível em: <<https://bomcontrole.com.br/ordem-de-servico/>>. Acessado em 10 maio 2020.

Equipe TOTVS. **Benefícios de um sistema de ordem de serviço.** 2019. Disponível em <<https://www.totvs.com/blog/gestao-de-servicos/sistema-de-ordem-de-servico/>>. Acessado em 8 junho 2020.

Felipe Machado. **Motivos para ser um Programador Delphi.** 2020. <<http://delhiparainiciantes.com.br/10-motivos-para-ser-um-programador-delphi/>>. Acessado em 8 junho 2020.

Lexos ERP omnichannel. **Sistema de Ordem de Serviço Flexível.** 2020. <<https://www.lexos.com.br/sistema-ordem-de-servico/>>. Acessado em 8 junho 2020.

Bom Controle. **O que contém uma ordem de serviço.** 2020. <<https://bomcontrole.com.br/ordem-de-servico/>>. Acessado em 8 junho 2020.