

CENTRO ESTADUAL DE EDUCAÇÃO TECNOLÓGICA PAULA SOUZA

**FACULDADE DE TECNOLOGIA DE LINS PROF. ANTONIO SEABRA
CURSO SUPERIOR DE TECNOLOGIA EM BANCO DE DADOS**

**LEOCELIO APARECIDO SANTANA VALENTE
RODRIGO ALEXANDRE SHIRAKASHI**

**SISTEMA WEB PARA CONTROLE DE ESTOQUE DE UMA REVENDA
DE VEÍCULOS USADOS**

CENTRO ESTADUAL DE EDUCAÇÃO TECNOLÓGICA PAULA SOUZA

**FACULDADE DE TECNOLOGIA DE LINS PROF. ANTONIO SEABRA
CURSO SUPERIOR DE TECNOLOGIA EM BANCO DE DADOS**

**LEOCELIO APARECIDO SANTANA VALENTE
RODRIGO ALEXANDRE SHIRAKASHI**

SISTEMA WEB PARA CONTROLE DE ESTOQUE DE UMA REVENDA DE VEÍCULOS USADOS

Trabalho de Conclusão de Curso apresentado à
Faculdade de Tecnologia de Lins para obtenção
do Título de Tecnólogo em Banco de Dados.

Orientador: Prof. Me. Gisele Molina Becari

**LINS/SP
1º SEMESTRE/2016**

**LEOCELIO APARECIDO SANTANA VALENTE
RODRIGO ALEXANDRE SHIRAKASHI**

**SISTEMA WEB PARA CONTROLE DE ESTOQUE DE UMA REVENDA
DE VEÍCULOS USADOS**

Trabalho de Conclusão de Curso apresentado à
Faculdade de Tecnologia de Lins, como parte dos
requisitos necessários para a obtenção do título de
Tecnólogo em Banco de Dados sob orientação do
Prof. Me. Gisele Molina Becari.

Data de aprovação: _____ / _____ / _____

Orientador (Prof. Me. Gisele Molina Becari)

Examinador 1 (Prof. Me. Fábio Lucio Meira)

Examinador 2 (Prof. Me. Rodrigo Moura J. Ayres)

Dedico este trabalho a Deus, que esteve presente em todos os momentos de minha vida. Aos meus pais que sempre lutaram para propiciar uma melhor qualidade de vida aos seus filhos. A minha esposa Elaine, que sempre esteve ao meu lado nos momentos mais difíceis. A minha filha Dhafine, meu motivo de existir. E ao meu parceiro Leocelio, grande amigo e companheiro.

Rodrigo Alexandre Shirakashi

Dedico este trabalho a Deus, que sempre me amparou em todos os momentos. Aos meus pais que sempre estiveram ao meu lado, lutando para me proporcionar uma vida melhor. A minha esposa Débora, que mesmo gestante e passando noites solitárias, nunca deixou de me apoiar. E a minha filha Sofia, razão do meu viver. E ao meu parceiro Rodrigo, sempre amigo e companheiro.

Leocelio Aparecido Santana Valente

AGRADECIMENTOS

No encerramento de uma etapa tão importante que se conclui, agradecemos primeiramente a Deus por nos dar força e saúde para vencermos esse desafio.

A nossa família que sempre nos incentivaram e nos encorajaram nos momentos mais difíceis dessa caminhada.

A nossa orientadora Prof.^a Me. Gisele Molina Becari, pela paciência em nos orientar, por sempre se preocupar com os rumos do projeto, incentivando e sugerindo soluções, sempre nos motivando e tentando extrair nosso melhor.

Também gostaríamos de registrar o sincero agradecimento a todo corpo docente e funcionários da FATEC Lins, pelos ensinamentos, conselhos compartilhados e todo auxílio auxílio e estrutura oferecidos.

Não podemos nos esquecer dos funcionários da Amigão Veículos, por disponibilizar algumas seu tempo para atender-nos, sendo fundamental para a análise de negócios e levantamento de requisitos do sistema.

RESUMO

Este projeto tem a finalidade de aplicar a tecnologia da informação nos processos administrativos de uma revenda de automóveis usados, através do desenvolvimento de um sistema web. As principais funções que serão informatizadas são: cadastro de clientes, funcionários, veículos, despesas, marca e modelo; controlar a compra, manutenção e venda dos mesmos. No inicio, são realizados estudos sobre as tecnologias aplicadas no desenvolvimento do sistema, como por exemplo, a arquitetura MVC, a IDE Netbeans, o sistema gerenciador de banco de dados MySQL e a linguagem de programação PHP. Após o estudo, foi realizada uma abordagem sobre a importância do controle de estoque, levantamento dos problemas encontrados na empresa e uma perspectiva do produto. Em seguida, fez-se à elaboração dos diagramas, essenciais para o desenvolvimento do trabalho. E por fim a implementação e apresentação de algumas telas do software. Apesar deste sistema poder ser utilizado, verificou-se ainda a necessidade de aperfeiçoar o mesmo, com o intuito de torná-lo mais completo e seguro, e ainda para suprir futuras exigências de quem o utilizará.

Palavras-chave: sistema web, controle de estoque, projeto, tecnologias

ABSTRACT

This project aims to apply information technology in the administrative processes of a used car dealership, by developing a web system. The main functions to be computerized are: customer base, employees, vehicles, expenses, make and model; control the purchase, maintenance and sale of the same. At first, studies are conducted on the technologies applied in the development of the system, such as the MVC architecture, the Netbeans IDE, the database management system MySQL and the PHP programming language. After the study, a discussion of the importance of inventory control was carried out survey of the problems encountered in the company and an overview of the product. Then we made the preparation of diagrams, essential for the development work. Finally the implementation and presentation of some screens of the software. Although this system can be used, although there is a need to improve it, in order to make it more complete and secure, and also to meet future demands of those who will use it.

Keywords: web system , inventory control , design, technology

LISTA DE ILUSTRAÇÕES

Figura 2.1 - Ficha Cadastral (Frente)	30
Figura 2.2 - Ficha Cadastral (Verso)	31
Figura 2.3 - Ficha Cadastral de Cliente.....	32
Figura 2.4 - Ficha Compra/Venda/Troca/Financia.....	33
Figura 3.1 - Diagrama de Caso de Uso Pessoa.....	37
Figura 3.2 - Diagrama de Classes.....	38
Figura 3.2 - Projeto de banco de dados	39
Figura 3.4 - Diagrama de Atividade Manter Pessoa.....	40
Figura 3.5 – Diagrama de Atividade Manter Funcionário	41
Figura 3.6 – Diagrama de Atividade Login	41
Figura 3.7 – Diagrama de Atividade Manter Marca	42
Figura 3.8 – Diagrama de Atividade Manter Veículos	42
Figura 3.9 – Diagrama de Atividade Manter Modelo	43
Figura 3.10 – Diagrama de Atividade Manter Movimento	43
Figura 3.11 – Diagrama de Atividade Manter Despesas	44
Figura 3.12 – Diagrama de Atividade Manter Despesas do Veículo	45
Figura 3.13 – Diagrama de MVC Manter Pessoa.....	45
Figura 3.14 – Diagrama de MVC Manter Despesas.....	46
Figura 3.15 – Diagrama de MVC Manter DespesasVeiculo	46
Figura 3.16 – Diagrama de MVC Manter Funcionário	47
Figura 3.17 – Diagrama de MVC Manter Login	47
Figura 3.18 – Diagrama de MVC Manter Movimento	48
Figura 3.19 – Diagrama de MVC Manter Marca.....	48
Figura 3.20 – Diagrama de MVC Manter Veiculo	49
Figura 3.21 – Diagrama de MVC Manter Modelo	49
Figura 3.22 – Diagrama de Sequência Manter Pessoa.....	50
Figura 3.23 – Diagrama de Sequência Manter Despesas.....	51
Figura 3.24 – Diagrama de Sequência Manter DespesasVeiculo	51
Figura 3.25 – Diagrama de Sequência Manter Funcionário	52
Figura 3.26 – Diagrama de Sequência Login	52
Figura 3.27 – Diagrama de Sequência Manter Movimento	53
Figura 3.28 – Diagrama de Sequência Manter Marca.....	53

Figura 3.29 – Diagrama de Sequência Manter Veiculo	54
Figura 3.30 – Diagrama de Sequência Manter Modelo	54
Figura 4.1 – Estrutura de tabelas do banco de dados.....	56
Figura 4.2 – Criação de tabelas do banco de dados	57
Figura 4.3 – Tela de Login.....	57
Figura 4.4 – Tela de Design Responsivo com resolução 1366 x 768.....	58
Figura 4.5 – Tela de Design Responsivo com resolução 800 x 600.....	58
Figura 4.6 – Tela com resolução 400x800 com menu expandido	59
Figura 4.7 – Tela com resolução 400x800 com menu recolhido	59
Figura 4.8 – Tela principal do Sistema	60
Figura 4.9 – Tela Menu Cadastro de Clientes - Opção Cadastrar.....	61
Figura 4.10 – Tela Menu Cadastro de Clientes, opção Consultar	62
Figura 4.11 – Tela Menu Cadastro de Clientes, opção Excluir.....	62
Figura 4.12 – Tela Validação CPF	63
Figura 4.13 – Tela Validação CNPJ	63
Figura 4.14 – Tela de Cadastro de Funcionários	64
Figura 4.15 – Tela de Menu Veiculo.....	64
Figura 4.16 – Tela de cadastro de marcas	65
Figura 4.17 – Tela de cadastro de modelos	65
Figura 4.18 – Tela de cadastro de despesas	66
Figura 4.19 – Tela de Movimento.....	66
Figura 4.20 – Tela de Movimento, Entrada	67
Figura 4.21 – Tela de Movimento, Consultar Entrada	68
Figura 4.22 – Tela de Movimento, Saída	69
Figura 4.23 – Tela de Movimento, Consultar Saída	70

LISTA DE TABELAS

Tabela 2.1 - Necessidades/Riscos.....	36
---------------------------------------	----

LISTA DE ABREVIATURAS E SIGLAS

UML: *Unified Modeling Language*

PHP: *Personal Home Page (Hypertext Preprocessor)*

MVC: *Model-View-Controller*

IDE: *Interface Development Environment*

SGBD: Sistema Gerenciador de Banco de Dados

MER: Modelo Entidade Relacionamento

SQL: *Structured Query Language*

HP: *Hewlett-Packard*

AIX: *Advanced Interactive Executive*

SCO: *Santa Cruz Operation*

OS: *Operating System*

API: *Application Programming Interface*

JSP: *Java Server Pages*

JSF: *Java Server Faces*

HTTP: *Hyper Text Transfer Protocol*

HTML: *Hypertext Markup Language*

SGML: *Standart Generalized Markup Language*

W3C: *World Wide Web Consortium*

HYTIME: *Hypermedia Time-base Structuring Language*

CSS: *Cascading Style Sheets*

ODBC: *Open Database Connectivity Standart*

SUMÁRIO

1 TECNOLOGIAS UTILIZADAS.....	17
1.1 ORIENTAÇÃO A OBJETOS.....	17
1.2 MVC	18
1.3 NETBEANS	18
1.4 ASTAH COMMUNITY	18
1.5 UML.....	19
1.6 BANCO DE DADOS.....	19
1.7 SGBD	21
1.8 MYSQL.....	21
1.9 APACHE TOMCAT	23
1.10 LINGUAGEM DE MARCAÇÃO	23
1.11 HYPERTEXT MARKUP LANGUAGE (HTML).....	23
1.12 CASCADING STYLE SHEETS CSS	24
1.13 PHP	25
1.14 JQUERY.....	26
1.15 BOOTSTRAP	26
1.16 AJAX	26
1.17 PHPMYADMIN	27
1.18 CONSIDERAÇÕES SOBRE O CAPÍTULO.....	27
2 ANÁLISE E PROJETO DE SISTEMA	28
2.1. ANÁLISE DE NEGÓCIO	28
2.1.1. Estoque	28
2.1.2 Instrução do problema.....	29
2.1.3 Atores e envolvidos no processo.....	34
2.2 VISÃO GERAL DO PRODUTO	34
2.2.1 Perspectiva do Produto (Análise de Mercado)	34
2.2.3 Características	36
3 ANÁLISE E MODELAGEM DO SISTEMA.....	37
3.1 ANÁLISE E PROJETO	37
3.1.1 Diagrama de Caso de Uso, visão geral.....	37
3.2 DIAGRAMA DE CLASSE	38
3.3 PROJETO DE BANCO DE DADOS	39
3.4 DIAGRAMA DE ATIVIDADES	40

3.4.1 Manter Pessoa	40
3.4.2 Manter Funcionário	41
3.4.3 Login.....	41
3.4.4 Manter Marca	42
3.4.5 Manter Veículos	42
3.4.6 Manter Modelo	43
3.4.7 Manter Movimento.....	43
3.4.8 Manter Despesas	44
3.4.9 Manter Despesas do Veículo	44
3.5 MVC	45
3.5.1 Manter Pessoa	45
3.5.2 Manter Despesas	46
3.5.3 Manter DespesasVeiculo.....	46
3.5.4 Manter Funcionário	47
3.5.5 Manter Login	47
3.5.6 Manter Movimento.....	48
3.5.7 Manter Marca	48
3.5.8 Manter Veiculo	49
3.5.9 Manter Modelo	49
3.6 DIAGRAMA DE SEQUÊNCIA	50
3.6.1 Manter Pessoa	50
3.6.2 Manter Despesas	51
3.6.3 Manter DespesasVeiculo.....	51
3.6.4 Manter Funcionário	52
3.6.5 Login.....	52
3.6.6 Manter Movimento.....	53
3.6.7 Manter Marca	53
3.6.8 Manter Veiculo	54
3.6.9 Manter Modelo	54
4 DESENVOLVIMENTO DO SISTEMA	56
4.1 DESENVOLVIMENTO DO BANCO DE DADOS.....	56
4.1 LOGIN	57
4.2 DESIGN RESPONSIVO	58

4.3 TELA PRINCIPAL	60
CONCLUSÃO.....	71
REFERÊNCIAS BIBLIOGRÁFICAS	72

INTRODUÇÃO

O mundo dos negócios, tanto em nível mundial quanto em nacional, tem sofrido profundas transformações nos últimos anos, as quais têm sido rigorosamente associadas com a tecnologia de informação. A aplicação da TI nas empresas objetiva maior rendimento e poder corporativo. O efeito deste acontecimento é constatado em todas as organizações, independentemente de seu tamanho ou tipo de negócio. A proporção do impacto, notoriamente, alterna de acordo com a disputa e turbulência do ambiente de cada área. Contudo, instituições de menor porte tendem a ter mais contratemplos e são mais frágeis às ameaças do mercado (DIAS, 1993).

O presente projeto tem como objetivo o desenvolvimento de um sistema que melhore a gestão das rotinas de uma revenda de veículos usados. O software terá que possibilitar ao usuário meios que facilitem o gerenciamento das vendas, os cadastros de pessoas, veículos e seus acessórios.

Para identificar os problemas encontrados na atual administração da empresa, foi realizado um processo de análise de negócio e levantamento de requisitos, para que em seguida o projeto seja desenvolvido com a menor quantidade de falhas possível.

Um estabelecimento que não utiliza um controle de estoque nos dias atuais pode apresentar sérios danos em suas finanças. Rotinas administrativas que são realizadas manualmente se tornam mais vulneráveis e propensas a apresentar erros. Outro fator negativo é o prazo mais extenso com que os afazeres são praticados, tornando o suporte ao cliente mais prolongado, deixando o cliente insatisfeito.

Essa série de fatores faz com que a empresa tenha inferioridade em relação aos concorrentes e também diminuição de suas receitas ao longo do tempo.

O projeto tem o propósito de elaborar um sistema que possa sanar as adversidades mencionadas acima, otimizando a forma com que a empresa trabalha atualmente.

O estudo foi dividido em quatro capítulos que serão melhor abordados nos parágrafos seguintes.

No primeiro capítulo são levantadas as tecnologias aplicadas para o evolução do sistema, tecnologias que serão utilizadas desde a modelagem do sistema, criação do banco de dados, até o desenvolvimento da aplicação web.

O segundo capítulo traz uma breve abordagem sobre a importância do controle de estoque e análise de negócio. Essa análise, esboça os procedimentos e estágios relevantes para a elaboração do sistema. Ainda é fornecida, uma descrição das dificuldades identificadas através de conversas com os funcionários da empresa.

No terceiro capítulo, é exibida a estrutura do sistema através de diagramas relacionados ao programa. Os diagramas que são abordados no capítulo são o de caso de uso, de classe, de atividades, MVC, e sequência.

No quarto capítulo são exibidas as telas do projeto que farão a interação com o funcionário que opera os computadores da empresa, externando suas utilidades.

1 TECNOLOGIAS UTILIZADAS

O primeiro capítulo será destinado às considerações das tecnologias que serão utilizadas na construção do software sugerido neste estudo. Para uma melhor elaboração e detalhamento de cada uma das características do sistema, será utilizada a linguagem de modelagem *Unified Modeling Language* (UML). A mesma irá auxiliar e nortear as decisões que serão tomadas no decorrer do trabalho. O conjunto formado pela linguagem de programação PHP e pelo banco de dados MySQL, assim como as demais ferramentas utilizadas durante o desenvolvimento do projeto, serão de suma importância para que o sistema tenha eficácia e consiga dar fim aos problemas gerenciais encontrados atualmente na empresa que comercializa veículos.

1.1 ORIENTAÇÃO A OBJETOS

Para Mendes (2010), embora tenha sido criada há muito tempo, a programação orientada a objetos ganhou maior destaque somente a partir da evolução da linguagem Java. Distinta da programação estruturada, a programação orientada a objetos procura se basear em características dos seres humanos para administrar as dificuldades de um software. Nesse contexto o mundo real é feito de objetos, que se comunicam, e cada objeto tem seu próprio estado que são chamados de atributos e comportamento que são conhecidos como métodos.

De acordo com Deitel (2010) desenvolvedores em Java centralizam-se na criação de classes. Cada classe contém informações e métodos que manejam os campos e manipulam os serviços. A partir dos dados dessas classes criadas eles constroem novas classes.

Ainda para Deitel (2010) através da orientação a objetos você pode fazer boa parte do software ajustando classes. Conforme surge uma nova classe, ela terá potencial de se tornar um novo “ativo de software” que poderá ser utilizado para aprimorar e facilitar a qualidade de seus esforços posteriores no desenvolvimento de um software.

1.2 MVC

A arquitetura *Model-View-Controller* (MVC) começou com a prática da interface de usuário Smalltalk-80. Atualmente grande parte dos softwares para a Internet são desenvolvidos com base nos modelos MVC, isso permite que projetos amplos e robustos possam ser desmembrados, possibilitando um desenvolvimento mais organizado e um resultado mais satisfatório. O modelo MVC também pode ser utilizado no planejamento de sistemas menos sofisticados. (BURBECK, 1992)

Ainda para Burbeck na criação de um sistema web a partir do modelo MVC, o mesmo se dividirá em três partes independentes: modelo, visão e controlador. O Modelo (Model) consiste nas informações da aplicação, lógica e funções, a Visão (View) corresponde a interação com o usuário ou diferentes programas e o Controlador (Controller) que manipula o andamento da aplicação, ou seja, recebe os dados da entrada e faz a comunicação com o Modelo e Visão.

1.3 NETBEANS

De acordo com Gonçalves (2006), o NetBeans é uma *Interface Development Environment* (IDE) sólida, criada pela Sun, que cada vez mais ganha espaço no mercado devido as suas características, como sua complexidade, por ser um software gratuito e por ser multiplataforma.

1.4 ASTAH COMMUNITY

Astah Community é um software utilizado no desenvolvimento de diagramas e ilustrações baseadas no padrão UML, o que torna o sistema mais rápido e melhora a qualidade da modelagem. Utilizando o Astah, pode-se utilizar a orientação a objetos. Com a ferramenta Astah, abre-se a possibilidade de apresentar-se um melhor entendimento do problema e dessa forma a compreensão do time de desenvolvimento, pois sua finalidade é obter uma melhor comunicação através dos diagramas. (ASTAH, 2012)

1.5 UML

De acordo com Medeiros (2004) a UML é um modo de transmitir algo e tem a utilidade de mostrar as diversas maneiras que podem ser apresentadas um programa em várias etapas de sua construção. Com ela é possível planejar um sistema em determinado local e compilar em outro, sempre com conversão entre ambos, porém de forma minimizada.

Segundo Deitel (2010) quando um sistema é projetado através da UML, são usados diagramas de atividades, dessa forma o fluxo de controle é apresentado em todas as instruções de controle Java, também são utilizados diagrama de classes UML para simular visualmente classes e seus relacionamentos de herança. Anteriormente todo esse processo era representado através dos fluxogramas.

Ainda para Deitel (2010), uma solução que chama a atenção na UML, é a sua flexibilidade. Além de ser extensível, ou seja, ela pode ser melhorada com recursos novos, a mesma também é independente de qualquer outro processo particular. Os modeladores de UML são acessíveis para aproveitar vários processos para esboçar sistemas, porém os desenvolvedores expressar seus designers com um conjunto padronizado de notações gráficas. A UML é uma linguagem gráfica complexa, rica em recursos.

1.6 BANCO DE DADOS

A automação de tarefas relacionadas ao processamento de dados existe desde antes da invenção dos computadores, através de cartões perfurados que foram inventados por Hermann Hollerith, com a finalidade de registrar dados de censo nos Estados Unidos e que posteriormente seriam tabulados por instrumentos mecânicos. (SILBERSCHATZ, 2006)

O crescimento do uso de computadores está causando um grande impacto nas tecnologias e nos bancos de dados, sendo bem visível a enorme importância que eles representam em quase todas as áreas, principalmente na forma como são armazenados os estoques. (ELSMASRI; NAVATHE, 2006)

Segundo Silberschatz (2006), vários mecanismos surgiram no decorrer do tempo para armazenagem dos dados até a criação dos discos rígidos, onde permitiu

o acesso direto aos dados independentemente do local em que se encontravam armazenados. Dando origem a criação dos primeiros bancos de dados.

Elmasri, Navathe (2006) definem banco de dados com uma coleção de dados relacionados, que são fatos que podem ser gravados e que possuem um significado subentendido, podendo ser de qualquer tamanho e de complexidade variável.

Para Date (2003) os bancos de dados são com um armário de arquivo, onde os arquivos de dados computadorizados são armazenados, e através de um sistema gerenciador, permite aos usuários efetuarem operações de manipulação.

Os dados de um banco de dados são usualmente titulados de persistentes, pois são introduzidas por algoritmos implantadas a um Sistema Gerenciador de Banco de Dados (SGBD) e manipuladas através de requisições diretas ao SGBD. (DATE, 2003)

Conforme Nassu e Setzer (2009) existem os bancos de dados orientados a objetos e os bancos de dados relacionais, que são os dois principais tipos de banco de dados.

Banco de dados orientado a objetos consistem em um banco cuja cada informação é armazenada na forma de objetos, onde os dados só podem ser manipulados através de métodos definidos pela classe que se encontra o objeto. (ELMASRI; NAVATHE 2005)

Segundo Welling e Thomson (2005), os bancos de dados relacionais são compostos por várias tabelas, nas quais chaves específicas servem como referência de uma tabela para outra. Esse tipo de banco de dados é o mais utilizado.

Para um projeto de banco de dados, é preciso realizar análises minuciosas com a finalidade de refinar o esquema lógico final, pois o Modelo Entidade Relacionamento (MER) representa somente uma descrição aproximada dos dados criados no desenvolvimento de análise de requisitos. Após a criação do esquema lógico, deve-se dar ênfase ao critério desempenho e criar o esquema físico. Finalmente criam-se restrições de segurança e garantem acessibilidade aos dados somente aos usuários que necessitem. (RAMAKRISHNAN; GEHRKE, 2008)

1.7 SGBD

Segundo Date (2003), um Sistema Gerenciador de Base de Dados (SGBD) tem a função de isolar os usuários do banco de dados dos detalhes do nível de hardware, igualmente aos sistemas de linguagens de programação resguardam os programadores de aplicações dos detalhes de nível de *hardware*. O SGBD é responsável de fornecer aos usuários uma visão do banco de dados acima do nível de hardware, admitindo operações dos usuários, assim como as operações de SQL.

Para Elmasri e Navathe (2005), um SGBD é composto por um conjunto de programas que permite aos usuários criar e manter um banco de dados. Logo, o SGBD, consiste num sistema de *software* com a finalidade de facilitar os processos de definição, construção, manipulação e compartilhamento de bancos de dados entre vários usuários e aplicações.

Conforme definem Nassu e Setzer (2009), SGBD é um *software* ou um conjunto de *softwares* com a finalidade de gerenciar todas as ações de um banco de dados. Dentre elas tais funções como: definição da estrutura de dados, inserção e busca de dados, recuperação de falhas na comunicação ou no meio de gravação, controles de concorrência, de acesso e de segurança dos dados.

1.8 MYSQL

Conforme Milani (2007), o MySQL trata-se de um Sistema Gerenciador de Banco de Dados (SGBD) relacional, possuidora de licença dupla, onde uma delas é de *software* livre e a princípio foi projetado com o intuito de ser utilizada com aplicações de pequeno e médio portes, entretanto nos dias atuais está atendendo a aplicações de grande porte e possuindo maiores vantagens do que seus competidores. O MySQL possuí todos os atributos que um banco de dados de grande porte necessita. Isso fez com que algumas entidades a reconhecesse como o banco de dados *open source* altamente capacitada para concorrer com programas similares de código fechado, tais como SQL Server, da Microsoft e Oracle.

Para Gonçalves (2006), o MySQL é um sistema de gerenciamento de banco de dados relacional criado para suprir a necessidade de um SGBD rápido e flexível, além de ser confiável e de nível corporativo e possui código-fonte aberto.

Segundo o site MYSQL (2013a), o programa MySQL é registrada pela Oracle Corporation e possuidora de dois tipos de licença, uma gratuita e a outra comercial padrão Oracle. Por possuir um rico e favorável conjunto de funções, para esta monografia foi utilizado o banco de dados MySQL *Community Edition*, que é uma versão livre e possui código fonte aberto e possui uma enorme comunidade ativa de desenvolvedores. Sua conectividade, velocidade e segurança fazem do MySQL uma ferramenta ideal para uso de banco de dados na *web*.

O programa de banco de dados MySQL é escrito na linguagem C e C++ e compatível com uma enorme variedade de compiladores diferentes e com diversas plataformas operacionais, dentre pode-se destacar: o Linux (Fedora, Core, Debian, SuSE, RedHat); Unix (Solaris, HP-UX, AIX, SCO); FreeBSD; Mac OS X Server; Windows (2000, 2003, xp). (MILANI, 2007)

De acordo com o Devmedia (2013a), através da integração das ferramentas *GNU Automake*, *Autoconf* e *Libtool*, fazem com que o MySQL seja uma aplicação altamente portável entre diferentes sistemas, plataformas e compiladores.

Em questão de conectividade, o MySQL possui conectores e *Application Programming Interface* (APIs), permitindo conexões e instruções MySQL sejam executadas com várias outras linguagens de programação, tais como o Java, Python, PHP, Pearl, C, C++, entre outras. (MILANI, 2007)

A manipulação de dados é feita através de interface *prompt* de linha de comando ou através de um programa de interface gráfica como, por exemplo, o MySQL *Workbench*, que consiste em uma ferramenta de modelagem de dados, desenvolvimento de SQL e sua administração. (MYSQL, 2013b)

O MySQL, possui diferentes motores de armazenamento, que são mecanismos para armazenamento de dados. Para o presente trabalho foi utilizada o mecanismo InnoDB. O InnoDB foi desenvolvido especialmente para obter o máximo de performance ao processar um grande volume de dados que provavelmente ganha em eficiência de qualquer outro tipo de banco de dados relacional baseando-se em disco e possui sua própria área de buffer para armazenar dados e índices na memória principal. (MYSQL, 2013c)

1.9 APACHE TOMCAT

Segundo o Devmedia (2013b), o servidor Tomcat, desenvolvido pela fundação Apache é um container *web* de código fonte aberto baseado em Java criado para executar aplicações *web* que utilizam tecnologias Servlets, JSPs e JSFs. É indicado para uso em aplicações *web* e sua principal característica técnica é estar centrada na linguagem de programação Java.

Servlets são classes Java executadas do lado servidor, que trabalham com requisições *Hyper Text Transfer Protocol* (HTTP) que são enviadas por clientes (página *web*) e envio de resposta para essas requisições. Os Servlets possuem ciclos de vida que incluem o seu carregamento, inicialização, atendimento de requisições, finalização e descarregamento. Esse ciclo de vida é gerenciado pelo Servlet Container que também pode ser definido como um módulo de um servidor *web*. (SILVA, 2009)

1.10 LINGUAGEM DE MARCAÇÃO

Segundo Lemay (2002), o *Hypertext Markup Language* (HTML) é uma linguagem de marcação e foi desenvolvida para criar páginas *web*. No documento são acrescentadas marcações denominadas *tags*, onde através delas são estruturadas e que são interpretadas pelo *browser* e exibidas conforme a especificação.

1.11 HYPERTEXT MARKUP LANGUAGE (HTML)

De acordo com Lemay (2002), a linguagem HTML é a abreviação de *Hypertext Markup Language* que significa Linguagem de Marcação de Hipertexto, se baseou na *Standart Generalized Markup Language* (SGML), por sua vez é uma linguagem para a descrição da estrutura de um documento e não sua real apresentação. Seu conceito é que a maior parte dos documentos possua elementos em comum, como por exemplo: títulos, parágrafos ou listas.

Segundo o site w3c (2013), HTML é uma linguagem entendida universalmente por diversos meios de acesso. Em HTML um para publicação de conteúdo como: texto, imagem, áudio, vídeo, etc., na *web* e baseado no conceito de

Hipertexto que são conjuntos de elementos ligados através de uma conexão, formando uma grande rede de informações, mas não são conectados linearmente em forma de texto onde um assunto é ligado ao outro linearmente. A conexão em um hipertexto é feito de forma imprevista, mas permite a comunicação dos dados, organizando conhecimentos e guardando informações relacionadas.

Para Gonçalves (2007), a utilização de hipertexto é utilizada o *Hypermedia Time-base Structuring Language (HYTIME)* onde existe um padrão para representação de hipermídia e informação baseada em tempo.

O HTML possui complexidade inferior ao SGML, onde utiliza um pequeno conjunto de *tags* que provem as necessidades para criação da estrutura das páginas e gerenciamento simples de documentos. (GONÇALVES, 2007)

Conforme o site html.net (2013), *tags* são rótulos usados para indicar ao navegador como deve ser apresentado o *website*. As *tags* são expressas no começo por um sinal de menor “<” e terminam com um sinal de maior “>”, e podem ser de dois tipos, as de abertura: <comando> e as tags de fechamento: </comando>. Tudo que estiver contido entre uma tag de abertura e uma tag de fechamento será processado segundo o comando contido na *tag*.

1.12 CASCADING STYLE SHEETS CSS

Cascading Style Sheets (CSS) é definido como um mecanismo simples para adicionar estilo. Como por exemplo: fontes, cores, espaçamentos, aos documentos web. (W3C,2013)

Segundo Silva (2012), CSS é a abreviação para o termo em inglês *Cascading Style Sheet*, traduzido para o português como folhas de estilo em cascata.

Conforme Freeman e Freeman (2008), as regras CSS são aplicadas aos elementos de uma página, atribuindo um valor para cada atributo que se almeja alterar. Com o emprego das CSS, torna-se plausível a alteração de vários elementos de várias páginas de uma só vez, facilitando a manutenção em casos de alterações. Fora a estilização das páginas web, as CSS tendem facilitar a definição e inserção dos estilos que serão aplicados.

Cabe ao CSS devolver à marcação HTML o seu propósito inicial da linguagem, que foi criada exclusivamente de marcação e estruturação de conteúdo. Conforme seus idealizadores não compete ao HTML fornecer informações ao

agente do usuário sobre apresentação dos elementos como, por exemplo: cores de fontes, tamanhos de textos, posicionamento e todo o aspecto visual do conteúdo. Todas as funções de apresentação de um documento pertencem às regras do CSS. (SILVA, 2012)

1.13 PHP

Segundo Welling e Thomson (2005), o PHP é uma linguagem de criação de *scripts* do lado do servidor que foi projetada especificamente para *web*. Dentro de uma página HTML, é introduzido o código de PHP que será executado sempre que a página for solicitada. O código PHP é decifrado no servidor *web* e gera HTML ou outra saída que o visitante visualizará.

O PHP é a abreviação de *Hypertext Preprocessor* ou em tradução livre: Pré-processador de Hipertexto. É uma linguagem de *script* de código-fonte aberto onde seu principal objetivo é gerar conteúdo dinâmico para páginas *web* incluindo o código HTML junto com a linguagem PHP. Outra vantagem do PHP é que por ser executado no lado do servidor, seu código-fonte não é exibido ao internauta, que apenas terá acesso ao seu HTML resultante. (MELO; NASCIMENTO, 2007)

O PHP é uma das linguagens mais utilizadas na *web*. Existem milhões de *sites* no mundo inteiro utilizando PHP. Mas uma das principais diferenças em relação a outras linguagens é a capacidade do PHP interagir com o mundo *web*, dando dinamismo a *websites*. (NIEDERAUER, 2011)

Conforme Welling e Thomson (2005), o PHP possui muitas características como: eficiente desempenho, com um único servidor de baixo custo é capaz de atender milhões de acessos por dia; integração com os mais confiáveis bancos de dados, com a utilização do *Open Database Connectivity Standard* (ODBC), que vincula a qualquer banco de dados que forneça um *driver* de ODBC, abrangendo produtos *Microsoft* e muitos outros; possui várias bibliotecas integradas, facilitando as instruções de código de programação; custo benefício, sendo uma ferramenta gratuita; fácil aprendizagem na programação, visto como base linguagens Pearl e C; possui recursos orientado a objetos muito bem elaborados, a partir da versão 5 do PHP; quanto a portabilidade o PHP está disponível para muitos sistemas operacionais diferentes, como: Unix gratuitos (Linux, FreeBSD), Unix Comerciais (Solaris, IRIX) ou em versões *Microsoft Windows*; possui código-fonte aberto;

disponibiliza suporte e softwares relacionados numa base comercial, através da Zend Technologies (www zend.com).

1.14 JQUERY

O JQuery é uma biblioteca JavaScript ágil, pequena e possui muitos recursos, com isso tarefas como a passagem de um documento HTML e manipulação de acontecimentos, animação e Ajax se tornam muito mais compreensíveis com a utilização de uma API de simples utilização. Outra vantagem é que a biblioteca opera em uma imensidão de navegadores. (JQUERY, 2014)

1.15 BOOTSTRAP

Bootstrap é um framework front-end de CSS utilizado para a implementação de sistemas web responsivos, ou seja, sites que se adaptam a diferentes tamanhos de tela, se ajustando ao aparelho do usuário. Nos dias de hoje, o Bootstrap é o framework mais usado na internet para esta finalidade. Em termos de responsividade no layout, o Bootstrap desenha seu projeto para ser apresentado em telefones, tablets ou desktops, através de uma única base de código, facilitando assim, a mudança de plataformas e facilitando o suporte do código. (BOOTSTRAP, 2015)

1.16 AJAX

Segundo o Devmedia (2013c), o AJAX é uma ferramenta de desenvolvimento que gera um novo padrão de execução, nesse novo padrão, o navegador deixa de ser um “Terminal Burro” e se torna um container de aplicação, em outras palavras, o navegador para de mostrar o conteúdo e começa a monitorar as ocorrências, executar scripts e remodelar o layout da página. Atualmente várias tecnologias são utilizadas em sincronia possibilitando excelentes efeitos.

1.17 PHPMYADMIN

O phpMyAdmin é um software livre e de código aberto criado com a linguagem PHP e tem como primordial atividade a gerenciar o banco de dados MySql. Essa ferramenta possibilita a criação, alteração e exclusão de banco de dados, tabelas, chaves primárias e estrangeiras, realizar comando SQL, e outras funcionalidades. O usuário também terá acesso a estatísticas, como quantos comandos foram executados, tráfico de rede mostrando a quantidade de bytes enviados e recebidos pelo servidor e outras opções. (PHPMYADMIN,2011)

1.18 CONSIDERAÇÕES SOBRE O CAPÍTULO

Conforme foi visto, neste capítulo observou-se de forma mais basilar demonstrar, proporcionar e explanar as tecnologias que foram empregadas no desenvolvimento desta monografia. A linguagem Java, assim como as demais ferramentas, são de suma importância para o sucesso do projeto e cada uma com suas características peculiares facilitam o desenvolvimento dessa aplicação.

No próximo capítulo serão abordados detalhes sobre a importância do controle de estoque e o papel efetivo que um software desempenha em uma empresa.

2 ANÁLISE E PROJETO DE SISTEMA

No presente capítulo serão abordadas as fases de desenvolvimento do software de controle de estoque de veículos, além de elencar os elementos que compõem do processo organizacional da empresa e apresentar os procedimentos que serão informatizados. Para isso, inicialmente, será descrito a funcionalidade de um controle de estoque. Esta fase será muito importante para entendimento do trabalho, e irá aprimorar o desenvolvimento do projeto.

2.1. ANÁLISE DE NEGÓCIO

2.1.1. Estoque

De acordo com Dias (1993), o estoque de uma corporação é todo material físico confeccionado na etapa de fabricação ou necessário para a fabricação de forma direta ou indireta, que são guardados por um certo espaço de tempo.

O estoque pode ser definido como “o conjunto de bens armazenados, com características próprias, e que atende às necessidades da empresa”. (MOURA 2004, p.)

De acordo com Fortunado (2004), o monitoramento do estoque, dentro de uma empresa, se torna cada vez mais importante, considerando o mundo moderno em que vivemos. Nessas circunstâncias, se ocorrer algum lucro, isso fatalmente irá refletir no seu estoque, através de métodos de controle adequados.

Conforme Wanke (2008) existem duas características que são muito importantes se tratando de controle de estoque. Uma delas é reação a demanda, onde a partir de um controle de estoque equilibrado, a empresa consegue controlar os gastos com estoque de acordo com a necessidade de seus clientes. Outra característica notável que deve ser trabalhada no controle de estoque é o planejamento com base em previsões, onde a empresa traça metas de vendas e compras, considerando o estoque atual, os produtos mais consumidos e mais procurados.

Ainda para Wanke (2008), existem algumas funções que devem obter maior destaque no momento de controlar um estoque. Um exemplo é o estoque em nosso

poder, ou seja, o número máximo de cada item, que empresa tem interesse em manter estocado.

Outro fator que deve ser levado em conta é o tempo de utilização, isto é, a quantidade de dias em que o produto vai ser utilizado e a assiduidade com que esse material é empregado. Também é importante levar-se em consideração os custos de armazenamento, investimentos, compras, movimentação, e de transportes, ou seja, todos os custos que direta ou indiretamente influem na aquisição de um item em estoque. Deve-se notar ainda as condições de mercado e tendência dos preços, ou seja, se o mercado contém pessoas que possam adquirir ou comercializar produtos levando em conta com altas e baixas empresariais, isto é, a conveniência que o item proporciona ao utilizador.

Para o estoque de carros em uma garagem, é imprescindível que o estoque seja muito bem controlado devido a uma série de fatores como a grande rotatividade. Os veículos se desvalorizam rapidamente, e também são pagas taxas anuais como IPVA, Licenciamento, DPVAT. Outro fato importante que deve ser destacado é o fato de que os veículos são comprados e não são transferidos para a empresa, dessa forma, os antigos donos, também podem cobrar a maior agilidade na transferência.

A partir dessas informações, pode-se perceber que uma empresa que comercializa automóveis, necessita urgentemente de um sistema que faça o controle de seu estoque e que norteie suas tomadas de decisões, tornado a empresa mais dinâmica e suas rotinas mais eficazes.

2.1.2 Instrução do problema

Atualmente a empresa Amigão Veículos não faz uso de computadores em suas rotinas referentes ao controle de veículos. Todo processo é controlado através de planilhas ou documentos de papel, que são impressos em gráficas. As informações referentes aos veículos que são adquiridos são preenchidas na frente de uma ficha, chamada Ficha Cadastral, conforme Figura 2.1.

FICHA CADASTRAL		
Marca	Cor	Ano
<hr/>		<hr/>
Chassi		Renavam
<hr/>		<hr/>
Cidade		Placa
<hr/>		<hr/>
PROPRIETÁRIO		
Nome _____		
End. _____		
CIC _____		Fone _____
RG _____		CEL. _____
ADQUIRIDO DE		
Nome _____		
End. _____		
CIC _____		Fone _____
RG _____		CEL. _____
VENDIDO PARA		
Nome _____		
End. _____		
CIC _____		Fone _____
RG _____		CEL. _____
D A T A		VALORES
Compra _____		Compra _____
Venda _____		Despesas _____
		Total _____
		Venda _____
		Lucro _____

Figura 2.1 – Ficha Cadastral (Frente) 1
Fonte: Elaborado pelos autores, 2013.

A photograph of a blank ledger page. At the top center, the word "DESPESAS" is printed in bold capital letters. Below it is a grid of horizontal lines for writing entries. There are approximately 20 lines in total, with a slightly larger space at the top and bottom.

Figura 2.2 – Ficha Cadastral (Verso) 1

Fonte: Elaborado pelos autores, 2013.

No verso da ficha, figura 2.2 são cadastrados os gastos que foram realizados pela empresa para deixar o veículo no padrão que agrade o cliente (por exemplo: lavagem, polimento, multas, etc.). Essas despesas são acrescidas ao valor que foi pago no automóvel e são levadas em consideração no momento de calcular o valor de venda do veículo.

Os clientes também são cadastrados em fichas manuais, chamadas Fichas Cadastrais de Clientes, Figura 2.3 que recebem seus dados, e posteriormente são consultados para verificação de possíveis restrições ou problemas que possam indeferir a compra do veículo.

O Amigão Veículos		FICHA CADASTRAL DE CLIENTE PESSOA FÍSICA				
VENDEDOR LEANDRO PEROSSO	FONE / CELULAR (18) 3652-2156 / 99162-5352 / 9673-0856		FAX (18) 3652-3590	EMAIL		
DADOS PESSOAIS						
NOME COMPLETO		CPF		RG		
DATA NASCIMENTO		EXPEDIÇÃO DO RG		LOCAL DE NASCIMENTO	UF	ESTADO CIVIL
NOME DO PAI		NOME DA MÃE				
DADOS RESIDENCIAIS						
ENDEREÇO RESIDENCIAL				NÚMERO	COMPLEMENTO	
BAIRRO		CIDADE		UF	CEP	TEMPO RESID.
TELEFONE (DDD + NÚMERO)		CELULAR (DDD + NÚMERO)		RESIDÊNCIA (X) PRÓPRIA () ALUGUEL		
DADOS PROFISSIONAIS						
EMPRESA EM QUE TRABALHA				OCCUPAÇÃO PROFISSIONAL (CARGO/FUNÇÃO)		
RENDAS		DATA ADMISSÃO		CNPJ (QUANDO FOR SÓCIO)		TELEFONE CONTADOR
ENDEREÇO COMERCIAL (LOGRADOURO)				NÚMERO	COMPLEMENTO	
BAIRRO		CIDADE		UF	CEP	
TELEFONE (DDD+NÚMERO)		RAMAL		CONTATO		
DADOS CONJUGE						
NOME COMPLETO				CPF	RG	
DATA NASCIMENTO		NACIONALIDADE		LOCAL DE NASCIMENTO		UF
NOME DO PAI				NOME DA MAE		
EMPRESA EM QUE TRABALHA			DATA ADMISSÃO		CNPJ (QUANDO FOR SÓCIO)	
ENDEREÇO COMERCIAL (LOGRADOURO)				NÚMERO	COMPLEMENTO	
BAIRRO		CIDADE		UF	CEP	
TELEFONE (DDD+NÚMERO)		RAMAL		FONE DO CONTADOR		
REFERENCIA BANCÁRIA E PESSOAL						
BANCO(S)		AGENCIAS		Nº CONTA CORRENTE	CLIENTE DESDE	
REF. PESSOAL - NOME		TELEFONE		REF. PESSOAL - NOME		TELEFONE
DADOS FINANCIAMENTO						
VALOR DE COMPRA DO VEICULO		ENTRADA		VALOR DO CRÉDITO FINANCIADO		
VALOR DA TAC		COEFICIENTE	TABELA N.	PRAZO	CARENÇIA	
VALOR DA PARCELA			VEÍCULO / MODELO			
MARCA		ANO		MODELO	PLACA	RENAVAM
CHASSI		COR			COMBUSTIVEL	
OBSERVAÇÕES						

Figura 2.3 – Ficha Cadastral de Cliente
Fonte: Elaborado pelos autores, 2013

Também existe na empresa uma ficha chamada Ficha Compra/Venda/Troca/Financia, Figura 2.4 que elenca os dados de todos os automóveis que se encontram no pátio da empresa, ou seja, essa ficha controla o estoque da empresa. Se um veículo for vendido, ele é automaticamente riscado da relação.

Fonte: Elaborado pelos autores, 2013

A solução proposta no presente trabalho é informatizar todo o processo de gerenciamento, tornando-o mais dinâmico, seguro e eficiente o funcionamento na empresa.

Com a utilização de computadores e um software que organize todo procedimento de controle de estoque, vários problemas que ocorrem atualmente serão sanados, controle de clientes; busca mais rápida e precisa.

O método arcaico que é utilizado pela empresa torna a mesma defasada perante seus concorrentes que já utilizam algum sistema informatizado. Esse atraso acaba refletindo nos funcionários, que têm mais dificuldade para desempenhar suas tarefas diárias utilizando a mesma rotina várias vezes.

2.1.3 Atores e envolvidos no processo

A empresa é formada por três funcionários, o gerente, o vendedor e o comprador, cada um tem uma função específica.

O gerente é o responsável pela parte burocrática, como consultar multas e licenciar veículos, administrar contas a pagar e contas a receber, também é responsável pela última negociação e sacramentar os negócios.

Ao vendedor, que é um funcionário interno da empresa, cabe o papel de fazer o primeiro contato com o público, apresentando o veículo, buscando fechar o negócio que no momento oportuno será finalizado pelo gerente.

Outro funcionário que tem um papel importante é o comprador que atua nas ruas em busca de boas oportunidades de negócios. Quando necessário auxilia o vendedor, já que também domina o ramo de vendas de veículos.

2.2 VISÃO GERAL DO PRODUTO

2.2.1 Perspectiva do Produto (Análise de Mercado)

O projeto que está sendo desenvolvido terá como principais características a gravação de dados criados pelos processos administrativos que foram citados anteriormente, visando melhorar o desempenho, a segurança e organização da empresa. Um fato importante que merece ser observado é que o software está

sendo elaborado para a empresa Amigão Veículos, mas poderá ser empregado em outras empresas do mesmo segmento, ou até mesmo em outros tipos de negócios, desde de que seja previamente adaptado para essa mudança.

Entre as peculiaridades que o projeto monográfico não possui estão a gestão financeira, a integração com sites de publicidade e redes sociais, adaptação a legislação, emissão de notas fiscais, etc. Ressaltando que o trabalho está sendo desenvolvido, já preparado para receber essas funcionalidades no futuro.

Uma funcionalidade que difere o software que está sendo desenvolvido é a funcionalidade de acesso a dispositivos móveis; com ela será possível que o comprador de veículos da empresa melhore o desempenho de seus serviços e ainda possa fazer divulgação dos produtos da empresa já que poderá expor seus produtos através do acesso on-line ao sistema, que futuramente será implementada a inclusão de fotos dos veículos para análise.

2.2.2 Premissas e Dependências

A revendedora de veículos aspira um programa que controle todos os procedimentos de clientes, cadastro de fornecedores, cadastro de despesas, gerenciamento de compra, manutenção e venda de veículos, ou seja, tudo que tenha ligação aos automóveis da empresa, desde o primeiro momento (aquisição), até o último instante (venda).

Para se obter um salto de qualidade nas rotinas da empresa, além do desenvolvimento do software também será necessário adquirir pelo menos um computador com a seguinte configuração: Processador Intel Core i3, Memória RAM de 4 GB ou mais, HD de 500 ou mais, uma impressora multifuncional, HD externo para backup, assinatura de um plano de internet, parceria com empresas que disponibilizam crédito a compradores de veículos, acesso ao sistema de consulta de multas e outras informações relevantes sobre os veículos que serão comprados e compra de licença de serviços de consulta sobre a situação de clientes e cheques, para não concretizar negócios com pessoas que tenham restrições em seu nome.

2.2.3 Características

Necessidade	Riscos
Cadastro de Clientes	Alta
Cadastro de Veículos	Alta
Cadastro de vendedores	Alta
Acesso a dispositivos móveis	Alta
Gastos com Veículos	Alta
Movimento de entrada	Alta
Movimento de saída	Alta

Tabela 2.1 - Necessidades/Riscos

Fonte: Elaborada pelos autores, 2016.

Na tabela 2.1, observa-se as necessidades e riscos para o desenvolvimento do sistema.

Neste capítulo a finalidade foi exibir e explanar as tecnologias utilizadas no projeto. Também foi realizada uma abordagem sobre o funcionamento atual da empresa e como o sistema poderá a melhorar o desempenho e confiabilidade das rotinas diárias.

No próximo capítulo serão apresentados os diagramas utilizados para desenvolver o sistema de controle de estoque de veículos.

3 ANÁLISE E MODELAGEM DO SISTEMA

O terceiro capítulo será responsável por apontar os passos necessários de análise e projeção do desenvolvimento do sistema que fará o controle das vendas de veículos.

Nesta parte serão abordados os requisitos, estes, serão de fundamental importância para o entendimento das funcionalidades que compõem o desenvolvimento do software para controle de veículos.

3.1 ANÁLISE E PROJETO

3.1.1 Diagrama de Caso de Uso, visão geral

O caso de uso representará as funcionalidades em comum, os atores que fazem parte do sistema. O administrador possui permissão total no sistema, entre suas atribuições estão cadastrar um novo Atendente e inserir novos serviços. O Atendente, figura 3.1 é responsável por cadastrar Cliente, Funcionário, Veiculo, Modelo, Marca, Despesas, Entrada e Saída de veículos.



Figura 3.1 – Diagrama de Caso de Uso Cliente
Fonte: Elaborado pelos autores, 2016.

3.2 DIAGRAMA DE CLASSE

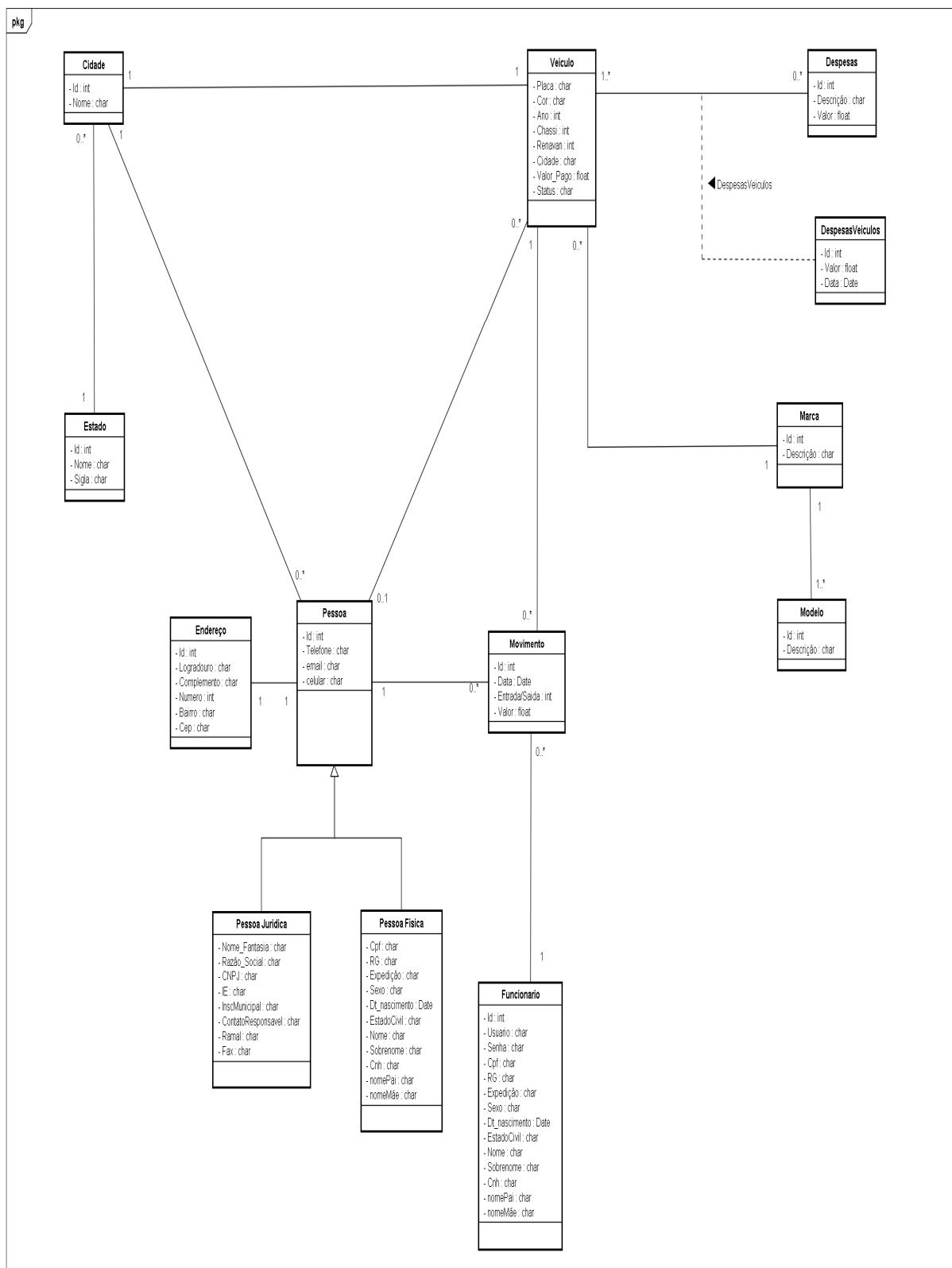


Figura 3.2 – Diagrama de Classe

Fonte: Elaborado pelos autores, 2016.

3.3 PROJETO DE BANCO DE DADOS

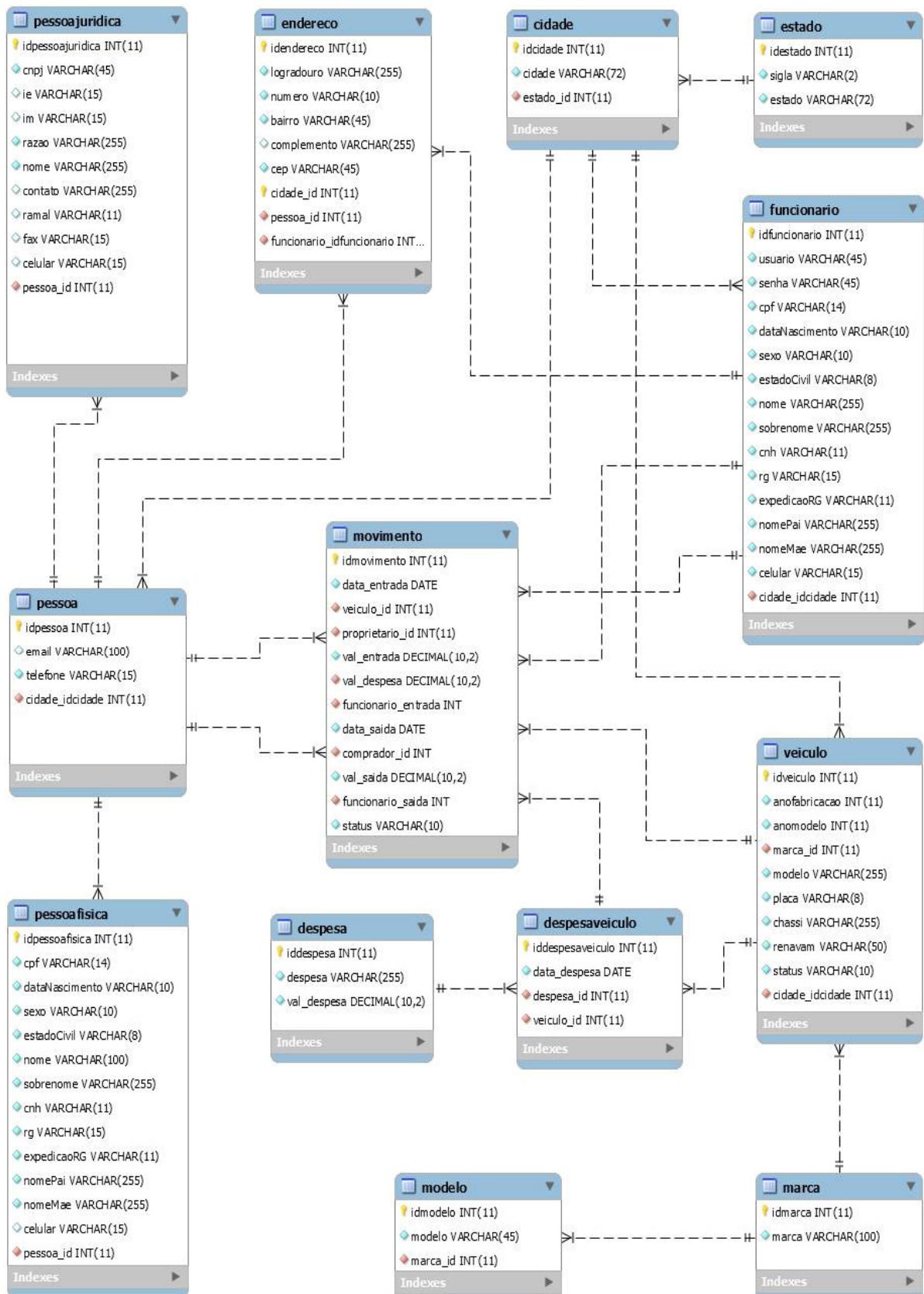


Figura 3.3 – Projeto de banco de dados
Fonte: Elaborado pelos autores, 2016.

Nota-se na figura 3.2, a representação do Diagrama de Classe, observando um conjunto de classes e seus relacionamentos, facilitando a visualização do sistema em questão.

3.4 DIAGRAMA DE ATIVIDADES

Diagramas de atividade são reproduções gráficas que demonstram o andamento de atividades abrangendo as fases de um único processo, os diagramas são empregados para realizar a modelagem de aspectos ativos do sistema. (DUARTE, 2012).

Neste tema serão mostrados todos os diagramas de atividade inerentes ao projeto, que expõem os caminhos que serão percorridos para a conclusão de uma atividade no sistema.

3.4.1 Manter Pessoa

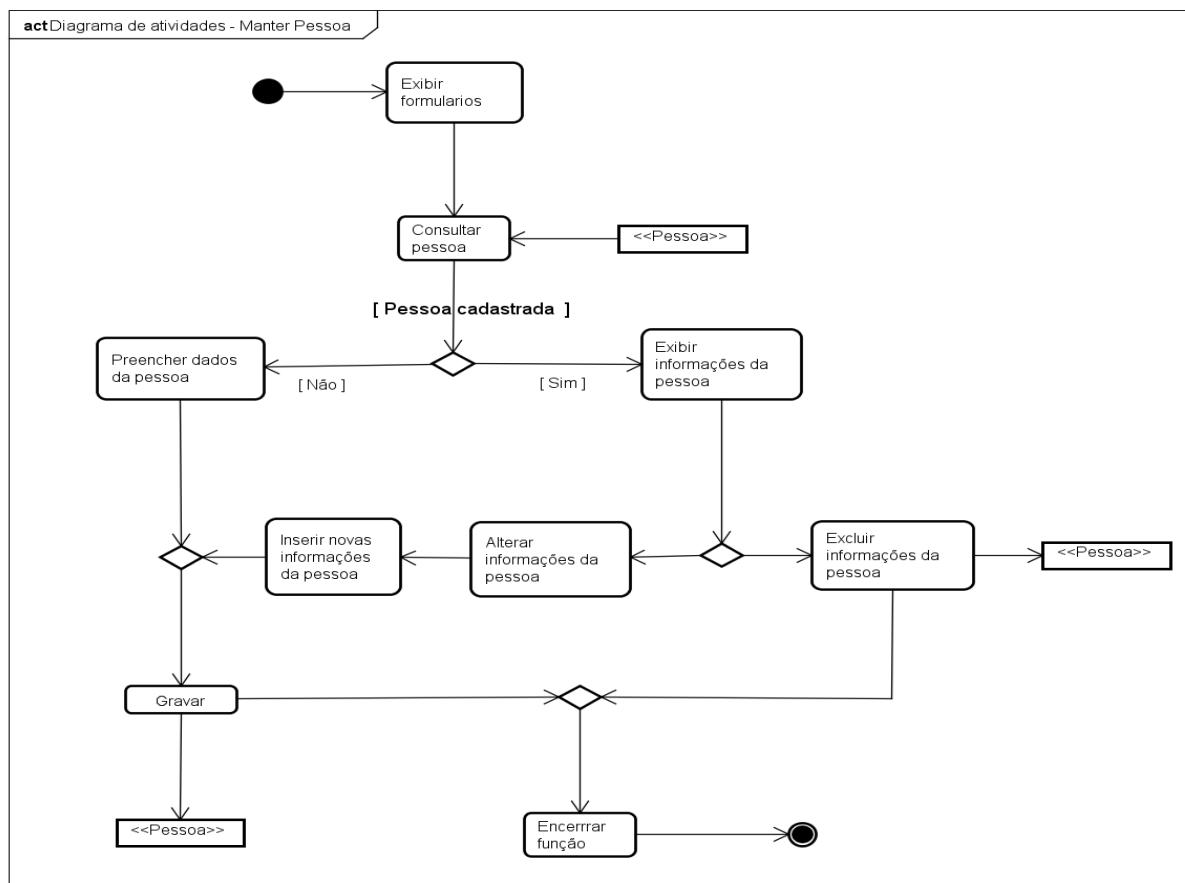


Figura 3.4 – Diagrama de Atividade Manter Pessoa
Fonte: Elaborado pelos autores, 2016.

3.4.2 Manter Funcionário

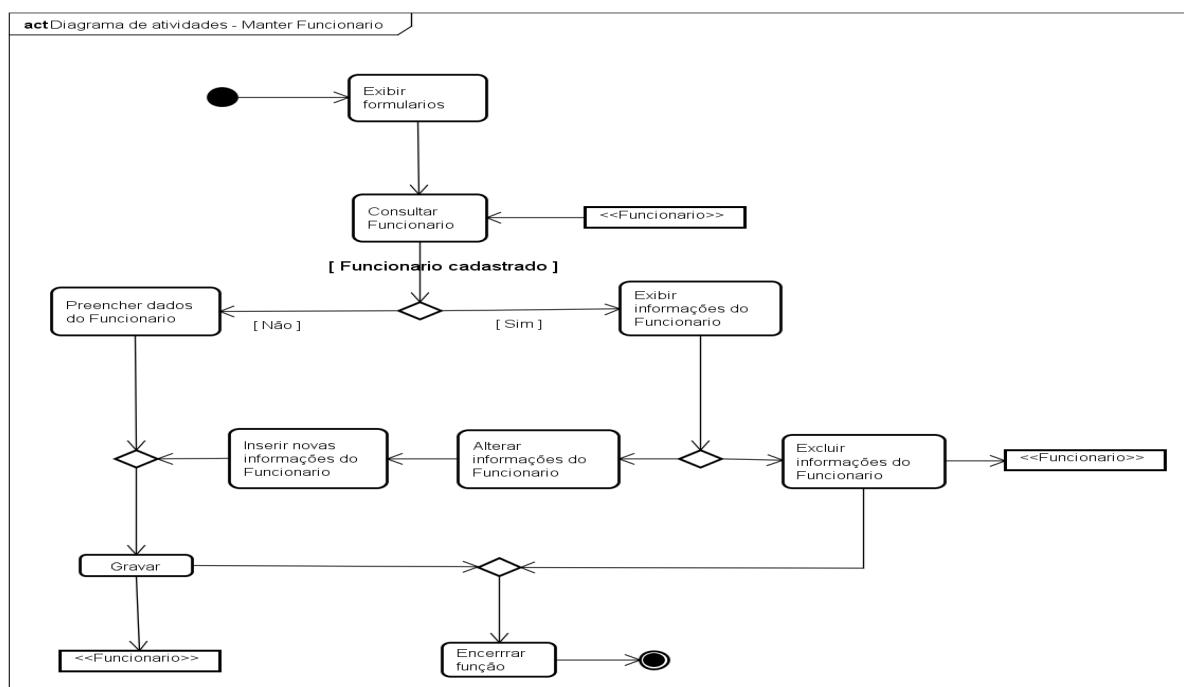


Figura 3.5 – Diagrama de Atividade Manter Funcionário

Fonte: Elaborado pelos autores, 2016.

3.4.3 Login

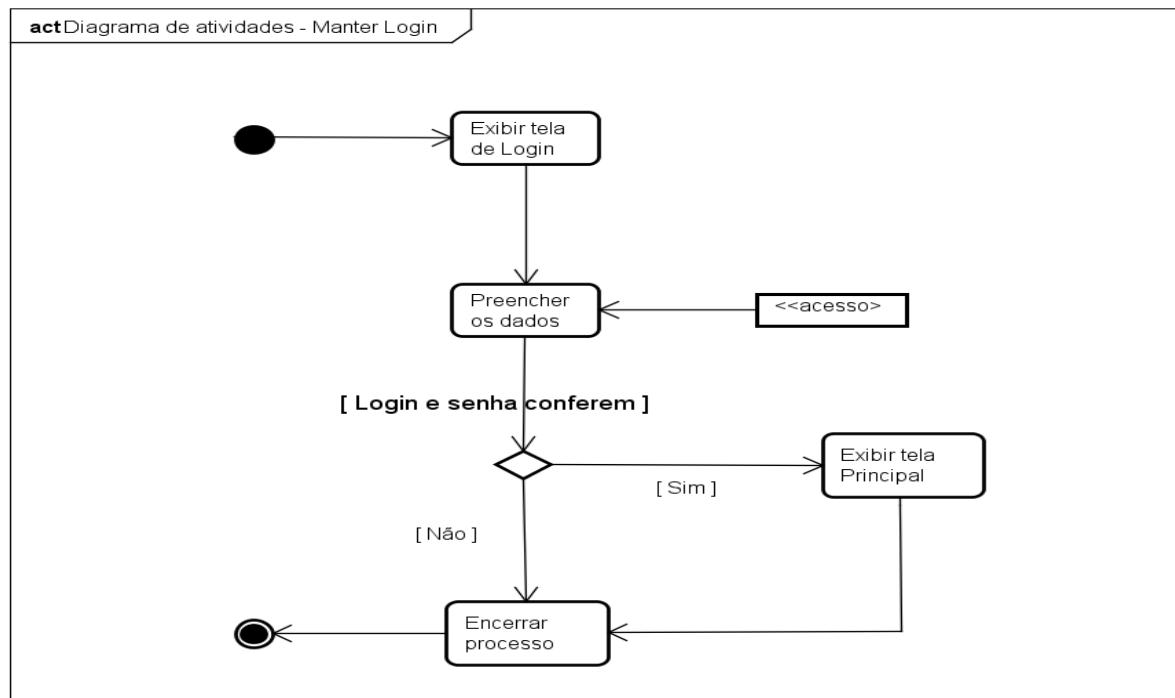


Figura 3.6 – Diagrama de Atividade Login

Fonte: Elaborado pelos autores, 2016.

3.4.4 Manter Marca

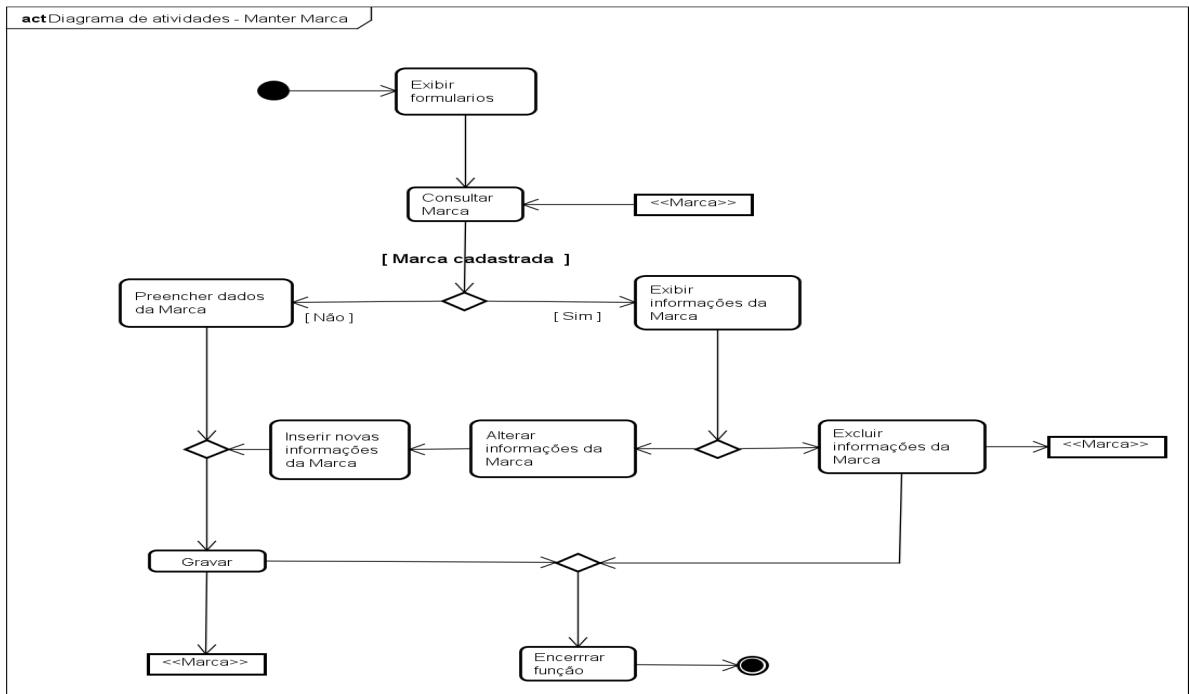


Figura 3.7 – Diagrama de Atividade Manter Marca

Fonte: Elaborado pelos autores, 2016.

3.4.5 Manter Veículos

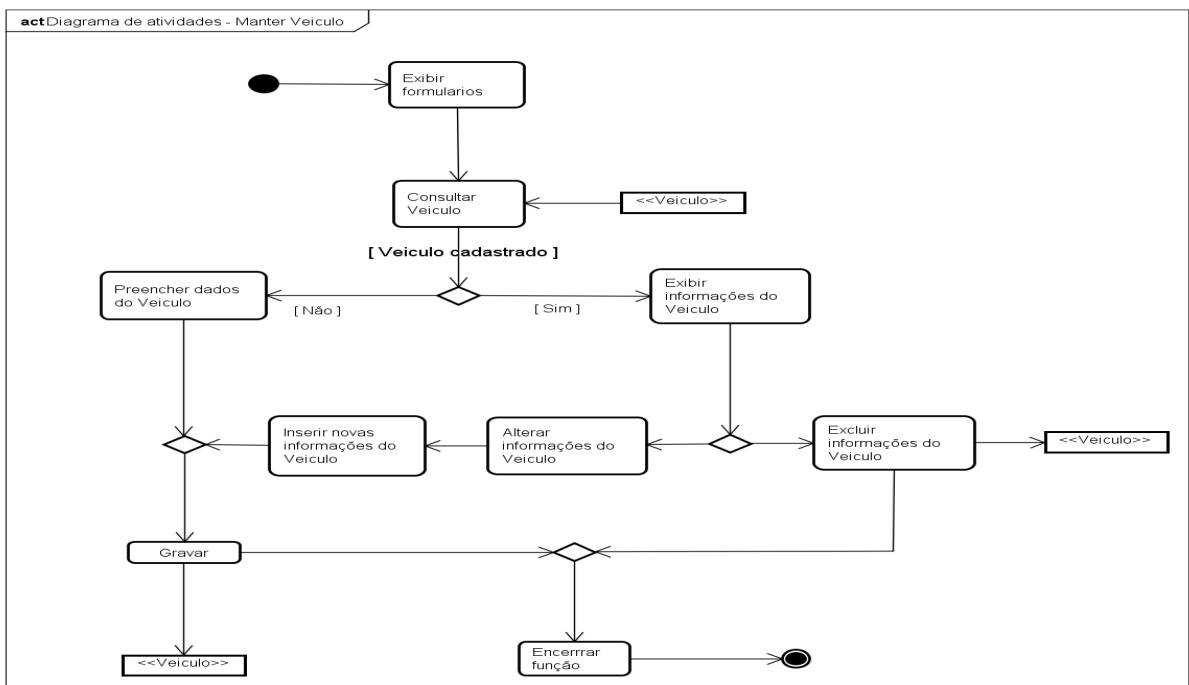


Figura 3.8 – Diagrama de Atividade Manter Veículos

Fonte: Elaborado pelos autores, 2016.

3.4.6 Manter Modelo

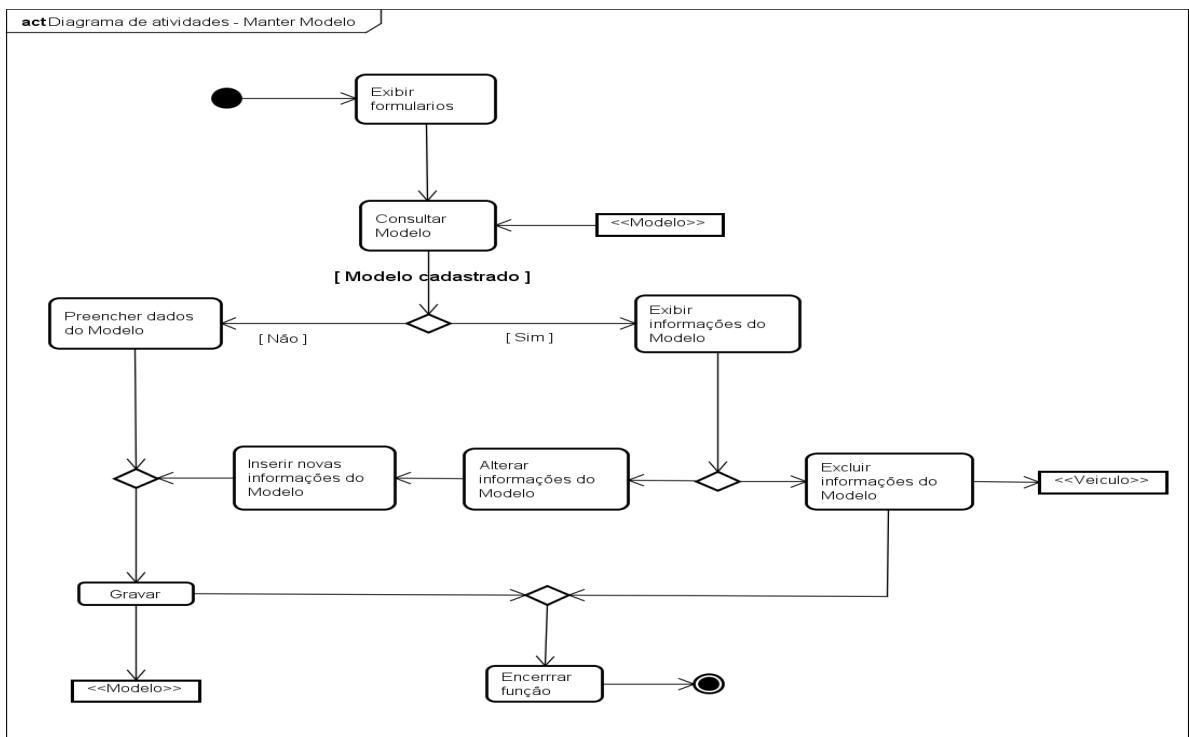


Figura 3.9 – Diagrama de Atividade Manter Modelo
Fonte: Elaborado pelos autores, 2016.

3.4.7 Manter Movimento

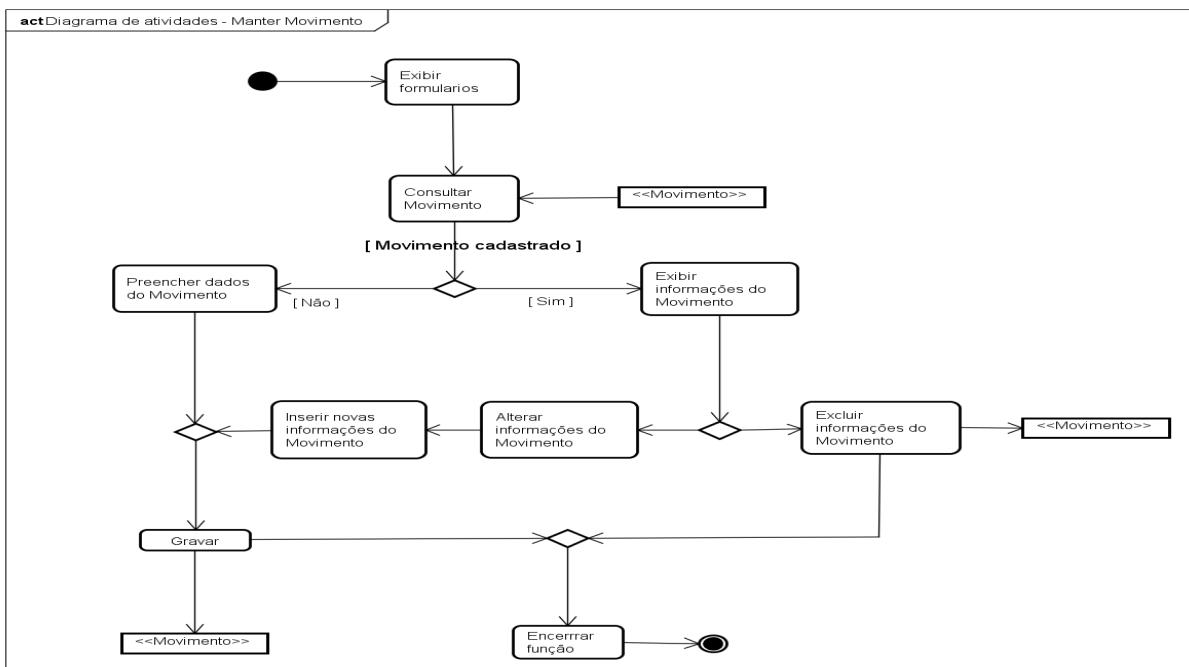


Figura 3.10 – Diagrama de Atividade Manter Movimento
Fonte: Elaborado pelos autores, 2016.

3.4.8 Manter Despesas

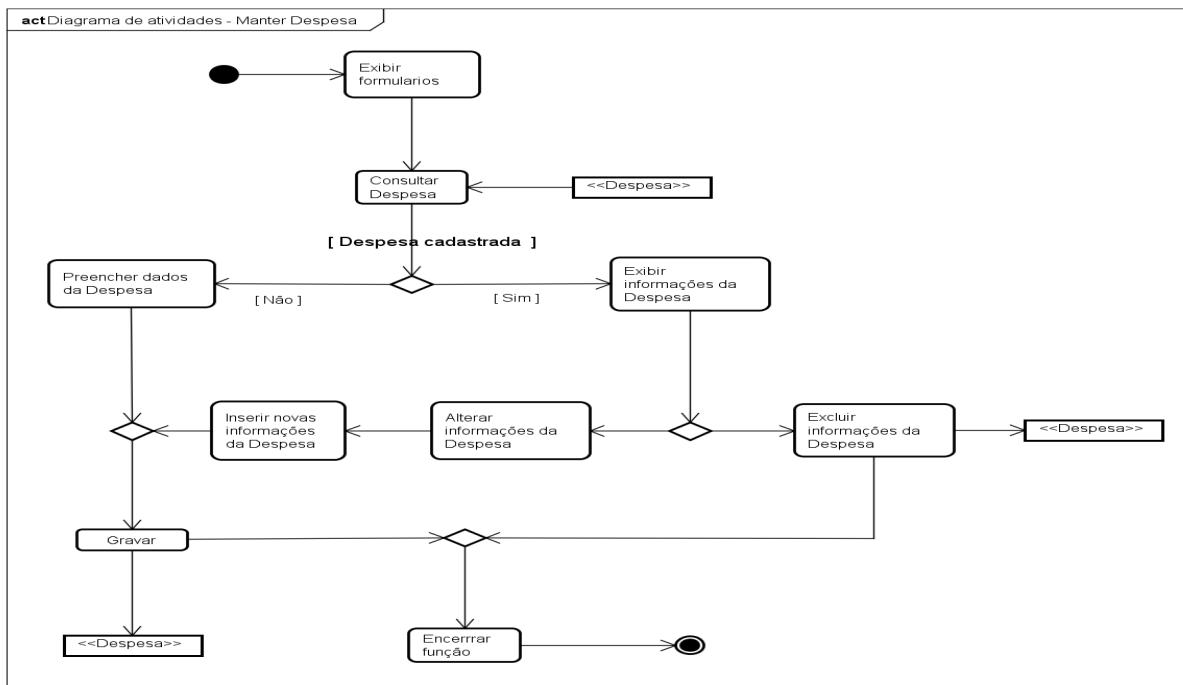


Figura 3.11 – Diagrama de Atividade Manter Despesas

Fonte: Elaborado pelos autores, 2016.

3.4.9 Manter Despesas do Veículo

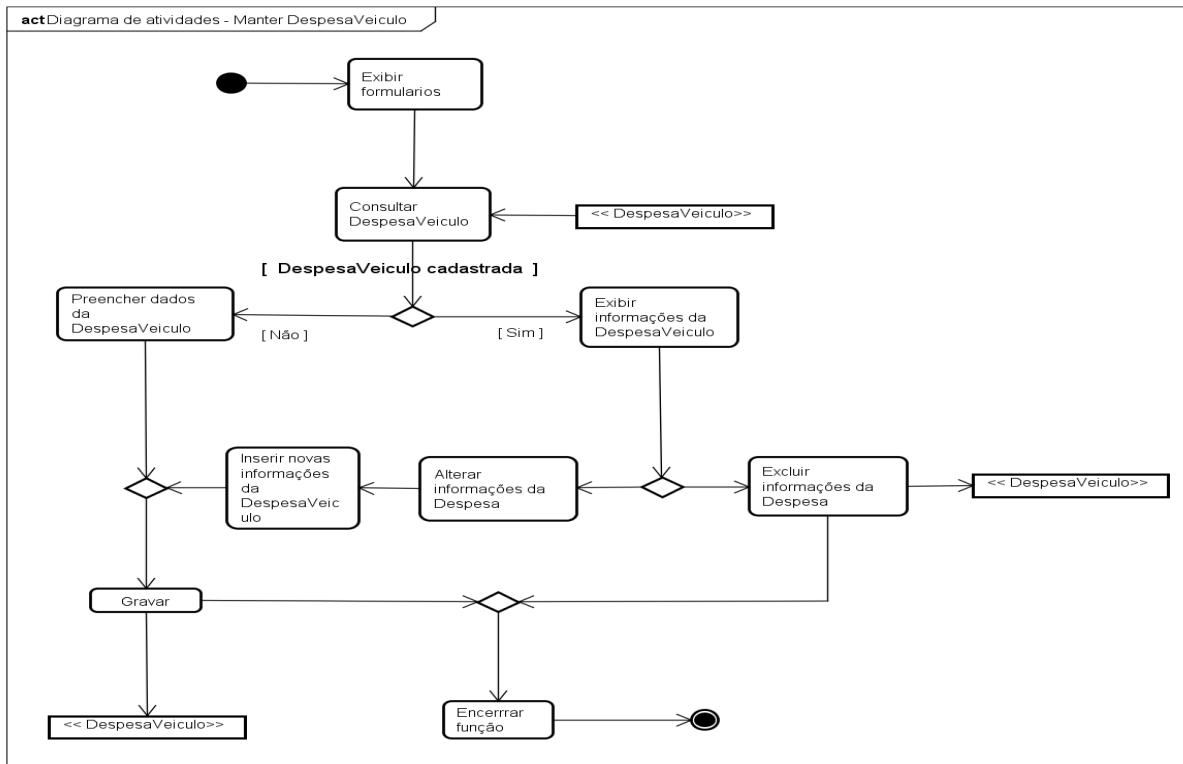


Figura 3.12 – Diagrama de Atividade Manter Despesas do Veículo

Fonte: Elaborado pelos autores, 2016.

3.5 MVC

O MVC é um modelo de arquitetura para criação de softwares que possui como alvo principal a separação das regras de negócio e as interfaces em diferentes camadas, facilitando assim a manutenção e a adição de novos componentes no software. (GAMA, 2011).

O padrão MVC possui três elementos que são chamados de Modelo, Visão e Controlador. A camada denominada modelo é o local onde são definidas as regras de negócio, determinar as validações, busca a banco de dados, entre outros. Na visão, os dados são inseridos e mostrados. O controlador tem a função de receber as informações da camada visão e repassá-las para a camada modelo, sem qualquer regra de negócio criada. (GAMA, 2011).

3.5.1 Manter Pessoa

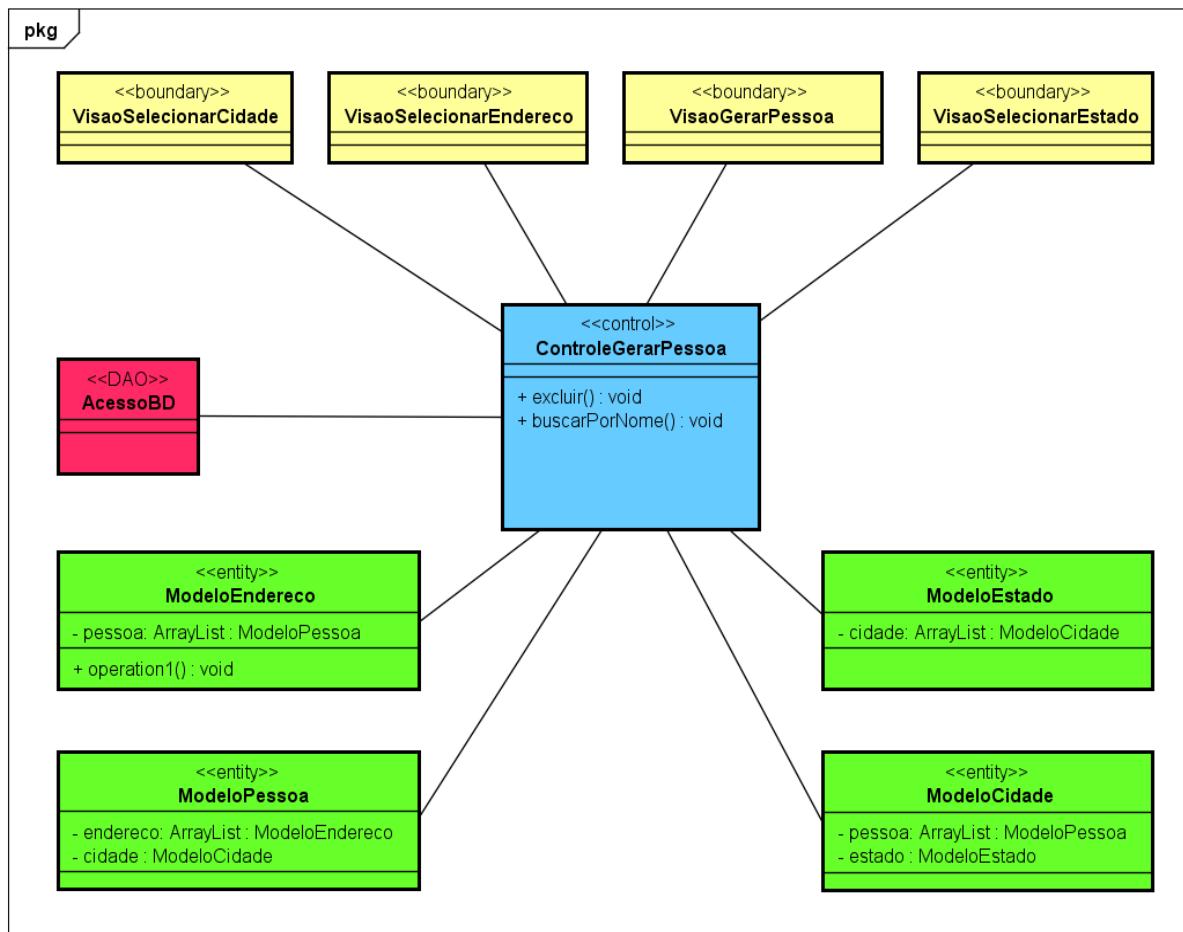


Figura 3.13 – Diagrama de MVC Manter Pessoa
Fonte: Elaborado pelos autores, 2016.

3.5.2 Manter Despesas

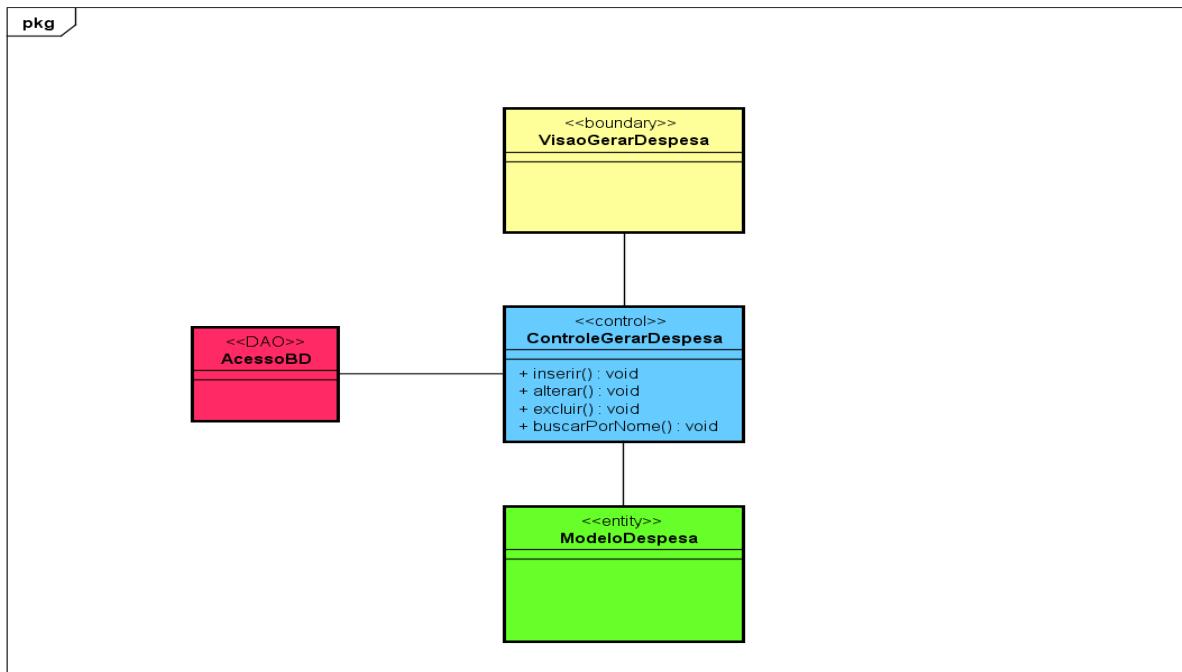


Figura 3.14 – Diagrama de MVC Manter Despesas

Fonte: Elaborado pelos autores, 2016.

3.5.3 Manter DespesasVeiculo

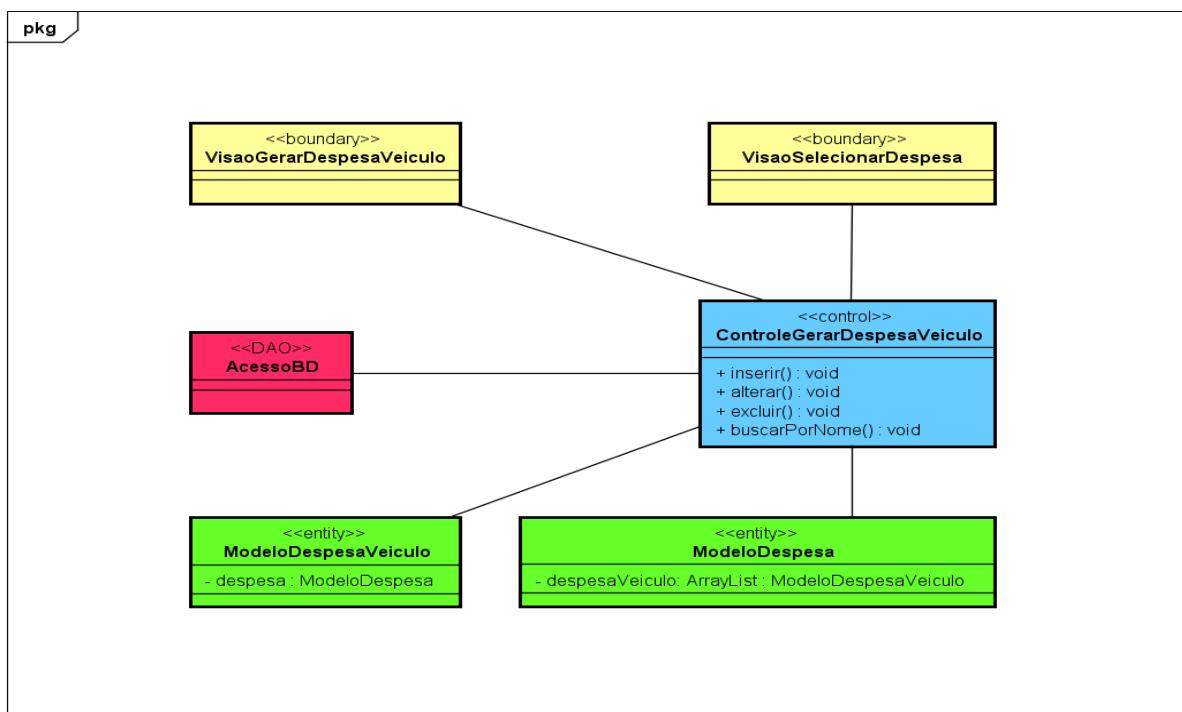


Figura 3.15 – Diagrama de MVC Manter DespesasVeiculo

Fonte: Elaborado pelos autores, 2016.

3.5.4 Manter Funcionário

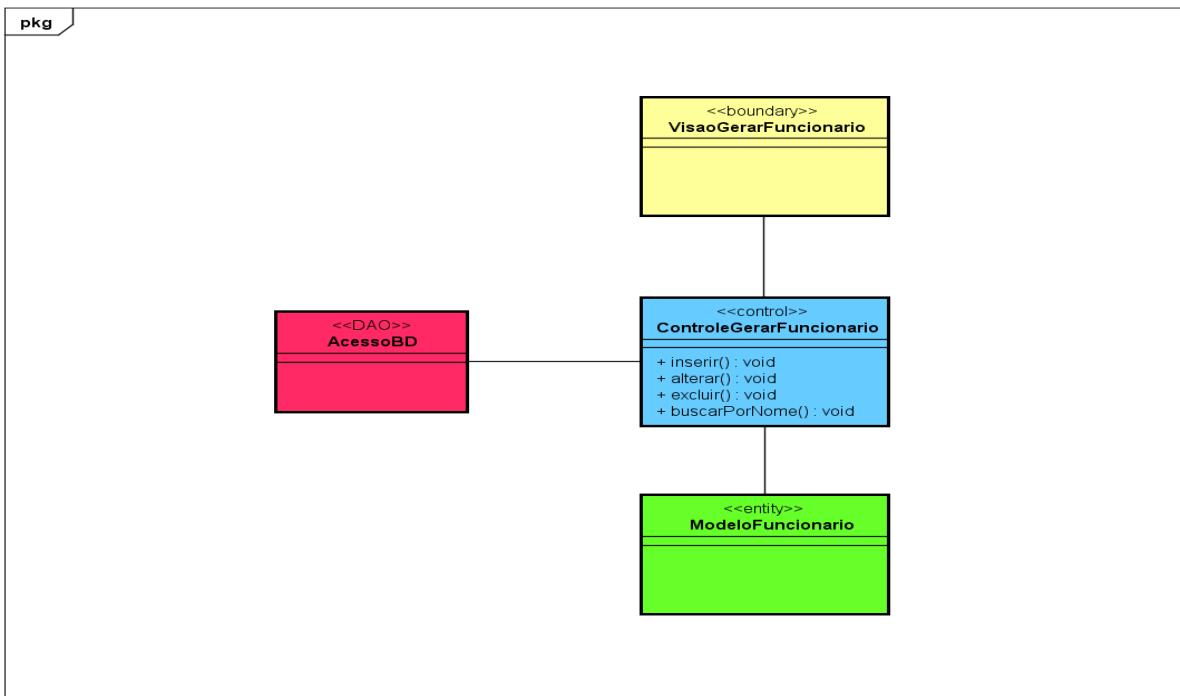


Figura 3.16 – Diagrama de MVC Manter Funcionário

Fonte: Elaborado pelos autores, 2016.

3.5.5 Manter Login

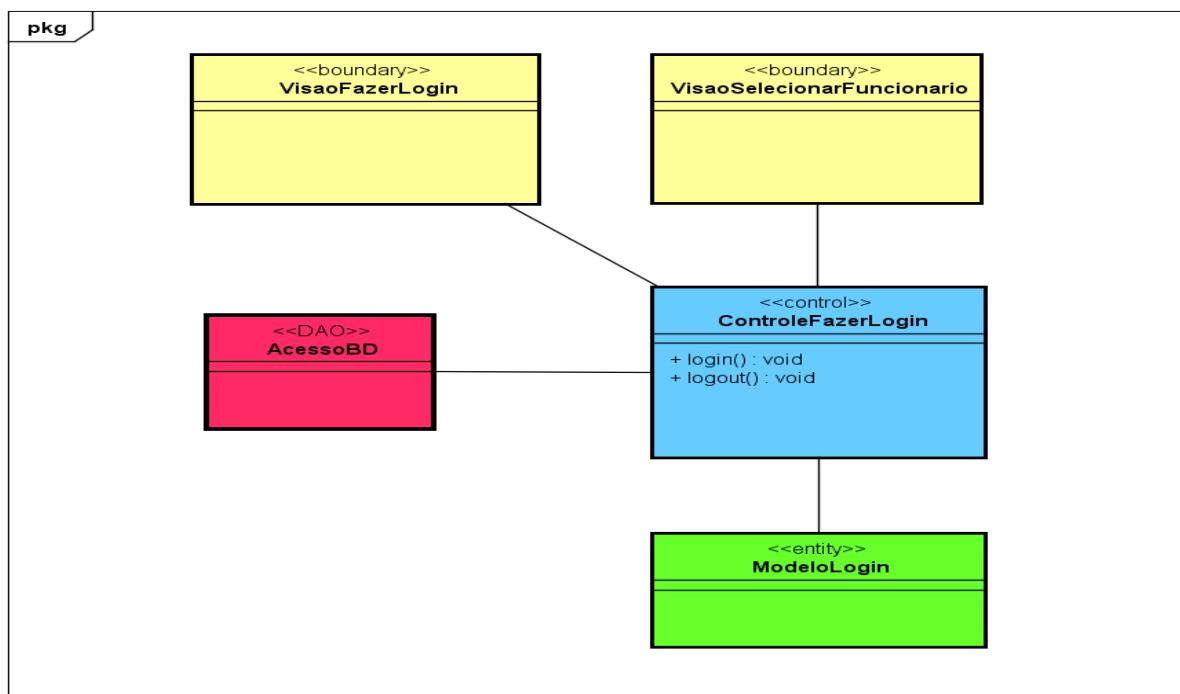


Figura 3.17 – Diagrama de MVC Manter Login

Fonte: Elaborado pelos autores, 2016.

3.5.6 Manter Movimento

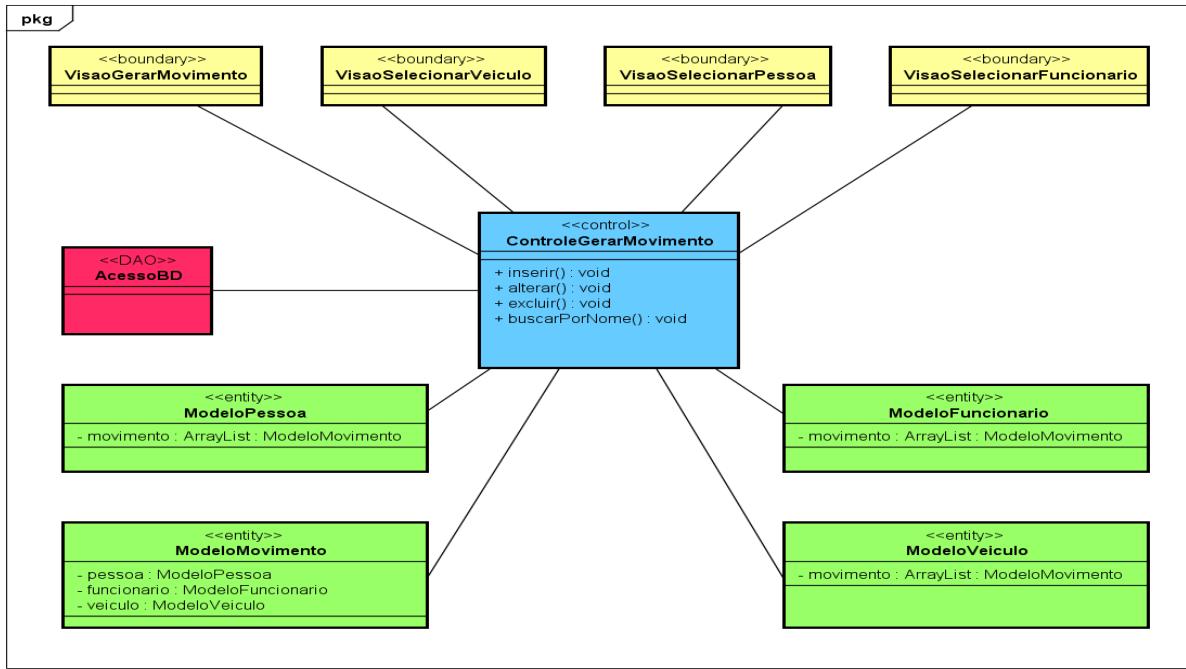


Figura 3.18 – Diagrama de MVC Manter Movimento

Fonte: Elaborado pelos autores, 2016.

3.5.7 Manter Marca

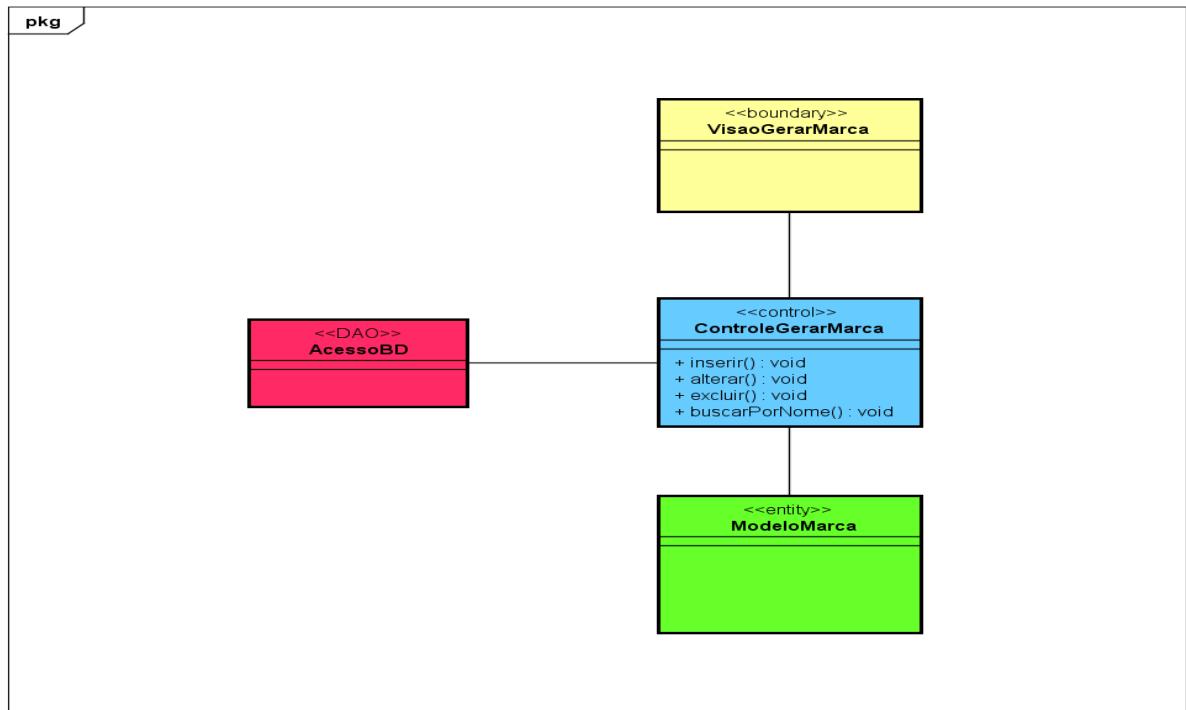


Figura 3.19 – Diagrama de MVC Manter Marca

Fonte: Elaborado pelos autores, 2016.

3.5.8 Manter Veiculo

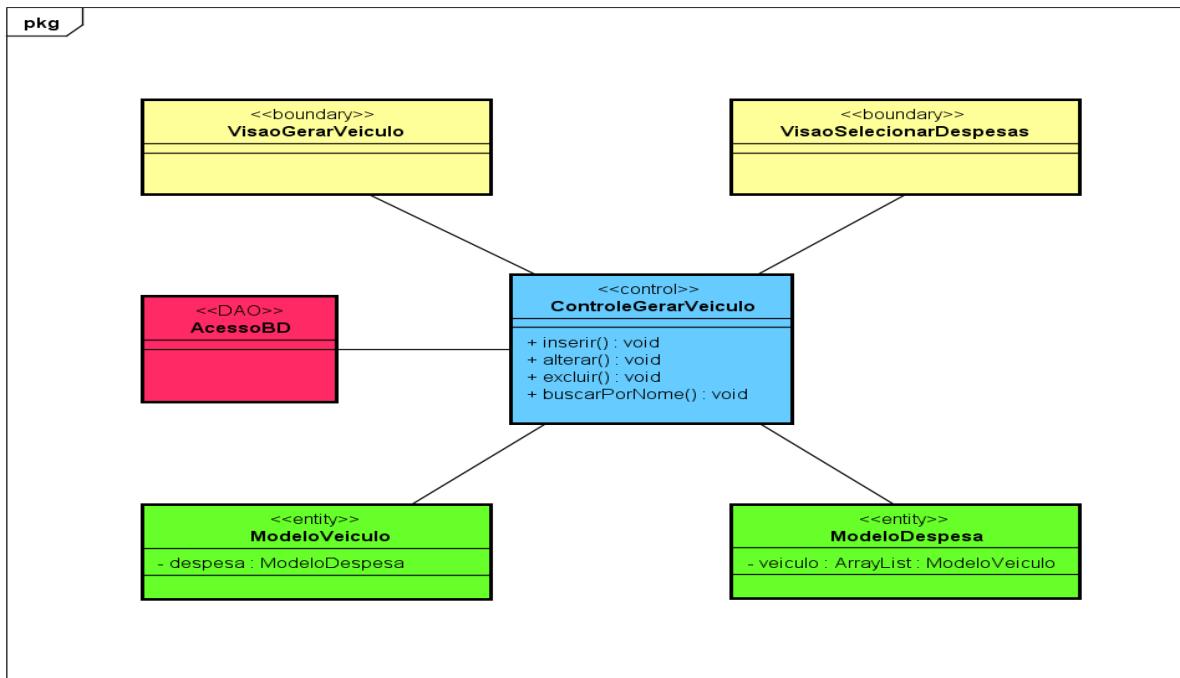


Figura 3.20 – Diagrama de MVC Manter Veiculo

Fonte: Elaborado pelos autores, 2016.

3.5.9 Manter Modelo

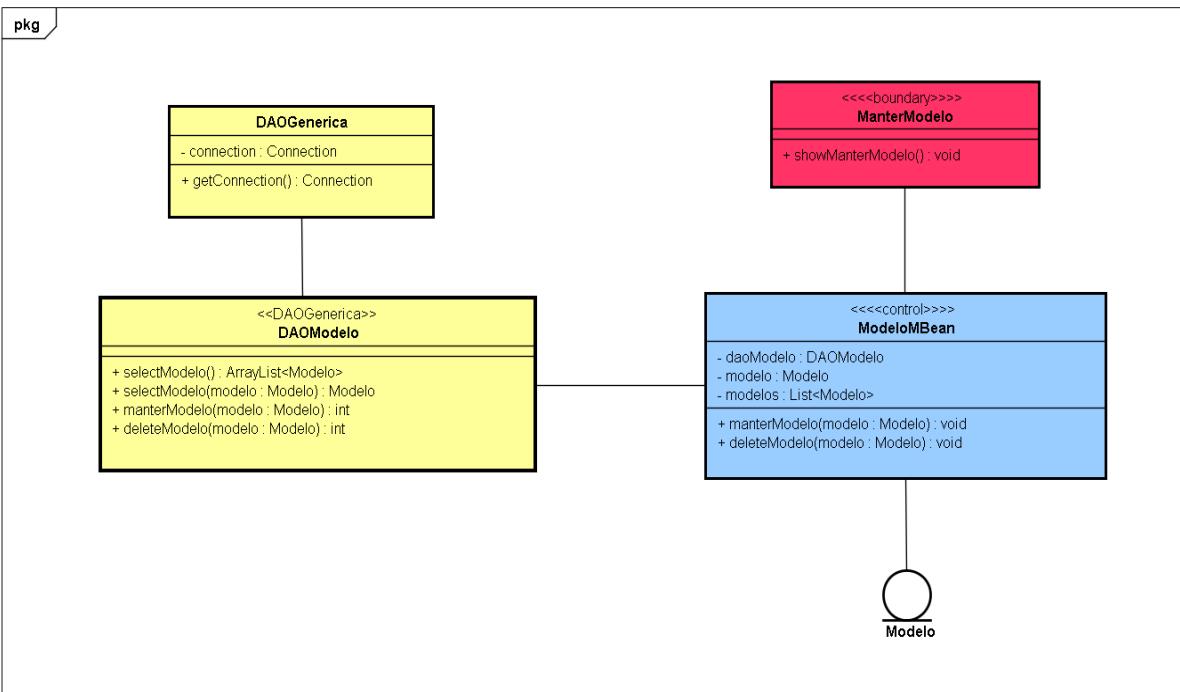


Figura 3.21 – Diagrama de MVC Manter Modelo

Fonte: Elaborado pelos autores, 2016.

3.6 DIAGRAMA DE SEQUÊNCIA

O diagrama de sequência visa demonstrar através de gráficos o desempenho dos elementos de uma aplicação em tempo real, ele também tem a função de simular a troca de mensagens entre as partes do sistema em determinada ação, (BAESSO, 2004).

Este tópico apresentará os diagramas de sequência que fazem parte do sistema.

3.6.1 Manter Pessoa

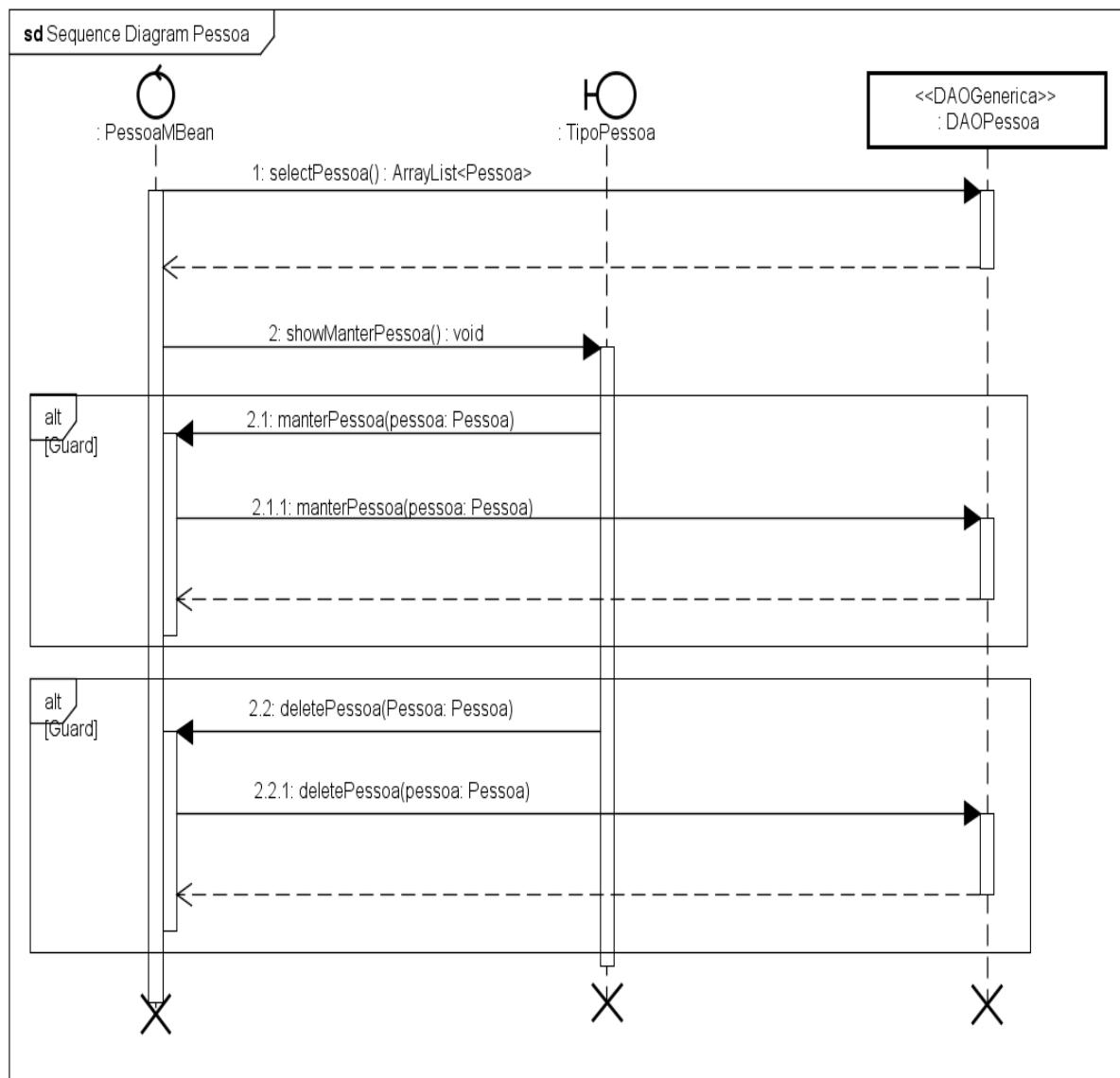


Figura 3.22 – Diagrama de Sequência Manter Pessoa
Fonte: Elaborado pelos autores, 2016.

3.6.2 Manter Despesas

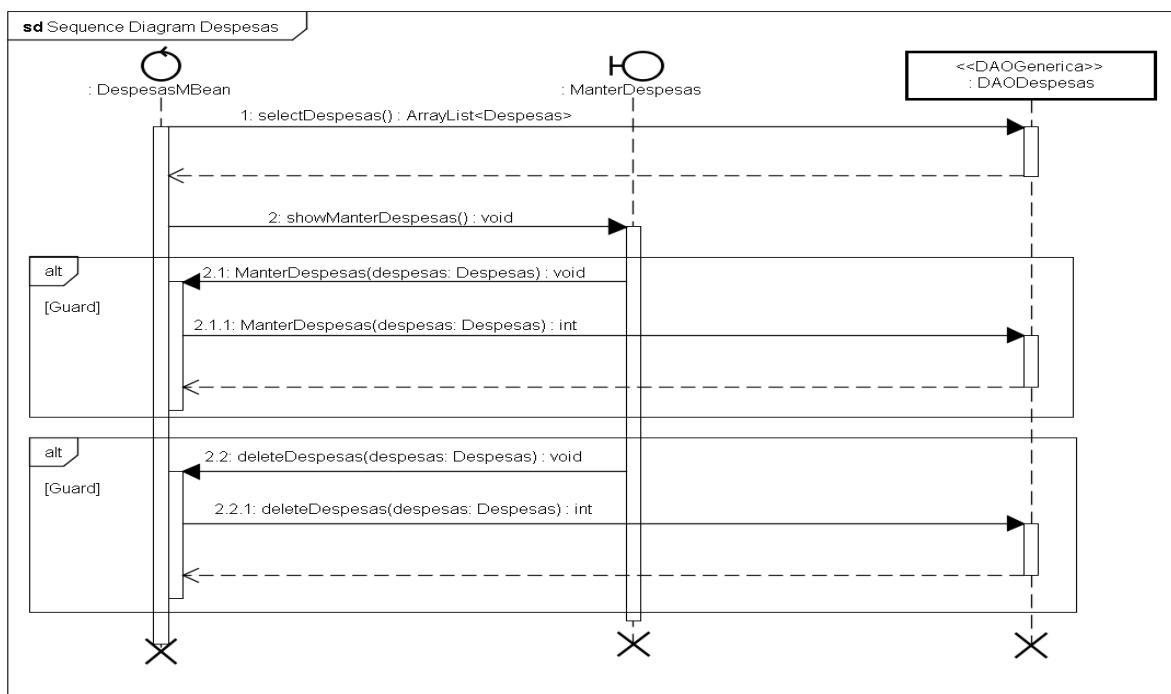


Figura 3.23 – Diagrama de Sequência Manter Despesas

Fonte: Elaborado pelos autores, 2016.

3.6.3 Manter DespesasVeiculo

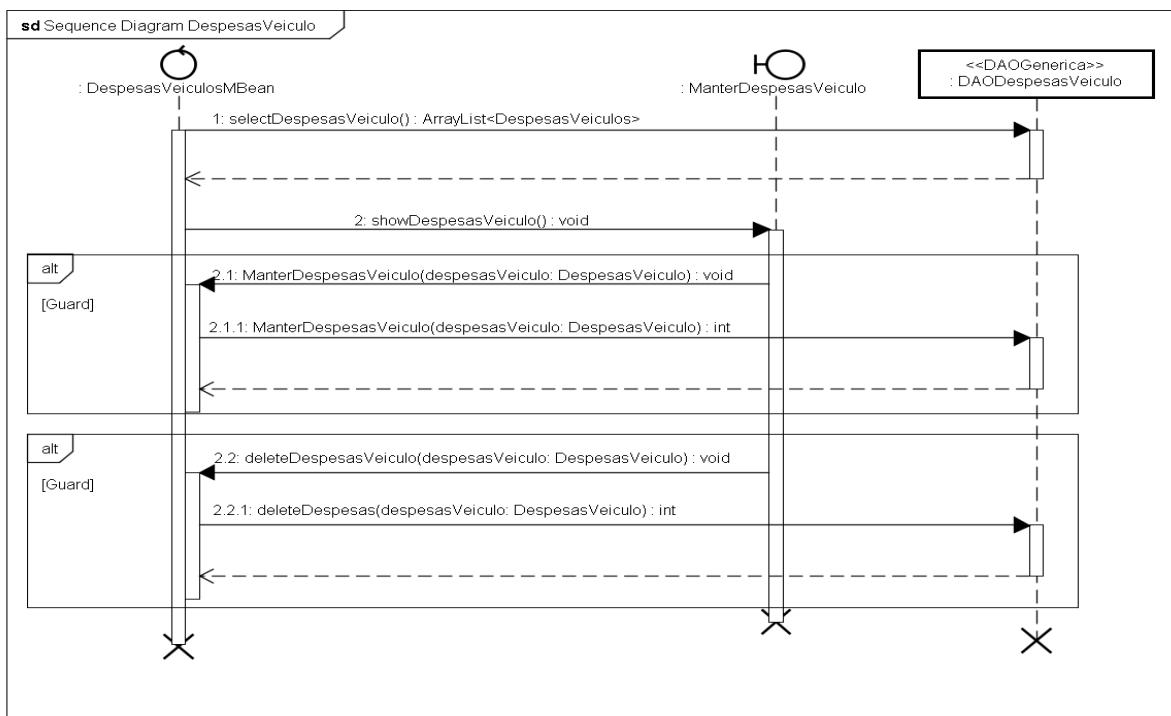


Figura 3.24 – Diagrama de Sequência Manter DespesasVeiculo

Fonte: Elaborado pelos autores, 2016.

3.6.4 Manter Funcionário

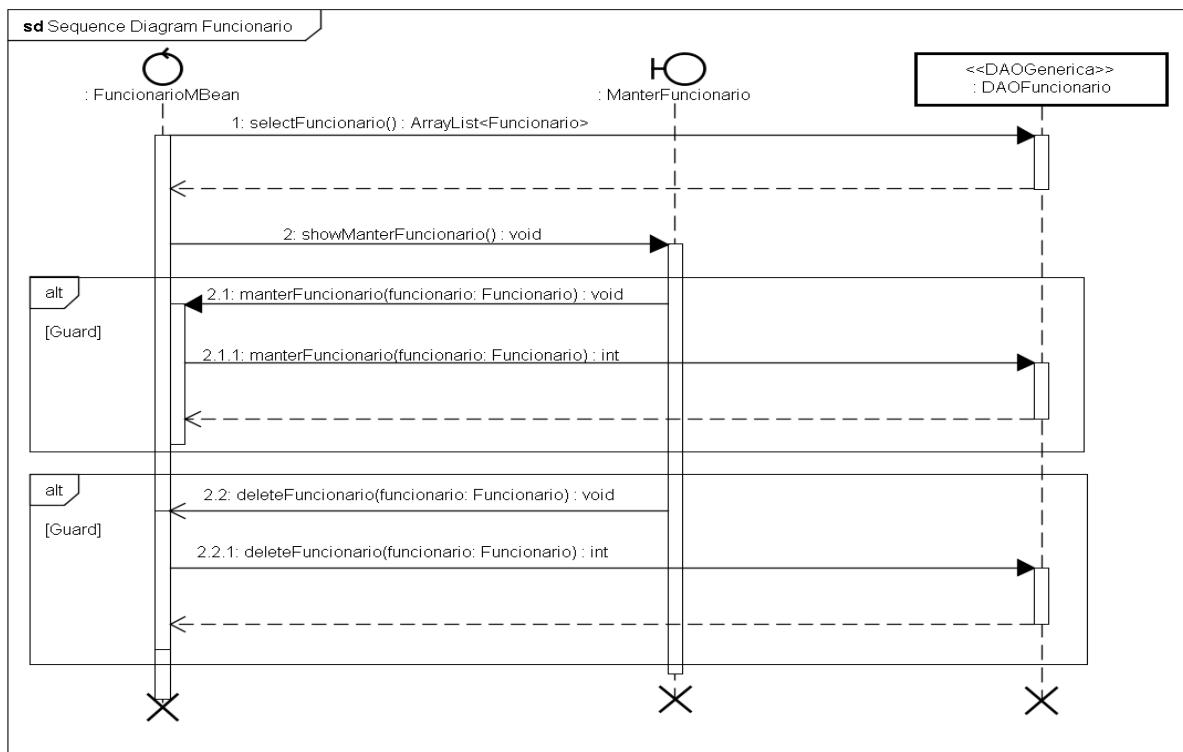


Figura 3.25 – Diagrama de Sequência Manter Funcionário
Fonte: Elaborado pelos autores, 2016.

3.6.5 Login

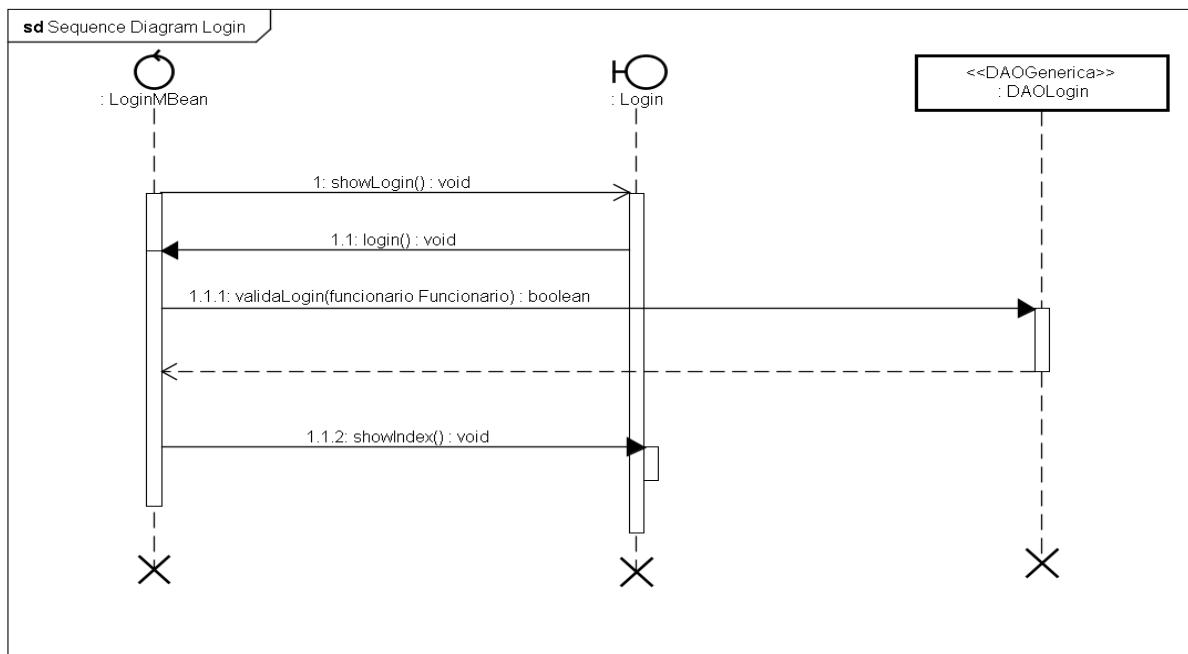


Figura 3.26 – Diagrama de Sequência Login
Fonte: Elaborado pelos autores, 2016.

3.6.6 Manter Movimento

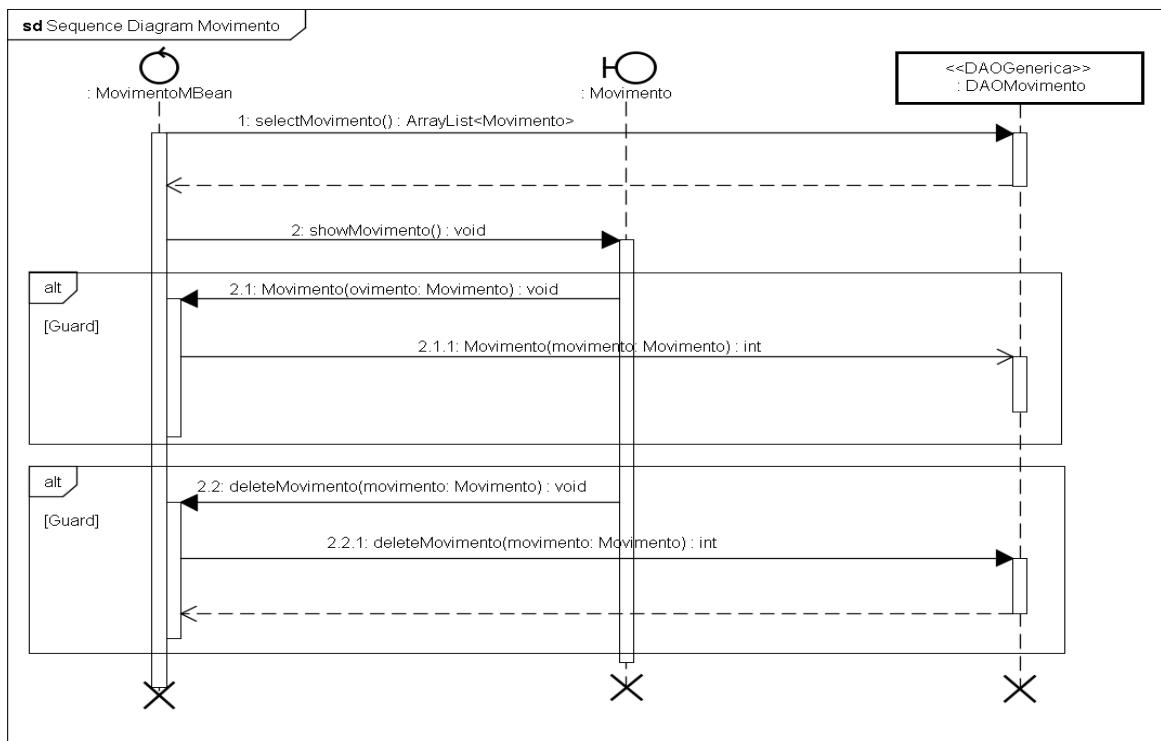


Figura 3.27 – Diagrama de Sequência Manter Movimento

Fonte: Elaborado pelos autores, 2016.

3.6.7 Manter Marca

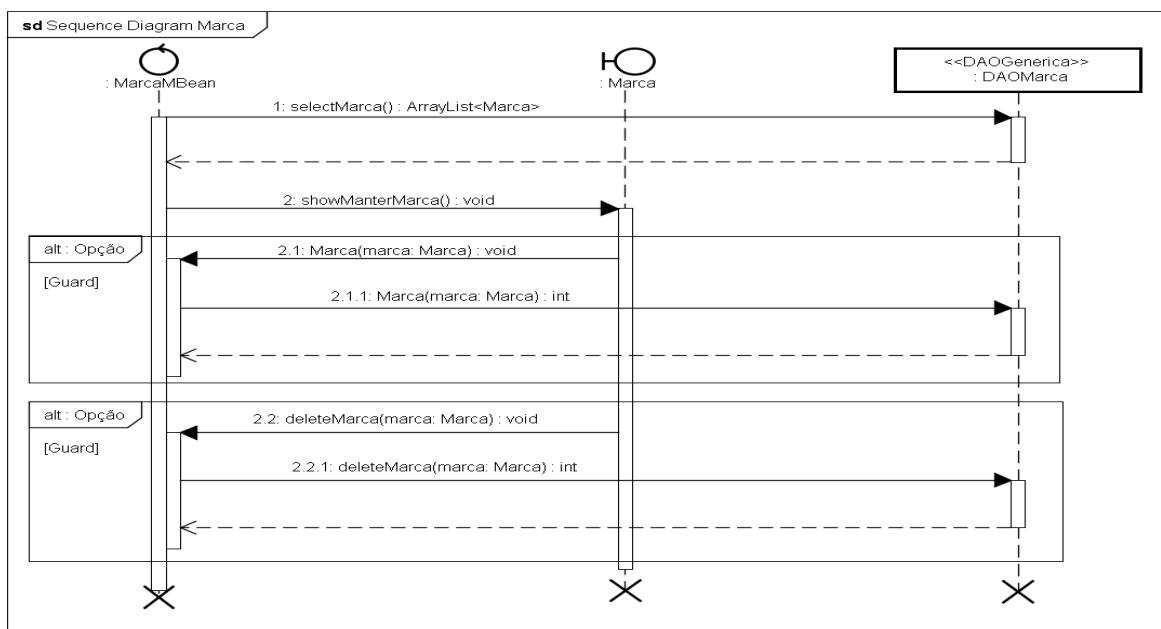


Figura 3.28 – Diagrama de Sequência Manter Marca

Fonte: Elaborado pelos autores, 2016.

3.6.8 Manter Veiculo

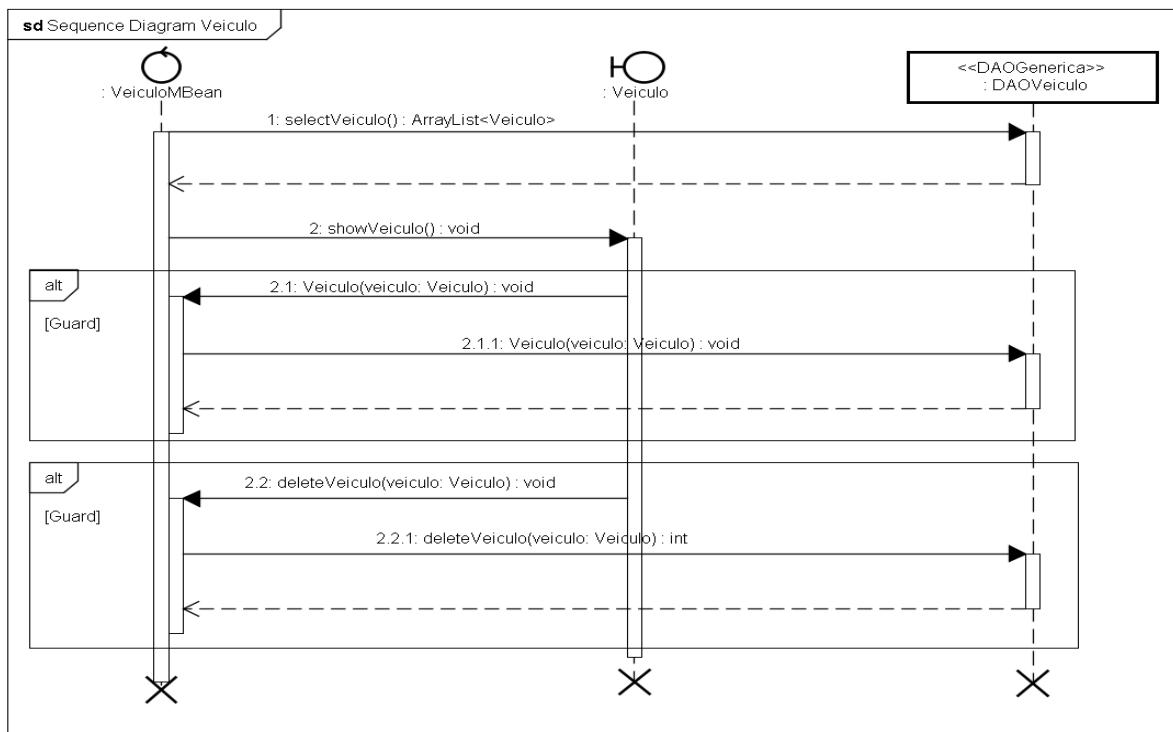


Figura 3.29 – Diagrama de Sequência Manter Veiculo
Fonte: Elaborado pelos autores, 2016.

3.6.9 Manter Modelo

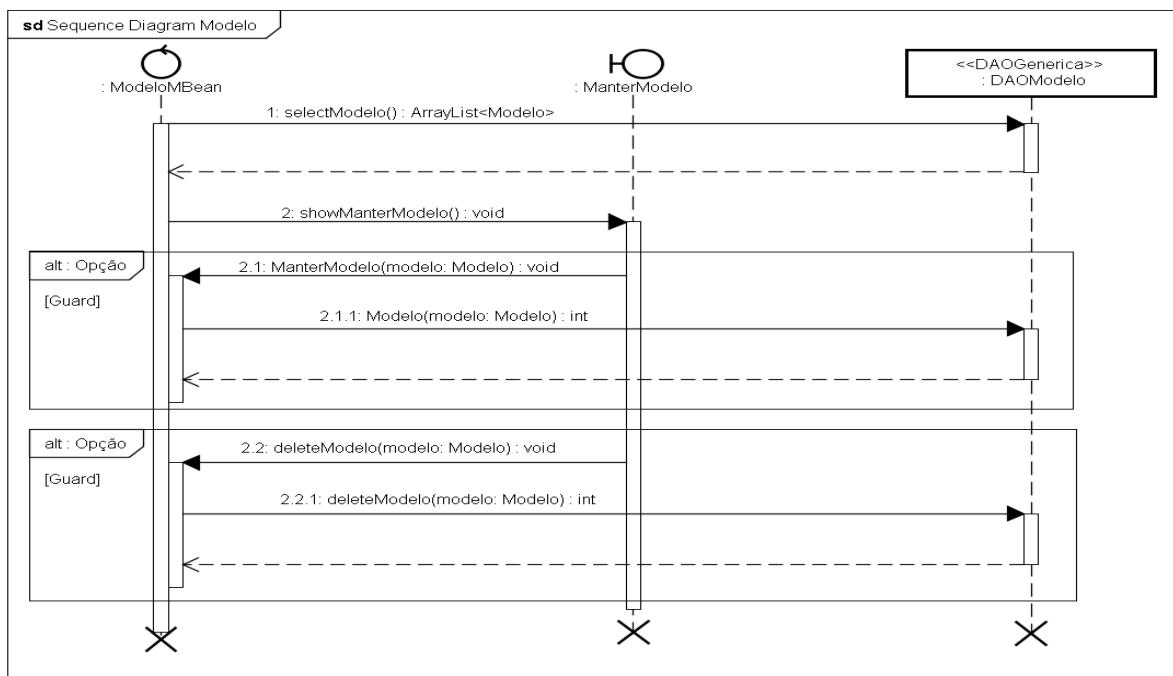


Figura 3.30 – Diagrama de Sequência Manter Modelo
Fonte: Elaborado pelos autores, 2016.

No capítulo seguinte será apresentado a implementação do sistema, mostrando imagens dos procedimentos que o sistema realiza e também telas do phpMyAdmin, software que foi utilizado para gerenciar o banco de dados.

4 DESENVOLVIMENTO DO SISTEMA

O quarto capítulo tem o objetivo de exibir a forma com que as ferramentas tecnológicas foram empregadas no desenvolvimento do software. O presente capítulo também tem a função de mostrar o desempenho do sistema Web e gerenciamento das rotinas desempenhadas na revenda de veículos.

Assim sendo, este capítulo trabalhará as imagens das telas mais utilizadas no sistema.

4.1 DESENVOLVIMENTO DO BANCO DE DADOS

A implementação do banco de dados foi realizada com base nos diagrama apresentados no capítulo anterior. O SGBD utilizado foi o MySQL, em concordância com o texto descrito no primeiro capítulo.

As criações de todas as tabelas, consultas SQL, e outras atividades relacionadas ao banco de dados foram feitas utilizando a ferramenta phpMyAdmin, conforme figura 4.1.

Tabela	Ações	Registros	Tipo	Agrupamento (Collation)	Tamanho	Suspensa
caracteristica	Procurar Estrutura Pesquisar Insere Limpa Elimina	~2	InnoDB	utf8_general_ci	32 KB	-
cidade	Procurar Estrutura Pesquisar Insere Limpa Elimina	~822	InnoDB	utf8_unicode_ci	728 KB	-
despesa	Procurar Estrutura Pesquisar Insere Limpa Elimina	~4	InnoDB	latin1_swedish_ci	16 KB	-
endereco	Procurar Estrutura Pesquisar Insere Limpa Elimina	~6	InnoDB	utf8_general_ci	32 KB	-
entrada	Procurar Estrutura Pesquisar Insere Limpa Elimina	~8	InnoDB	latin1_swedish_ci	16 KB	-
estado	Procurar Estrutura Pesquisar Insere Limpa Elimina	~27	InnoDB	utf8_unicode_ci	64 KB	-
funcionario	Procurar Estrutura Pesquisar Insere Limpa Elimina	~3	InnoDB	utf8_general_ci	32 KB	-
marca	Procurar Estrutura Pesquisar Insere Limpa Elimina	~88	InnoDB	utf8_general_ci	16 KB	-
modelo	Procurar Estrutura Pesquisar Insere Limpa Elimina	~390	InnoDB	utf8_general_ci	336 KB	-
pessoa	Procurar Estrutura Pesquisar Insere Limpa Elimina	~3	InnoDB	utf8_general_ci	16 KB	-
pessofisica	Procurar Estrutura Pesquisar Insere Limpa Elimina	~3	InnoDB	utf8_general_ci	32 KB	-
pessoajuridica	Procurar Estrutura Pesquisar Insere Limpa Elimina	~8	InnoDB	utf8_general_ci	32 KB	-
saida	Procurar Estrutura Pesquisar Insere Limpa Elimina	~8	InnoDB	latin1_swedish_ci	16 KB	-
veiculo	Procurar Estrutura Pesquisar Insere Limpa Elimina	~2	InnoDB	utf8_general_ci	16 KB	-
veiculodespesa	Procurar Estrutura Pesquisar Insere Limpa Elimina	~8	InnoDB	latin1_swedish_ci	16 KB	-
15 tabelas	Soma	13,942	InnoDB	latin1_swedish_ci	1.4 MB	0 Bytes

Figura 4.1 – Estrutura de tabelas do banco de dados
Fonte: Elaborado pelos autores, 2016.

Representação do modelo de uma tela, com a criação de uma tabela e seus atributos, conforme a figura 4.2.

The screenshot shows the MySQL Workbench interface with the following details:

- Table name:** pessoa
- Add:** 1 column(s)
- Executar** button
- Nome** column header
- Tipo** column header
- Tamanho/Valores*** column header
- Predefinido** column header
- Agrupamento (Collation)** column header
- Atributos** column header
- Nulo** column header
- Índice** column header
- A_J** column header
- Comentários** column header
- id**: INT, None, None, None, PRIMARY, checked
- nome**: VARCHAR(255), None, None, None, None, None
- : INT, None, None, None, None, None
- : INT, None, None, None, None, None
- Comentários da tabela:** (empty)
- Storage Engine:** InnoDB
- Collation:** (empty)
- PARTITION definition:** (empty)
- Guarda** button
- quarta-feira, 15 de junho de 2016** (bottom right corner)

Figura 4.2 – Estrutura de tabelas do banco de dados

Fonte: Elaborado pelos autores, 2016.

4.1 LOGIN

Para oferecer maior segurança, o usuário só poderá ter acesso às funcionalidades do sistema quando concretizar a efetivação do Login. Explanação da tela de acesso ao programa, representada na figura 4.3.



Figura 4.3 – Login

Fonte: Elaborado pelos autores, 2016.

4.2 DESIGN RESPONSIVO

Com o auxílio da ferramenta Bootstrap, o sistema foi desenvolvido de forma responsiva, ou seja, os elementos que fazem parte da tela de visualização se adaptam automaticamente à largura de tela do dispositivo no qual ele está sendo visualizado. Conforme representação das figuras 4.4, 4.5, 4.6 e 4.7.



Figura 4.4 – Tela de Design Responsivo com resolução 1366 x 768
Fonte: Elaborado pelos autores, 2016.



Figura 4.5 – Tela de Design Responsivo com resolução 800 x 600
Fonte: Elaborado pelos autores, 2016.

Pode-se notar que a partir do momento que o sistema começa a operar com uma tela menor, o menu é recolhido, e para acessá-lo é necessário clicar no botão Menu que é mostrado nas figuras 4.6 e 4.7.



Figura 4.6 –Resolução 400x800 com menu recolhido
Fonte: Elaborado pelos autores, 2016.

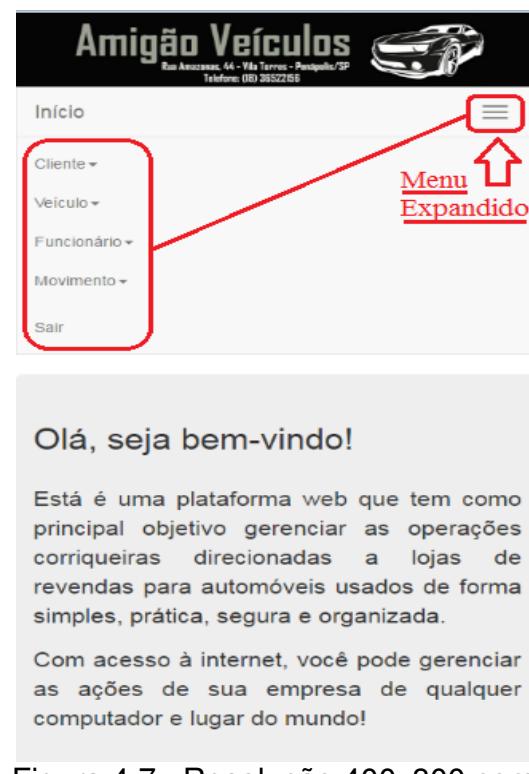


Figura 4.7 –Resolução 400x800 com menu expandido
Fonte: Elaborado pelos autores, 2016.

4.3 TELA PRINCIPAL

Após realizar a autenticação, o usuário terá acesso a tela principal do sistema, onde o mesmo poderá realizar os cadastros e controle de estoque. No menu principal, figura 4.8, será possível realizar as seguintes ações: cadastro de cliente, funcionário, movimento (onde estarão localizadas as opções de compra e venda), veículo, onde além de cadastrar os veículo, também será possível realizar a inserção da marca e modelo e possíveis despesas do veículo.



Figura 4.8 – Tela principal do Sistema
Fonte: Elaborado pelos autores, 2016.

No menu cadastro de clientes, o usuário contará com os seguintes sub-menus: cadastrar, consultar e excluir, representada na figura 4.9.

Se o usuário optar pela opção cadastrar será aberta a tela para inserção de dados do cliente, após a digitação dos dados, deverá ser escolhida a opção cadastrar, conforme figura 4.9.

Amigão Veículos
Rua Amazonas, 44 - Vila Torres - Penápolis/SP
Telefone: (18) 36522156

Início Cliente Véículo Funcionário Movimento Sair

Cadastro de Cliente

Tipo de Cliente

Pessoa Física

Dados Pessoais

*CPF:

*Data de Nascimento:
 Dia | Mês | Ano

*Sexo:

*Estado Civil:

*Nome:

*Sobrenome:

*CNH:

*RG:

*Expedição do RG:

*Nome do Pai:

*Nome da Mãe:

Dados Residenciais

*Endereço:

*Número:

*Bairro:

*CEP:

*Complemento:

*UF:

*Cidade:

Contato:

*Telefone:

*Celular:

Email:

Cadastrar

Figura 4.9 – Tela Menu Cadastro de Clientes - opção Cadastrar
Fonte: Elaborado pelos autores, 2016.

Na opção consultar, figura 4.10, será possível visualizar os dados inseridos na opção cadastrar. Caso dentro da consulta o usuário queira efetuar alguma alteração, basta fazer a alteração e escolher a opção atualizar.

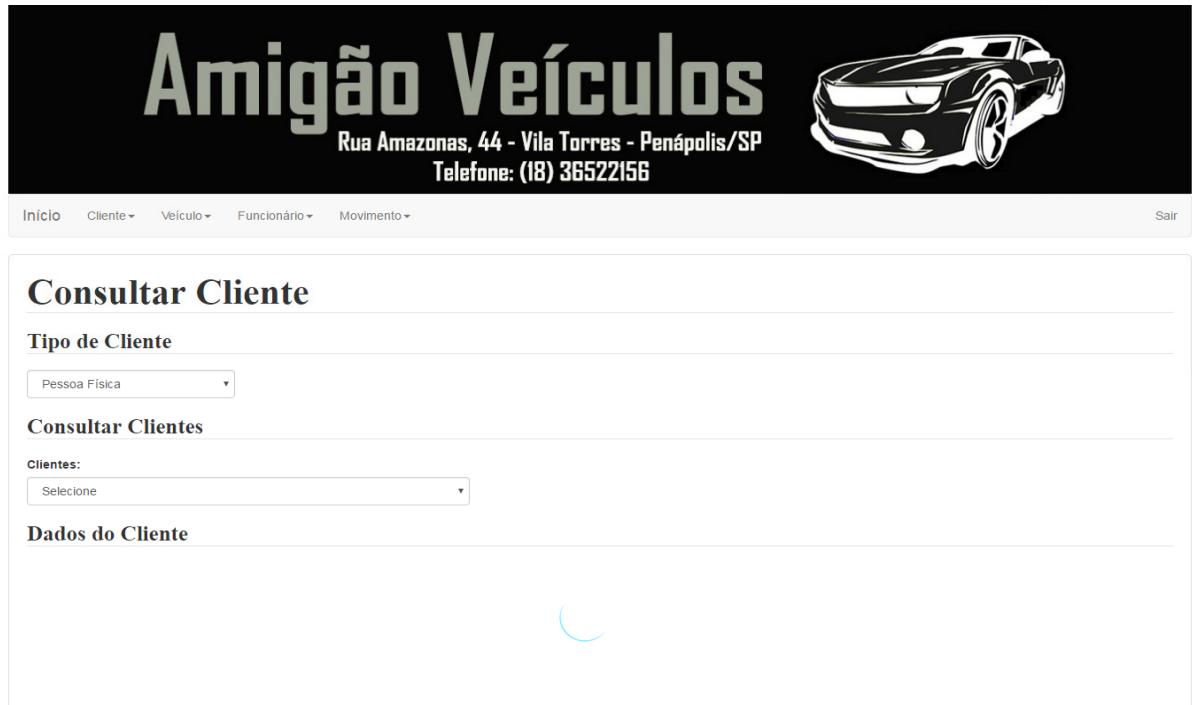


Figura 4.10 – Tela Menu Cadastro de Clientes, opção Consultar
Fonte: Elaborado pelos autores, 2016.

Também poderá ser escolhida a função excluir, no qual o usuário poderá acessar essa opção e ao selecionar o cliente desejado e clicar no botão excluir, o registro será apagado do banco de dados de forma permanente, representado na figura 4.11.

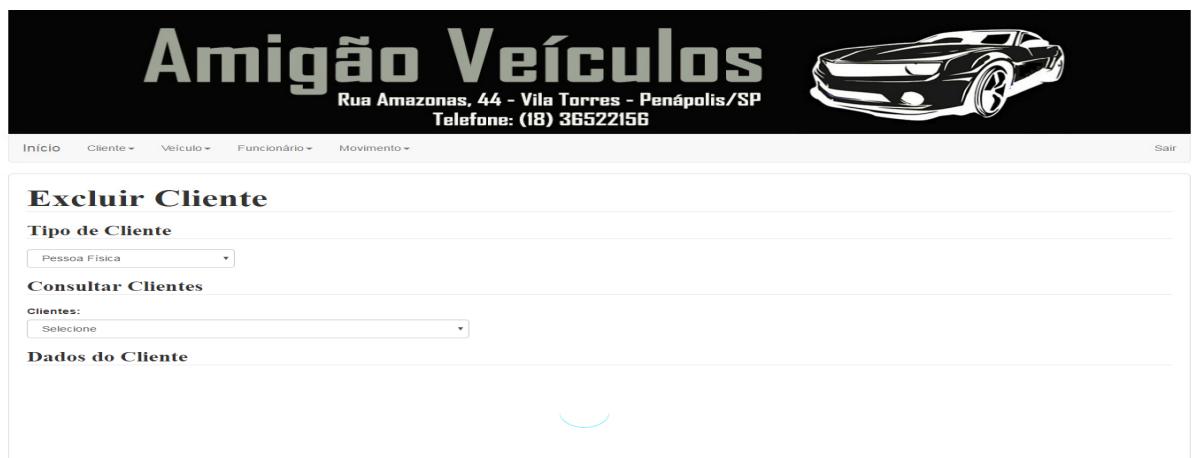


Figura 4.11 – Tela Menu Cadastro de Clientes - opção Excluir
Fonte: Elaborado pelos autores, 2016.

Ao escolher a opção incluir, o usuário optará pela opção pessoa física ou pessoa jurídica. Os campos CPF e CNPJ foram submetidos a tratamento de validação de dados, ou seja, o sistema não aceita o cadastro caso seja digitado uma sequência inválida de números, conforme podemos ver nas figuras 4.12 e 4.13.

Figura 4.12 – Tela Validação CPF
Fonte: Elaborado pelos autores, 2016.

Figura 4.13 – Tela Validação CNPJ
Fonte: Elaborado pelos autores, 2016.

O cadastro de funcionários segue o mesmo padrão do cadastro de clientes, conforme a Figura 4.14.



Figura 4.14 – Tela de Cadastro de Funcionários

Fonte: Elaborado pelos autores, 2016.

Também existe a opção veículo, que é composta pelos sub-menus, cadastrar, consultar e excluir veículos, marca, modelo e despesas, representada na Figura 4.15.



Figura 4.15 – Tela de Menu Veículo

Fonte: Elaborado pelos autores, 2016.

A figura 4.16 mostra o funcionamento da opção cadastrar marcas.

ID	Marca	Ações
1	ACURA	<button>Editar</button> <button>Excluir</button>
2	AGRALE	<button>Editar</button> <button>Excluir</button>
3	ALFA ROMEO	<button>Editar</button> <button>Excluir</button>
4	AM GEN	<button>Editar</button> <button>Excluir</button>
5	ASIA MOTORS	<button>Editar</button> <button>Excluir</button>
6	AUDI	<button>Editar</button> <button>Excluir</button>
7	BMW	<button>Editar</button> <button>Excluir</button>
8	BRM	<button>Editar</button> <button>Excluir</button>
9	BUGGY	<button>Editar</button> <button>Excluir</button>

Figura 4.16 – Tela de cadastro de marcas
Fonte: Elaborado pelos autores, 2016

A figura 4.17 exibe a tela da função de cadastro de modelos de veículos.

ID	Marca	Ações
5986	ACURA	<button>Editar</button> <button>Excluir</button>
2200	INTEGRA GS 1.8	<button>Editar</button> <button>Excluir</button>
2199	LEGEND 3.2/3.5	<button>Editar</button> <button>Excluir</button>
2198	NSX 3.0	<button>Editar</button> <button>Excluir</button>

Figura 4.17 – Tela de cadastro de modelos
Fonte: Elaborado pelos autores, 2016

Também dentro do menu veículo, existe a opção que possibilita o cadastro de despesas, Figura 4.18. Se o usuário escolher essa opção, ele poderá cadastrar as despesas que depois serão inseridas na opção movimento, para saber o valor final do veículo.

Cadastro de Despesa

Dados de Despesa

*Despesa:

*Valor:

Cadastrar

***Despesas Cadastradas:**

ID	Despesas	Valor	Ações
1	POLIMENTO	R\$ 100,00	Editar Excluir
2	LAVAGEM SIMPLES	R\$ 50,00	Editar Excluir
3	LAVAGEM COMPLETA	R\$ 90,00	Editar Excluir
4	REVITALIZAÇÃO EM FAROL	R\$ 15,00	Editar Excluir
5	HIGIENIZAÇÃO	R\$ 80,00	Editar Excluir

Figura 4.18 – Tela de cadastro de despesas
Fonte: Elaborado pelos autores, 2016

O controle de estoque será feito na aba Movimento, Figura 4.19. Onde estarão as opções Entrada, Consultar Entrada, Saída e Consultar Saída. Esta parte do menu será responsável por controlar as compras e vendas de veículos.

Olá, seja bem-vindo!

Está é uma plataforma web que tem como objetivo gerenciar as operações corriqueiras direcionadas a lojas de revendas para automóveis usados de forma simples, prática, segura e organizada.

Com acesso à internet, você pode gerenciar as ações de sua empresa de qualquer computador e lugar do mundo!

Figura 4.19 – Tela de Movimento
Fonte: Elaborado pelos autores, 2016.

Se o usuário clicar na opção Entrada, será exigido a inserção de dados como o cliente que vendeu o carro, funcionário que realizou a compra, veículo que está sendo comprado, despesas que serão necessárias para a posterior venda, entre outras. Conforme Figura 4.20.

Amigão Veículos
Rua Amazonas, 44 - Vila Torres - Penápolis/SP
Telefone: (18) 36522156

Entrada de Veículo

Dados do Veículo

*Funcionário:

*Data:

*Valor:

Adicionar Veículo

Selecionar:

Dados do Veículo

*Marca:

*Modelo:

*Placa:
 /

Fabricação / Modelo:

Cor:

Despesas do Veículo

Despesas:	Data	Despesa	Valor
		TOTAL	

Proprietário

Tipo de Cliente

Pessoa Física

Consultar Clientes

Clientes:

Dados do Cliente

*CPF:

*RG:

*Endereço:

*Número:

*Bairro:

*CEP:

*UF:

*Cidade:

Cadastrar

Figura 4.20 – Tela de Movimento - Entrada
Fonte: Elaborado pelos autores, 2016.

Na opção Consultar Entrada, figura 4.21, será possível visualizar os dados inseridos na opção Entrada. Caso dentro da consulta o usuário queira efetuar alguma alteração, basta editar o campo desejado e escolher a opção atualizar.

Amigão Veículos
Rua Amazonas, 44 - Vila Torres - Penápolis/SP
Telefone: (18) 36522156

Início Cliente Véiculo Funcionário Movimento Sair

Consultar Entradas

*Data Inicial:

*Data Final:

Consultar Veículos

Veículos:

Dados do Véículo

*Marca:

*Placa:

Fabricação / Modelo:

Cor:

Dados da Entrada

Data	Funcionário Responsável	Valor

Despesas do Véículo

Despesas:	Data	Despesa	Valor
	29/06/2016		
		TOTAL	

Proprietário

*Razão Social:

*Fantasia/Nome:

*CNPJ/CPF:

*IE/RG:

*Endereço:

*Número:

*Bairro:

*CEP:

*UF:

*Cidade:

Atualizar

Figura 4.21 – Tela de Movimento – Consultar Entrada
Fonte: Elaborado pelos autores, 2016.

Já a opção saída ou venda, nessa opção será necessário informar o funcionário que realizou a venda, o veículo que está sendo comercializado, as despesas que foram gastos para cálculo do valor venal e o cliente comprador. Representada na Figura 4.22.

Amigão Veículos
Rua Amazonas, 44 - Vila Torres - Penápolis/SP
Telefone: (18) 36522156

Saída de Veículo

Dados do Veículo

*Funcionário:

*Data:

*Valor:

Veículo

*Placa:

*Marca:

*Modelo:

*Ano de Fabricação:

*Ano Modelo:

*Placa:

Despesas:

Descrição:	Valor:
POLIMENTO	R\$ 100,00
LAVAGEM SIMPLES	R\$ 50,00
LAVAGEM COMPLETA	R\$ 90,00
REVITALIZAÇÃO EM FAROL	R\$ 15,00
HIGIENIZAÇÃO	R\$ 80,00
TOTAL	R\$ 335,00

Proprietário

*Nome:

*CPF:

*RG:

*Endereço:

*Número:

*Bairro:

*CEP:

*UF:

*Cidade:

*Telefone:

*Celular:

Comprador

*Nome:

*CPF:

*RG:

*Endereço:

*Número:

*Bairro:

*CEP:

*UF:

*Cidade:

*Telefone:

*Celular:

Cadastrar

Figura 4.22 – Tela de Movimento - Saída
Fonte: Elaborado pelos autores, 2016.

Na opção consultar, figura 4.23, será possível visualizar os dados inseridos na opção Cadastrar Saída. Se desejar fazer alguma alteração, basta editar o campo desejado e escolher a opção atualizar.

Amigão Veículos

Rua Amazonas, 44 - Vila Torres - Penápolis/SP
Telefone: (18) 36522156



Início
Cliente
Veículo
Funcionário
Movimento
Sair

Consultar Saída

*Data Inicial:

*Data Final:

Consultar Veículos

Veículos:

Dados do Veículo

*Marca:

*Placa: /

Fabricação / Modelo:

Cor:

Despesas do Veículo

Despesas:	Data	Despesa	Valor
TOTAL			

Dados da Entrada

Data	Funcionário Responsável	Valor

Valor de Saída

Data	Descrição	Valor

Proprietário

*Razão Social:

*Fantasia/Nome:

*CNPJ/CPF:

*IE/RG:

*Endereço:

*Número:

*Bairro:

*CEP:

*UF:

*Cidade:

Comprador

*Razão Social:

*Fantasia/Nome:

*CNPJ/CPF:

*IE/RG:

*Endereço:

*Número:

*Bairro:

*CEP:

*UF:

*Cidade:

Figura 4.23 – Tela de Movimento – Consultar Saída
Fonte: Elaborado pelos autores, 2016.

CONCLUSÃO

No presente projeto foi realizada a elaboração de um sistema *Web* para gerenciamento de compra, manutenção e venda de veículos usados, com o propósito de informatizar as procedimentos cotidianos.

Para o bom desenvolvimento do projeto, foi necessária a busca pelo conhecimento de tecnologias que não foram abordadas em salas de aula. O uso da linguagem de programação PHP, do framework Bootstrap que trouxe responsividade ao sistema, e do sistema gerenciador de bancos de dados MySQL para o armazenamento de informações e criação de tabelas, foram de suma importância na elaboração do sistema. Contando com vários mecanismos de busca de informações como internet, revistas, livros e artigos, foi possível extrair muitas informações sobre a utilização dessas poderosas ferramentas.

No projeto desenvolvido, empregou-se grande parte do conhecimento assimilado durante os semestres, tendo em vista que o sistema utiliza de forma prática os concepções e fundamentos disponibilizados pelas disciplinas, como a Modelagem de dados, conceito de Orientação a Objetos, Interação do ser humano com computador, Administração e Empreendedorismo, estudo de Bancos de Dados, Linguagens de Programação, entre outros conteúdos afins.

Com a aplicação do sistema de controle de veículos, é esperado, um grande progresso nos procedimentos diários, cooperando com a velocidade e segurança no cumprimento de tarefas como cadastros, consultas e alterações e afins.

Apesar do sistema já poder ser utilizado, o mesmo pode ser melhorado através da otimização dos recursos existentes e criação de novos módulos, tais como a emissão de relatórios, inserção de fotos dos veículos, busca personalizadas, contas a pagar e contas a receber; login com diferentes com níveis de acessos de acordo com o usuário, entre outros, e também lapidar a programação a fim de tornar este protótipo sólido e confiável, para que futuramente possa ser distribuído para fins comerciais.

REFERÊNCIAS BIBLIOGRÁFICAS

- ASTAH. **ASTAH COMMUNITY** Disponível em: <<http://astah.net/>> Acesso em 05 set. 2013.
- BAESSO, M. **Diagrama de Caso de Uso e Diagrama de Sequência.** 2004. Disponível em: <<http://www.dmo.fee.unicamp.br/~henrique/cursoc++/diagrama.pdf>> Acesso em 23 mai. 2016a.
- BURBECK, S. **APPLICATION PROGRAMMING IN SMALLTALK-80(TM) (AND PYTHON):** How to use Model-View-Controller (MVC), 1992.
- BOOTSTRAP. **Bootstrap Framework de CSS** Disponível em: <<http://getbootstrap.com/>> Acesso em 03 jan. 2014.
- DATE, C. J. **INTRODUÇÃO A SISTEMAS DE BANCOS DE DADOS.** 7. ed. Rio de Janeiro: Editora Campus, 2003.
- DEITEL, P.; DEITEL H.; **Java: como programar.** Tradução Edson Furmankiewicz. 8. ed. São Paulo: Pearson Prentice Hall, 2010.
- DEVMEDIA. **As Principais Características do MySQL.** 2013. Disponível em: <<http://www.devmedia.com.br/as-principais-caracteristicas-do-mysql/2680>> Acesso em 27 out. 2013a.
- DEVMEDIA. **Introduzindo o servidor de aplicação Apache Tomcat.** 2011. Disponível em: <<http://www.devmedia.com.br/introduzindo-o-servidor-de-aplicacao-apache-tomcat/27939#ixzz2kICaHHmV>> Acesso em: 27 out. 2013b.
- DEVMEDIA. **Unindo JavaServer Faces a Ajax: melhorando o processo de desenvolvimento Web.** 2006. Disponível em: <<http://www.devmedia.com.br/visualizaComponente.aspx?comp=3199&site=6>>. Acesso em: 13 nov. 2013c.
- DIAS, M. A. P. **Administração de materiais.** 4. ed. São Paulo: Atlas 1993.
- DIAS, D. **Motivação e resistência ao uso da tecnologia da informação: um estudo entre gerentes.** 2000. Disponível em <<http://www.scielo.br/pdf/rac/v4n2/v4n2a04.pdf>> Acesso em 01 jun. 2016a.
- ELMASRI, R.; NAVATHE, S. B. **Sistemas de Banco de Dados.** 4. ed. São Paulo: Person Education, 2005.
- Ferreira, E.; Eis, D. **HTML 5:** Curso w3c Escritório Brasil. Disponível em <<http://www.w3c.br/pub/Cursos/CursoHTML5/html5-web.pdf>> Acesso em: 15 out. 2013.

FORTUNATO, E. et al. **Gerenciamento de estoque**. 2004. Monografia (Graduação em Ciências Contábeis) – Faculdade de Ciências Administrativas e Contábeis de Lins, Lins.

FREEMAN, E.; FREEMAN, E. **Use a Cabeça! HTML com CSS e XHTML**. 2. ed. Rio de Janeiro: Alta Books, 2008.

GAMA, A. **MVC simples e prático, Parte I**. 2011. Disponível em: <<https://alexandregama.org/2011/03/24/mvc-simples-e-pratico-parte-i/>> Acesso em: 13 abr. 2013.

GONÇALVES, E. **Desenvolvendo Aplicações Web com JSP, Servlets, Java, Server Faces, Hibernate, EJB Persistence e Ajax**. Rio de Janeiro: Ciência Moderna, 2007.

GONÇALVES, E. **Dominando NetBeans**. Rio de Janeiro: Ciência Moderna, 2006.

HORSTMANN, C.S.; CORNELL, G.; Core java volume 1: fundamentos. Tradução Carlos Schafranski e Edson Furmarkiewicz. 8. ed. São Paulo: Pearson Prentice Hall, 2010.

HTML.NET. **O que são tags HTML**. Disponível em: <<http://pt-br.html.net/tutorials/html/lesson3.php>> Acesso em: 20 fev. 2013.

JQUERY: **O que é jQuery?** 2014. Disponível em: <<http://www.jquery.com>> Acesso em: 17 de out. 2015.

LEMAY, L. **Aprenda a Criar Páginas Web com HTML e XHTML**. 3. Ed. São Paulo: Pearson Education do Brasil, 2002.

MEDEIROS, Ernani Sales de. **Desenvolvendo software com UML 2.0**. 1. ed. São Paulo: Person Makron Books, 2004.

MELO, A. A.; NASCIMENTO, M. G. G. **PHP PROFISSIONAL**: Aprenda a Desenvolver Sistemas Profissionais Orientados a Objetos com Padrões de Projeto. 1. ed. Novatec Editora, 2007.

MENDES, Douglas R. **Programação em Java com ênfase em Orientação a Objetos**. São Paulo: Novatec, 2010.

MILANI, A.: **MySQL – GUIA DO PROGRAMADOR**. 1. ed. São Paulo: Novatec Editora Ltda, 2007.

MYSQL. **MySQL 5.5 Reference Manual 2.1.1. Operating Systems Supported by MySQL Community Server**. Disponível em: <<http://dev.mysql.com/doc/refman/5.5/en/supported-os.html>> Acesso em: 15 out. 2013a.

_____. **MySQL 5.5 Reference Manual Chapter 25. MySQL Workbench**. Disponível em: <<http://dev.mysql.com/doc/refman/5.5/en/workbench.html>> Acesso em: 15 out. 2013b.

_____ . MySQL 5.5 Reference Manual 14.3. The Inno DBStorage Engine. Disponível em: <<http://dev.mysql.com/doc/refman/5.5/en/innodb-storage-engine.html>> Acesso em: 15 out. 2013c.

MOURA, C. E. **Gestão de estoques**. Rio de Janeiro: Ciência Moderna, 2004.

NASSU, E. A.; SETZER, V. W. **Banco de Dados Orientados a Objetos**. São Paulo: Edgard BlucherLtda, 2009.

NIEDERAUER, J. **DESENVOLVENDO WEBSITES COM PHP**. 2. ed. São Paulo: Novatec Editora, 2011.

PHPMYADMIN. **The phpMyAdmin Project**. 2011. Disponível em: <<http://www.phpmyadmin.net/>> Acesso em: 05 jun. 2016.

RAMAKRISHNAN, R.; GEHRKE, J. **Sistemas de Gerenciamento de Banco de Dados**. 3. ed. São Paulo: McGraw-Hill, 2008.

SILBERSCHATZ, A.; KORTH, H. F.; SUDARSHAN, S. **Sistema de banco de dados**. Tradução Daniel Vieira. 5. ed. Rio de Janeiro: Elsevier, 2006.

SILVA, Izalmo P. **Artigo Java Magazine 66 – Desenvolvendo Java para web**. 2009. Disponível em: <<http://www.devmedia.com.br/artigo-java-magazine-66-desenvolvendo-java-para-web/12289>> Acesso em: 17 out. 2013.

SILVA, M. S. **CSS3** Desenvolva Aplicações WebProfissionais com uso dos Poderosos Recursos de Estilização das CCS3. 1. ed. São Paulo: Novatec Editora Ltda, 2012.

W3C. **CASCADING STYLE SHEETS**: O que é CSS? Disponível em <<http://www.w3.org/Style/CSS/>> Acesso em: 15/11/2012.

WANKE, P. **Gestão de Estoques na Cadeia de Suprimentos**. 2 ed. São Paulo: Editora Atlas S.A., 2008.

WELLING, L.; THOMSON, L. **PHP e MySQL: Desenvolvimento Web**. 3. ed. Rio de Janeiro: Elsevier Editora Ltda, 2005.

WELLING, L.; THOMSON, L. **PHP e MySQL: Desenvolvimento Web**. 3. ed. Rio de Janeiro: Elsevier Editora Ltda, 2005.