



CENTRO UNIVERSITÁRIO UNIFACVEST
CURSO DE CIÊNCIA DA COMPUTAÇÃO
TRABALHO DE CONCLUSÃO DE CURSO II
EDEMILSON TELLES

**AUXILI CARS: Sistema especialista de apoio à decisão para compra de
automóveis novos online.**

LAGES (SC)

2015

EDEMILSON TELLES

AUXILI CARS: Sistema especialista de apoio à decisão para compra de automóveis novos online.

Projeto apresentado à banca examinadora do Trabalho de Conclusão de Curso II de Ciência da Computação para análise e aprovação.

LAGES (SC)

2015

EDEMILSON TELLES

AUXILI CARS: sistema especialista de apoio à decisão para compra de automóveis novos online.

Trabalho de Conclusão de Curso II de Ciência da Computação apresentado ao Centro Universitário UNIFACVEST como parte de requisitos para obtenção do título de bacharel em Ciência da Computação.

Prof. Msc. Márcio Sembay

Lages. SC ____/____/ 2015. Nota _____

LAGES (SC)

2015

AGRADECIMENTOS

A Deus, por ter me dado forças e fé para seguir meu caminho, alcançando meus objetivos;

Aos familiares e amigos por acreditarem no meu potencial, sempre com palavras de incentivo;

Aos professores por passarem o conhecimento adquirido no decorrer do curso.

RESUMO

Esta página web apresenta um modelo de sistema especialista no auxílio à compra de veículos novos na internet. Como as compras pela internet estão cada vez mais presentes na vida de todos, um sistema especialista é importante no auxílio para aquisição de produtos através da rede mundial de computadores, o *e-commerce* entra no contexto para ditar regras que auxiliam a criação de páginas na compra e venda desses produtos. O projeto do sistema especialista em *PHP* mostrou-se eficiente na automatização dos processos para o auxílio de compras de automóveis através da internet. O objetivo principal do projeto é o desenvolvimento de uma ferramenta *on-line* para atender a necessidade de auxílio à compra de automóveis, tendo como objetivo específico atender ao público que acessa a página, no direcionamento em que tomar para compra do automóvel. A metodologia utilizada para a pesquisa foi pesquisa bibliográfica em livros de Sistemas Especialistas de Apoio a Decisão, Inteligência Artificial, livros de vendas pela internet, sites, artigos e materiais sobre a criação de *softwares*.

Palavras chaves: Sistema Especialista, *e-commerce*, *PHP*.

ABSTRACT

This web page presents an expert system model in aid to the purchase of new vehicles on the internet. As the internet shopping are increasingly present in the lives of everyone, an expert system is important in aid for the purchase of products through the World Wide Web, e-commerce enters the context to dictate rules that helps create pages buying and selling of these products. The project expert system in PHP proved to be efficient in process automation to the shopping aid car over the internet. The main objective of the project is to develop an online tool to meet the need for aid to car purchases, with the specific goal serve the public to access the page in the direction in which to take to buying the car. The methodology used for the survey was bibliographic research in books of Expert Systems Decision Support, Artificial Intelligence, sales of books over the Internet, websites, articles and materials about creating software.

Key words : Expert System , e- commerce, PHP.

RESUMEN

Esta página web muestra un modelo de sistema experto en ayuda a la compra de vehículos nuevos en el Internet. A medida que las compras por Internet son cada vez más presentes en la vida de todos, un sistema experto es importante en la ayuda para la compra de productos a través de la World Wide Web, comercio electrónico entra en el contexto de dictar normas que ayuda a crear páginas de compra y venta de estos productos. El sistema experto proyectado en PHP demostró ser eficiente en la automatización de procesos para el comercio de ayuda comercial en Internet. El objetivo principal del proyecto es desarrollar una herramienta en línea para satisfacer la necesidad de la ayuda a la compra de automóviles, con el objetivo específico de servir al público para acceder a la página en la dirección en la que tomar para comprar el coche. La metodología utilizada para el estudio fue la investigación bibliográfica en libros de Apoyo a la Decisión, Sistemas Expertos, Inteligencia Artificial, las ventas de libros en Internet, sitios web, artículos y materiales acerca de la creación de software.

Palabras clave: Sistema Experto , *e-commerce*, Java, *PHP*

LISTA DE SIGLAS

AJAX	– <i>Asynchronous Java Script And xml</i>
BC	– Base de Conhecimento
CSS	– Cascading Style Sheets (Folhas de Estilo em Cascata).
GPS	– General Problem Solver (Solucionador de Problemas Genéricos)
GPL	- General Public License
HTML	– <i>Hyper Text Markup Language</i> (Linguagem de Marcação de Hipertexto)
HTTP	– <i>Hypertext Transfer Protocol</i> (Protocolo de transferência de Hipertexto)
IA	– Inteligência Artificial
IDE	– <i>Integrated Development Environment</i> (Ambiente Integrado de Desenvolvimento)
MT	– Memória de Trabalho
MVC	– <i>Model-view-controller</i>
NSBC	– Núcleo do Sistema Baseado em Conhecimento
PDA	– <i>Personal Digital Assistants</i>
PHP	– <i>Hypertext Preprocessor</i> (Preprocessador de Hipertexto).
RBC	– Raciocínio baseado em casos
SE	– Sistema Especialista
SGBD	– Sistema de Gerenciamento de Banco de Dados
SQL	– <i>Structured Query Language</i> (Linguagem de Consulta Estruturada)
SSI	– <i>Structured Query Language</i> (Linguagem de Consulta Estruturada)
SSL	– Server Side Includes (Inclusão ao Lado Servidor)
TCP	– Transmission Control Protocol (Protocolo de Controle de Transmissão)
UFPA	– Universidade Federal do Pará
WWW	– <i>World Wide Web</i> (Teia Mundial)
XML	– <i>Extensible Markup Language</i>

LISTA DE FIGURAS

Figura 1: Estrutura dos Sistemas Especialistas	17
Figura 2: Expert Sinta.....	20
Figura 3: Stress Analysis	21
Figura 4: Ferramenta JESS	22
Figura 5: Herança não estrita	26
Figura 6: Herança Múltipla.....	26
Figura 7: Código Java.....	27
Figura 8: Protocolos de Comunicação utilizados em Windows e Linux	29
Figura 9 – Estrutura MVC do projeto <i>NetBeans 7.2.1</i>	34
Figura 10: Tela do projeto <i>NetBeans 7.2.1</i>	34
Figura 11: Tela do projeto <i>MySql Workbench</i>	35
Figura 12: Aplicação <i>EasyPHP</i>	35
Figura 13: Aplicação <i>EasyPHP</i>	36
Figura 14: Servidor <i>MySql</i>	36
Figura 15: Estrutura do banco de dados <i>MySql</i>	37
Figura 16: Diagrama de Estudo de caso	38
Figura 17: Diagrama de Classe.....	39
Figura 18: Diagrama de Fluxo de Dados.....	39
Figura 19: Página Principal do Projeto.....	41
Figura 20: Questões do sistema, opções de estilos.....	41
Figura 21: Questões do sistema, opções de cores.....	42
Figura 22: Questões do sistema, opções de marcas.....	42
Figura 23: Questões do sistema, opções de motores	43
Figura 24: Questões do sistema, opções de portas	43
Figura 25: Botão efetuar pesquisa	44
Figura 26: Resultados	44
Figura 27: Acesso da área administrativa.....	45
Figura 28: Tela inicial de autenticação do administrador do sistema.....	45
Figura 29: Página principal da área administrativa	46
Figura 30: Cadastro de perguntas	47
Figura 31: Perguntas cadastradas	47

Figura 32: Cadastro de respostas	48
Figura 33: Respostas cadastradas	48
Figura 34: Registro de carros	49
Figura 35: Características dos carros.....	49
Figura 36: Código fonte da classe de conexão ao banco de dados.....	53
Figura 37: Código fonte do resultado da pesquisa	53
Figura 38: Código fonte da função pesquisa	54

LISTA DE TABELAS

Tabela 1: Cronograma do TCC 1	15
Tabela 2: Comparação entre Conhecimento Humano e Conhecimento Artificial	16

SUMÁRIO

1 INTRODUÇÃO	13
1.1 JUSTIFICATIVA	13
1.2 OBJETIVOS	14
1.2.1 Objetivo Geral	14
1.2.2 Objetivos Específicos	14
1.3 Cronograma	15
1.4 ESTRUTURA DO TRABALHO	15
2 REVISÃO DA LITERATURA	15
2.1 SISTEMAS ESPECIALISTAS DE APOIO À DECISÃO	15
2.1.1 Sistemas Especialistas X Sistemas Tradicionais	16
2.1.2 Estrutura dos Sistemas Especialistas	17
2.1.3 Núcleo do Sistema Baseado em Conhecimento	17
2.1.4 Base de Conhecimento	18
2.1.5 Memória de Trabalho	18
2.1.6 Interface	19
2.1.7 Implementação de Sistema Especialista	19
2.2 INTELIGÊNCIA ARTIFICIAL	22
2.2.1 IA Forte e IA Fraco	23
2.2.2 Métodos Fracos e Métodos Fortes	24
2.2.3 Raciocínio baseado em casos	24
2.2.4 Seleção do <i>best match</i>	25
2.3 JAVA	25
2.3.1 Histórico Java	25
2.3.2 Polimorfismo	26
2.3.3 Desenvolvimento Orientada à Objeto	27
2.4 PHP	28
2.5 MYSQL	28
2.6 NetBeans IDE 7.2	29
2.7 SERVIDOR APACHE	30
2.8 JQUERY	30
2.8.1 JQueryUI	31
2.9 EMPREENDEDORISMO NA INTERNET	31
3 METODOLOGIA	32
4 PROJETO AUXILI CARS	33
5 CONCLUSÕES	50
REFERÊNCIAS	51
ANEXOS	53

1. INTRODUÇÃO

Com o crescimento acelerado da rede mundial de computadores, a internet, a comodidade de não precisar sair de casa para comprar produtos tornou-se hábito simples para todos que tem acesso a esse serviço.

O sistema especialista de apoio à decisão é a ferramenta mais indicada para a escolha do produto certo para o cliente certo, pois o consumidor típico das lojas virtuais é bem informado, tem boa formação educacional e capacidade de discernimento para saber o que é bom pra si. Mais do que tentar convencê-lo a realizar compras, é necessário fornecer o contexto e as informações que o ajudem a decidir. O produto x é um bom produto para um determinado cliente, porém, o produto y atenderia melhor aos seus anseios e poderia estabelecer uma relação custo-benefício mais interessante para ele.

As lojas virtuais devem, então, auxiliar esse cliente a tomar a melhor decisão, explicitando as reais características de cada produto e seus benefícios, para cada necessidade específica. Na medida em que o objetivo não é apenas vender o produto, mas ganhar um cliente que vai retornar várias vezes, o foco deve ser, sempre, maximizar o benefício oferecido a ele. (FELIPINI et al, 2004).

1.1 JUSTIFICATIVA

Os Sistemas Especialistas são utilizados em diversas áreas, cabe ao programador do curso de Ciência da Computação, analisar, codificar, implementar e usar as ferramentas adequadas para que a decisão tomada, seja a melhor possível, atendendo a exigência do cliente que estará comprando o produto.

Tem como função principal a obtenção do conhecimento e sua representação em sistemas computacionais, onde residirão as regras e os fatos que representam os processos decisórios para tomadas de decisão.

O Sistema Especialista contém o conhecimento de um especialista em determinada área de atuação, ele pode tomar suas próprias decisões. Já os sistemas de apoio a decisão, que possuem métodos de raciocínio limitados a uma única consideração, exigem efetivamente a presença daquele que toma a decisão, uma vez que não possuem o conhecimento adicional para este fim. Partindo deste princípio, o Sistema Especialista serviria como estrutura básica quanto à tomada de decisão (RAMOS, 1995).

A pergunta a qual busca-se responder neste trabalho conforme seus objetivos traçados é: Qual é a melhor forma no auxílio de compra de automóveis através de um site?

1.1.1 OBJETIVOS

No decorrer do estudo, foi necessário encontrar meios para responder as perguntas principais, foi determinado um objetivo geral seguido de seus objetivos específicos para tornar plausível o resultado do trabalho.

1.1.2 Objetivo Geral

Desenvolvimento de uma ferramenta on-line para atender a necessidade de auxílio à compra de automóveis novos pela internet.

1.2.2 Objetivos Específicos

Os objetivos específicos consistem em atender ao público que acessar a página, no direcionamento em que tomar para compra do automóvel, contendo as informações principais tais como:

- ✓ Opções de escolha das características dos automóveis utilizando a busca pelas principais montadoras de carros comercializados no Brasil.
- ✓ Opções de escolha de acordo com o perfil do comprador.
- ✓ Capacidade de cruzar informações de links na busca para compras.

Facilitando assim principalmente o auxílio aos usuários em geral, focando nas pessoas com menos conhecimento na área automobilística.

1.3 Cronograma

O seguinte cronograma foi utilizado para o desenvolvimento deste trabalho.

ATIVIDADES REALIZADAS	Jul	Ago	Set	Out	Nov	Dez
Pesquisa						
Desenvolvimento e apresentação do projeto						
Implementação do Sistema						
Ajustes no sistema						
Entrega e defesa do TCC II à banca avaliadora						

Tabela 1 – Cronograma do TCC 2 (Fonte: Próprio autor)

1.4 ESTRUTURA DO TRABALHO

Foram utilizadas as seguintes etapas para o desenvolvimento do trabalho: pesquisa de material bibliográfico, revisão bibliográfica, pesquisas on-line à páginas.

Na primeira etapa foi realizado um levantamento bibliográfico, coletando dados e informações necessários para o início do TCC, fundamentando a teoria, proporcionando um embasamento técnico. Feita pesquisas em livros, páginas da *Web*, trabalhos de conclusão de curso, artigos, entre outros.

2. REVISÃO DA LITERATURA

2.1 SISTEMAS ESPECIALISTA DE APOIO À DECISÃO

Sistemas especialistas são sistemas que têm um conhecimento específico com uma grande profundidade sobre áreas limitadas do conhecimento, com um poder de resolução de problemas bastante parecido com o especialista humano. Os sistemas especialistas caracterizam-se pela amplitude do conhecimento, no limite da perícia, simplificando a conclusão esperada, com habilidade para explicar o que e porque estão fazendo ao “pensar” e “decidir” (RABUSKE, 2000).

Apesar do desempenho de atividades semelhantes em alguns casos, as características dos sistemas especialistas e dos peritos reais divergem em muitos pontos. Os sistemas

especialistas nem sempre podem substituir os peritos, devido a algumas limitações inerentes. Entre elas, está a falta de criatividade, sensibilidade e adaptabilidade diante do fato total (SAVARIS, 2002).

Assim, um perito humano pode reorganizar informações e utilizá-las para sintetizar novo conhecimento, pode ainda, manusear eventos inesperados usando a imaginação e novas abordagens, servir-se de analogias com um domínio diferente de conhecimento. Já o Sistema Especialista tem vantagens como: o conhecimento não é perdido ao longo do tempo, se não for exercitado, imunidade a erros por fatores subjetivos como estado emocional, cansaço, falta de atenção e preconceito. A Tabela 1 apresenta uma comparação entre o conhecimento humano e o conhecimento artificial.

CONHECIMENTO HUMANO	CONHECIMENTO ARTIFICIAL
Precível	Permanente
Difícil de Transferir	Fácil de Transferir
Difícil de Documentar	Fácil de Documentar
Imprevisível	Consistente
Caro	Razoável
Discriminatório	Imparcial
Social	Individualizado
Criativo	Sem Inspiração
Adaptável	Inflexível
Enfoque Amplo	Enfoque Restrito
Baseado em Senso Comum	Técnico

Tabela 2- Comparação entre Conhecimento Humano e Conhecimento Artificial (Fonte: Savaris, 2002)

2.1.1 Sistemas Especialistas X Sistemas Tradicionais

Um Sistema Tradicional tem por objetivo processar dados seguindo as bases algorítmicas impostas a ele pelo homem e dar uma resposta correta em um curto espaço de tempo, pois caso contrário o sistema provavelmente errou (SAVARIS, 2002).

Um sistema Especialista, que tem sua base apoiada na heurística, tem como um dos objetivos solucionar problemas que não podem ser algoritmizados, e mesmo que pudessem demandariam um tempo de processamento que inviabilizaria o sistema. É muito importante estar ciente que um sistema especialista pode não levar a solução alguma, ou até a uma solução distorcida, ou seja, não é obrigatoriamente chegar a uma solução correta. O sistema

pode errar, mas seu erro ou não, a resposta estará dentro de determinadas circunstâncias que são justificadas pelo próprio sistema. Assim, Sistemas Especialistas processam o conhecimento como o perito do domínio de trabalho e este por ser um homem, não possui resposta para todas perguntas (SAVARIS, 2002).

2.1.2 Estrutura dos Sistemas Especialistas

Os sistemas especialistas apresentam a seguinte estrutura: o Núcleo do Sistema Baseado em Conhecimento (NSBC), a Base de Conhecimento (BC), a Memória de Trabalho (MT) e a Interface com o usuário (REZENDE, 2003). Através da Figura 1 podem-se notar as relações entre as partes da estrutura:

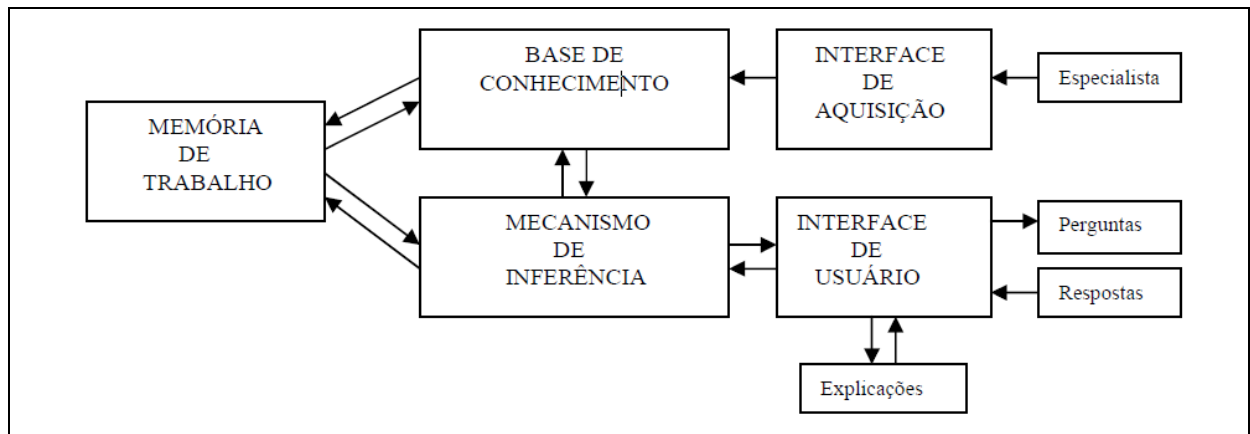


Figura 1 – Estrutura dos Sistemas Especialistas (Fonte: Rezende, 2003)

2.1.3 Núcleo do Sistema baseado em conhecimento

O NSBC é composto por 3 partes: Módulo Coletor de Dados, Motor de Inferência e Módulo de Explicações.

O Módulo Coletor de Dados é a parte responsável de intermediar a interação com o usuário de forma a obter os dados necessários para que o motor de inferência possa “raciocinar” (REZENDE, 2003).

O Motor de Inferência tem como principais características o método de raciocínio, estratégia de busca, resolução de conflito e representação de incerteza. Utiliza para isto informações da Base de Conhecimento e as obtidas pelo Módulo Coletor de Dados (REZENDE, 2003).

Segundo Rabuske (2000) o Motor Inferência é em geral entendido como o interpretador e escalonador das regras, nos casos dos sistemas especialistas que usam regras de produção. Já em outras formas de representação do conhecimento, pode estar ligado à estrutura de representação, sendo algo até um pouco difuso.

O Módulo de Explicações tem a responsabilidade pelas explicações e justificativas das conclusões que serão geradas ao usuário. Estas devem ser de alto nível.

2.1.4 Base de Conhecimento

A Base de Conhecimento (BC) contém conhecimento sob a forma de regras de produção, quadros, redes semânticas, ou outra forma qualquer. Ela tem um somatório de fatos, heurísticas e de crenças, é capaz de certos tipos de controle sobre si mesma, podendo suprir algumas informações ausentes. Ela está ligada ao Motor de Inferência e a Memória de Trabalho (RABUSKE, 2000). A BC contém o conhecimento de algum especialista no domínio da aplicação, necessário para achar a solução do problema.

A BC é a alma do sistema especialista, sendo ela que dá as características do sistema. Este terá o conhecimento estrito do que for colocado em sua BC. Um novo fato pode modificar todo um processo de inferência de acordo com as regras existentes sobre ele e também pelos novos fatos gerados pela avaliação destas regras.

A etapa de elaborar a BC é uma etapa muito importante e se caracteriza por ser longa, pois irá conter todo o conhecimento necessário para atingir o resultado proposto, mas mesmo tentando adquirir o conhecimento completo, nem sempre isso é conseguido. Estas limitações devem ser contornadas pelo Sistema Baseado em Conhecimento, que mostrará respostas aceitáveis ao usuário. Podem existir casos em que o sistema chegará a conclusões divergentes, então o sistema deverá analisar cada opção e emitir a resposta mais viável (REZENDE, 2003).

2.1.5 Memória de Trabalho

A Memória de Trabalho (MT) é uma área da memória onde o sistema vai gravando e apagando os dados que vai usando no processo de inferência, até chegar a uma solução. A MT

guarda informações, fatos e estruturas de suporte ao funcionamento do sistema, quando este efetua raciocínios (RABUSKE, 2000). Assim, tem a função de registrar todas as respostas do usuário, gerar conclusões intermediárias e armazenar a linha de raciocínio seguida pelo sistema.

Algumas das vantagens da MT são: oferecer ao usuário a linha de raciocínio das conclusões alcançadas pelo sistema; unicidade das perguntas realizadas ao usuário, ou seja, elimina a possibilidade de se fazer mais de uma vez a mesma pergunta; evita a repetição das sequências de raciocínio para chegar a conclusões intermediárias (REZENDE, 2003).

2.1.6 Interface

Um SE deve apresentar uma interface bem projetada, analisando ainda o grau de familiarização com o domínio de trabalho do sistema e com os computadores de uma maneira geral. As telas mostradas ao usuário devem ser fáceis de compreender e as explicações bastante claras. Além disso, a disponibilidade de alguns tipos de interface, conforme o tipo de usuário, seja ele especialista ou iniciante é uma característica interessante. Assim como o fato de poder parar a execução em certo ponto e retomar sem precisar de reprocessamento; a possibilidade de alterar entradas e comparar as conclusões alcançadas; gerar mensagens de erros claras; capacidade de capturar e armazenar telas de execução (AZEVEDO, 2005).

2.1.7 Implementação de Sistemas Especialistas

Na construção dos SE podem-se utilizar Shells, linguagens de manipulação simbólica. Os Shells são ferramentas de alto nível, auxiliando no desenvolvimento de sistemas especialistas, uma vez que já possuem a estrutura de representação do conhecimento e mecanismos de inferência embutidos (AZEVEDO, 2005).

O Expert Sinta é um Shell de inteligência artificial, que utiliza regras de produção como modelo de representação do conhecimento e possui uma máquina de inferência compartilhada, com capacidade de construir telas e menus de forma automática, inserir o conhecimento através de um editor de base, tratamento probabilístico das regras de produção, e explicações da base de conhecimento (AZEVEDO, 2005).

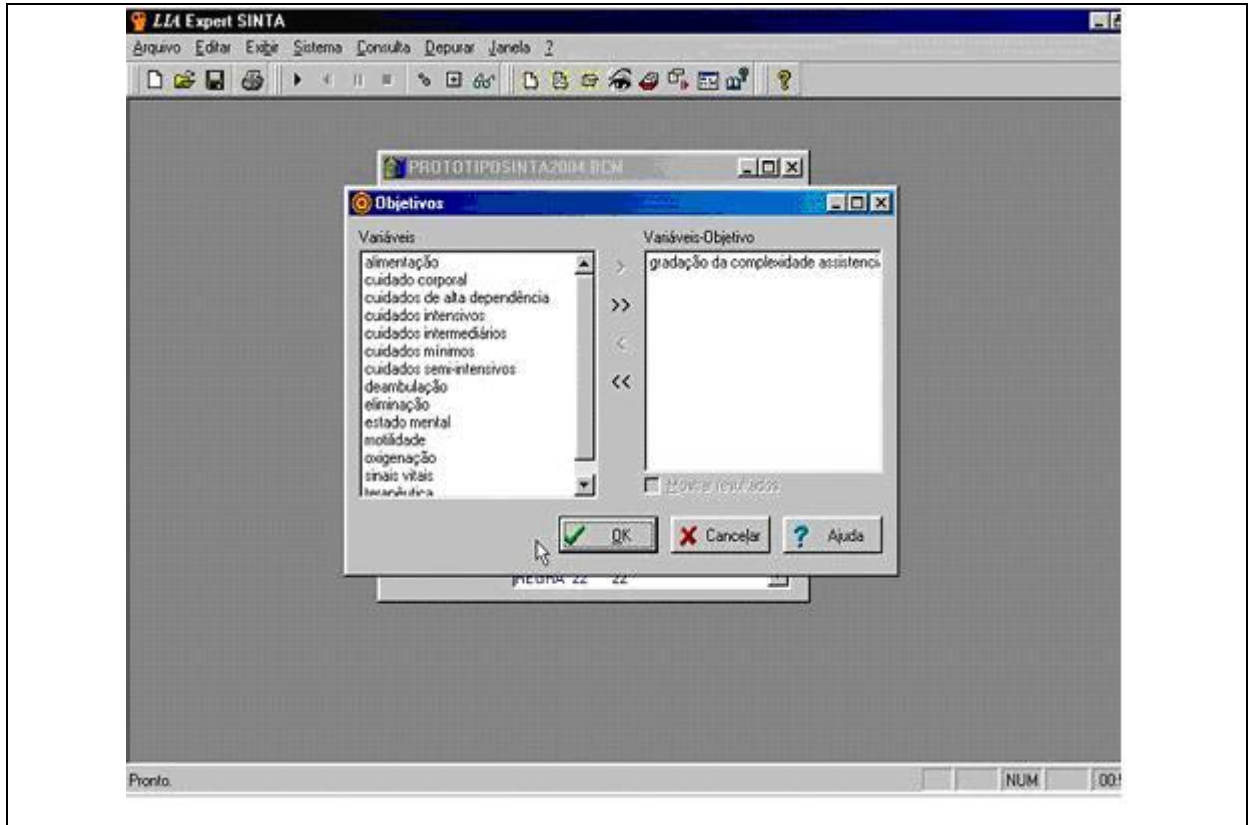


Figura 2 – Expert Sinta (Fonte: Internet, 2014)

STRESS é uma ferramenta Shell gráfica baseada em regras, que consiste em programas, que são: editor para construir os objetos e regras da base de conhecimento, máquina de inferência, duas ferramentas para monitoramento durante a execução da máquina de inferência, uma ferramenta para fazer o trace. Este Shell possibilita encadeamento para trás e para frente, utilização de fatores de confiança e da ferramenta de depuração, tem fácil instalação, mas tem funcionamento bem complexo com pouca documentação na *Web* (AZEVEDO, 2005).

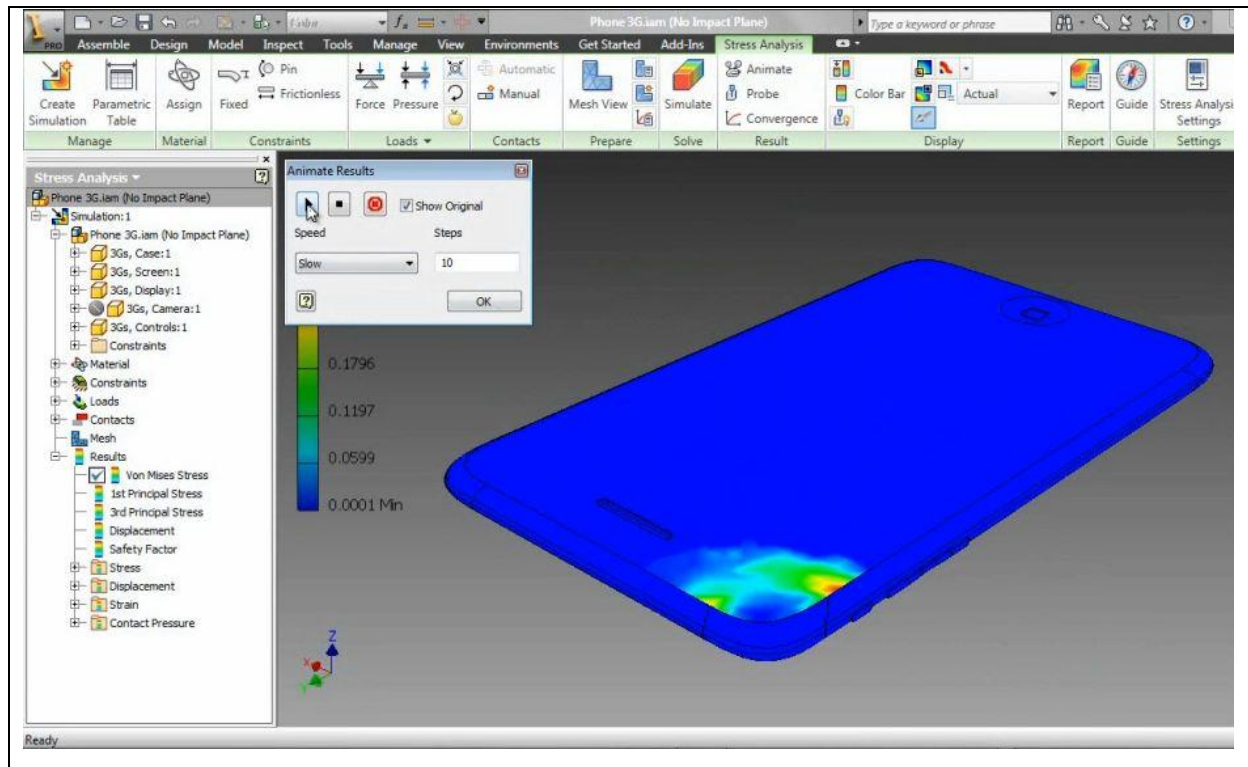


Figura 3 – Stress Analysis (Fonte: Internet, 2014)

JESS é uma estrutura desenvolvida nos Laboratórios Nacionais de Sandia, Canadá, e utiliza tecnologia Java da *Sun Microsystem*. Com esta estrutura é possível desenvolver Sistemas Especialistas baseados em regras, podendo ser acoplados a programas escritos na linguagem Java. Os métodos do Java podem ser chamados a partir do JESS, estender o JESS codificando em Java ou embarcar o JESS em uma aplicação Java (AZEVEDO, 2005).

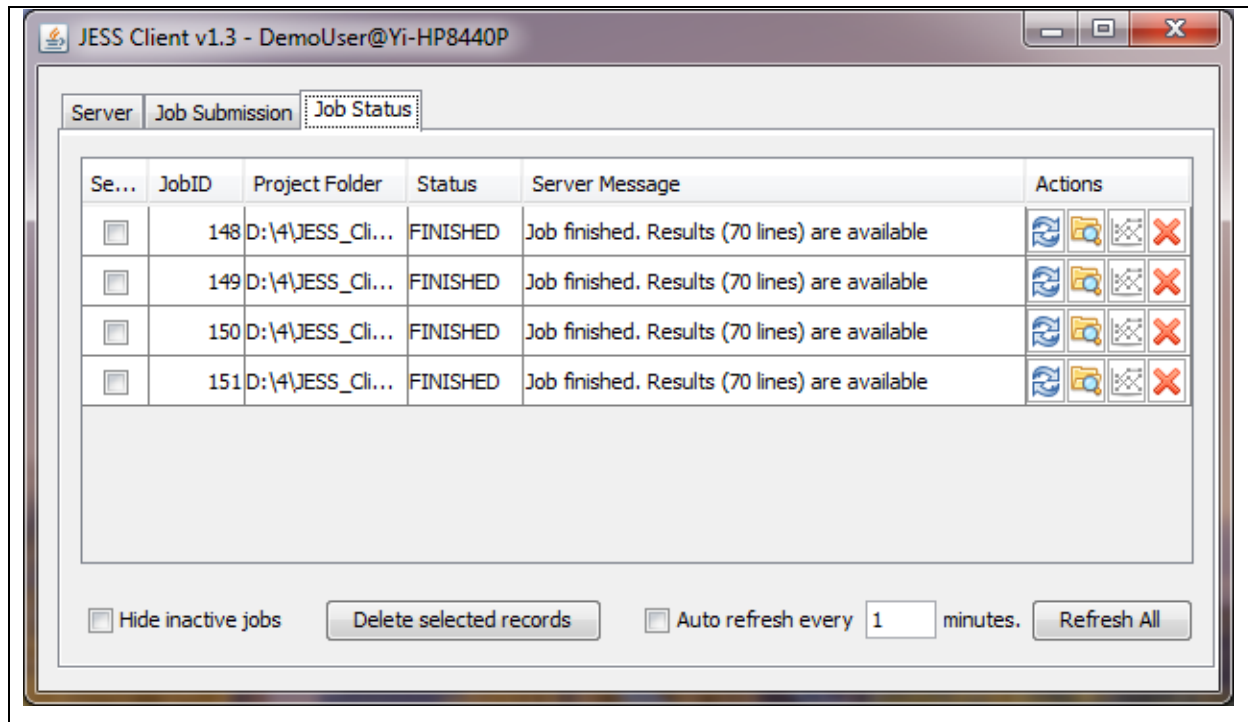


Figura 4 – Ferramenta JESS (Fonte: internet, 2014)

Algumas linguagens de manipulação simbólica como PROLOG e LISP podem fornecer um grande auxílio na construção de Sistemas Especialistas. A principal vantagem das linguagens em relação às ferramentas é a maior flexibilidade, mas o conhecimento prévio que se deve ter da linguagem adotada aparece como uma desvantagem (AZEVEDO, 2005).

2.2 INTELIGÊNCIA ARTIFICIAL

Esta é uma questão complexa, sem uma resposta bem definida, que tem intrigado biólogos, psicólogos e filósofos por séculos. (BEN COPPIN 2012).

Poder-se-ia certamente definir inteligência pelas propriedades de solucionar problemas, de responder a questões, de gerar planos e assim por diante. Talvez seja mais difícil definir a diferença entre a inteligência exibida por humanos e aquela exibida por golfinhos e macacos. (BEN COPPIN 2012).

Para Rabuske (1995) “Inteligência Artificial é o resultado da aplicação de técnicas e recursos, especialmente de natureza não numérica, viabilizando a solução de problemas que exigiriam do humano certo grau de raciocínio e perícia”.

Inteligência Artificial é o estudo dos sistemas que agem de um modo que a um observador qualquer pareceria ser inteligente (BEN COPPIN 2012).

Esta definição é boa, mas na verdade ela não abrange o todo da IA. Em muitos casos, técnicas de IA são utilizadas para solucionar problemas relativamente simples ou problemas complexos que fazem parte de sistemas complexos (BEN COPPIN 2012).

Segundo Winston (1987), “Inteligência Artificial é o estudo de conceitos que permitem aos computadores serem inteligentes”. A Inteligência Artificial nada mais é do que a tentativa de tornar o computador um pouco mais próximo do ser humano, dito inteligente.

2.2.1 IA Forte e IA Fraca

Os seguidores de IA forte acreditam que, dispondo de um computador com suficiente capacidade de processamento e fornecendo a ele suficiente inteligência, pode-se criar um computador que possa literalmente pensar a ser consciente do mesmo modo que um humano é consciente (BEN COPPIN 2012).

Muitos filósofos e pesquisadores da IA consideram esta visão como infundada e até mesmo absurda. A possibilidade de criar um robô com emoções e genuína consciência é aquela que é frequentemente explorada no âmbito da ficção científica, mas é raramente considerada como um objeto da Inteligência Artificial (BEN COPPIN 2012).

Por outro lado, a IA fraca é simplesmente a visão de que comportamento inteligente pode ser modelado e utilizado por computadores para solucionar problemas complexos. Este ponto de vista defende que apenas o fato de um computador agir inteligentemente não prova que ele seja verdadeiramente inteligente no sentido humano (BEN COPPIN 2012).

Segundo Rich E. Knight (1993), são difíceis os problemas que a IA abrange, por esse motivo existem técnicas específicas para a solução de uma variedade destes problemas. As técnicas de IA são métodos que exploram o conhecimento a ser representado, de maneira a capturar as generalizações, indicando situações que compartilham propriedades importantes e grupadas, seja compreendida pelas pessoas que o fornecem; seja modificada com facilidade para correção de erros, refletindo sobre as mudanças do mundo; possa ser usada em inúmeras situações.

2.2.2 Métodos Fracos e Métodos Fortes

Os métodos fracos da IA usam sistemas tais como lógica, raciocínio automatizado e outras estruturas gerais que podem ser aplicadas a uma ampla gama de problemas, mas que não necessariamente incorporam qualquer conhecimento genuíno sobre o mundo do problema que estão sendo solucionado. (BEN COPPIN, 2012).

Por outro lado, solucionar problemas por métodos fortes depende de um sistema de sólidos conhecimento sobre o seu mundo e sobre os problemas que ele deve encontrar. Solucionar problemas por métodos fortes depende de métodos fracos, pois um sistema com conhecimento, sem alguma metodologia para lidar com este conhecimento, é inútil. (BEN COPPIN, 2012).

As primeiras pesquisas em Inteligência Artificial concentravam-se em métodos fracos. O *General Problem Solver* (GPS – Solucionador de Problemas Genéricos) de Newell e Simon, foi uma tentativa de usar métodos fracos para construir um sistema que pudesse solucionar uma ampla gama de problemas gerais. O fato de esta abordagem ter finalmente falhado levou a percepção que era necessário mais do que simples representações e algoritmos para fazer a Inteligência Artificial funcionar: conhecimento era o ingrediente-chave. (BEN COPPIN, 2012).

Em muitas situações os métodos fracos são ideais para solucionar problemas. Entretanto, o acréscimo de conhecimento é quase sempre essencial para construir sistemas que sejam capazes de lidar inteligentemente com novos problemas; se o objetivo é construir sistemas que pareçam ter comportamento inteligente, então métodos fortes são certamente essenciais. (BEN COPPIN, 2012).

2.2.3 Raciocínio Baseado em Casos

Conforme Rich E. Knight (1993), os sistemas de Raciocínio Baseado em Casos (RBC), são sistemas de computadores que solucionam problemas novos por analogia com problemas antigos, tendo como ponto forte uma enorme biblioteca de casos em lugar de um conjunto de princípios básicos.

Para atingir o sucesso, os sistemas de RBC têm de responder às seguintes perguntas:

- Como os casos são organizados na memória?
- Como os casos relevantes são recuperados na memória?
- Casos anteriores podem ser adaptados a novos problemas?

- Como os casos são adquiridos originalmente?

Para aumento da eficiência de utilização da memória, precisa-se de um rico mecanismo de indexação. O objetivo é indexar episódios passados pelas características que estão presentes neles.

Devido à variação das características importantes de domínio para domínio, um sistema RBC genérico deve ser capaz de aprender o conjunto correto de índices com sua própria experiência.

O resultado do processo de recuperação é quase sempre um conjunto de casos. A etapa seguinte é selecionar o melhor caso e adaptá-lo à situação atual. Um método de escolher o melhor caso é usar as heurísticas da preferência. Exemplos:

- Preferência Orientada por Objetivos.
- Preferência por Características Notáveis.
- Preferência por Especificidade.
- Preferência por Frequência.
- Preferência por Recenticidade.
- Preferência por Facilidade de Adaptação.

2.2.4 Seleção do *best match*

Conforme Lee (1996) apud Fernandes (2003) “a seleção do *best match* é a etapa final da recuperação”. É esta etapa que define o que será apresentado como sendo a solução para o problema de entrada. A seleção dos casos pode ser feita por heurística ou por regras, contando com a participação de usuários onde contribuir com informações adicionais a fim de obter decisão do sistema.

2.3 JAVA

2.3.1 Histórico do Java

A *Sun* anunciou o Java formalmente em uma conferência do setor em maio de 1995. O Java chamou a atenção da comunidade de negócios por causa do enorme interesse na *Web* (os computadores que fornecem o conteúdo que vemos em nossos navegadores da *Web*), fornecer

aplicativos para dispositivos voltados para o consumo popular (como telefones celulares, *paggers* e PDAs) e para muitos outros propósitos.

Java utiliza herança que permite altos graus de reutilização de código. A herança é dividida em:

- a) Estrita
- b) Não Estrita
- c) Múltipla

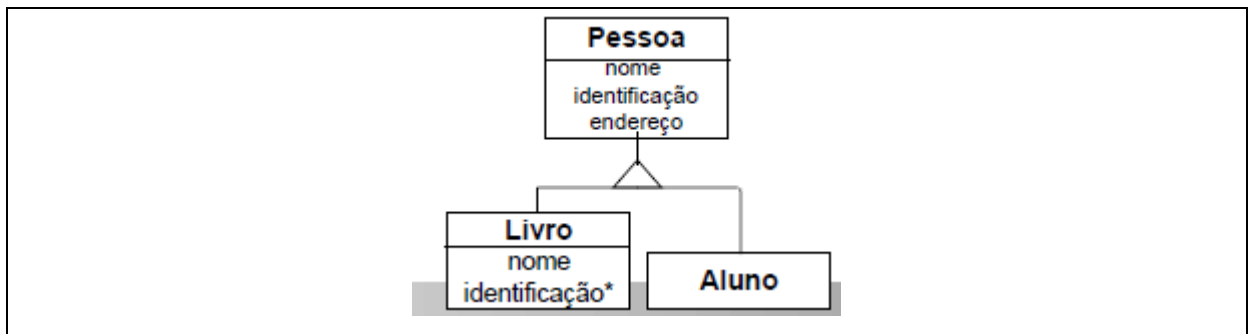


Figura 5 - Herança não estrita (Fonte: Info UFPA, 2007)

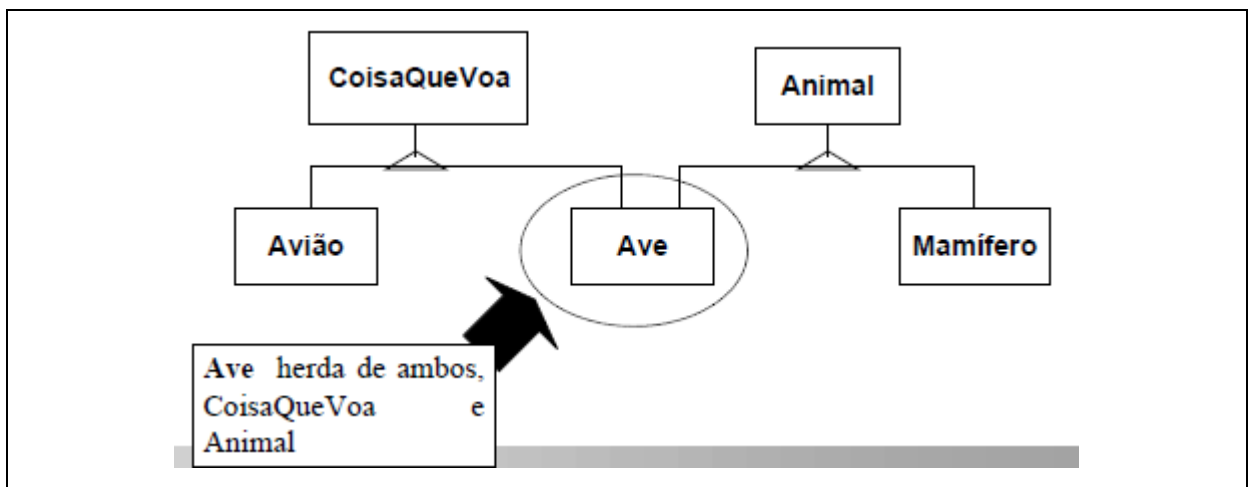


Figura 6 - Herança Múltipla: (Fonte: Info UFPA, 2007)

2.3.2 Polimorfismo

É a habilidade de variáveis terem mais de um tipo. Funções são ditas polimórficas quando seus operandos podem ter mais de um tipo.

2.3.3 Desenvolvimento Orientada à Objetos

- a) **Análise Orientada a Objetos:** É o processo de construção de modelos do domínio do problema, identificando e especificando um conjunto de objetos que interagem e comportam-se conforme os requisitos estabelecidos para o sistema.
- b) **Projeto Orientado a Objetos:** É o processo de geração de uma especificação detalhada do *software* a ser desenvolvido, de tal forma que esta especificação possa levar a direta implementação no ambiente alvo.
- c) **Programação Orientada a Objetos:** É um modelo de programação que baseia-se em conceitos como classes, objetos, herança, etc. Seu objetivo é a resolução de problemas baseada na identificação de objetos e o processamento requerido por estes objetos, e então na criação de simulações destes objetos. A programação é obtida através da definição de classes e criação de hierarquias, nas quais propriedades comuns são transmitidas das superclasses para as subclasses através do mecanismo de herança.

```

1 package com.Singleton;
2 /**
3  *
4  * @author mazzi
5  * Exemplo do uso de Singleton em Java.
6  */
7 public final class Janela {
8
9     //Atributos (variáveis) privados que NAO serao "visiveis" fora da classe//
10    private static final Janela INSTANCE = new Janela();
11    private static final String teste = "Ola Singleton";
12
13    //Atributos (variáveis) publicos que são visiveis em qualquer lugar do sistema//
14    public static final String Singleton = "Esse eu vejo fora";
15
16    // Construtor privado.
17    private Janela() {
18        // Operações de inicialização da classe
19    }
20
21    // Método público estático realizando o primeiro e unico acesso necessario nesse tipo de aplicacao.//
22    public static Janela getInstance(){
23        return INSTANCE;
24    }
25
26    // Aqui vou poder executar esse método de qualquer lugar do meu projeto//
27    public static void Abrir(){
28        System.out.println("Abriu a janela....");
29    }
30
31    // Aqui vou poder executar esse método de qualquer lugar do meu projeto//
32    public static void Fechar(){
33        System.out.println("Fechou a janela....");
34    }
35 }

```

1.) Criação da classe

2.) Atributos do tipo Private não são globais

3.) Método responsável pelo controle de uma única instância

4.) Métodos públicos que serão visíveis em todo o sistema

Figura 7 – Código Java (Fonte: Deitel, 2010)

2.4 PHP

A linguagem de programação PHP (atualmente simplesmente pela abreviatura de *Hypertext Preprocessor*, originalmente *Personal Home Page* em 1994) é uma linguagem interpretada criada em 1994 por Rasmus Lerdorf, inicialmente formada por um conjunto de scripts voltados à criação de páginas dinâmicas para monitorar o acesso ao seu currículo na internet. À medida que a ferramenta foi crescendo, Rasmus teve de escrever uma implementação em C, permitindo as pessoas desenvolverem de forma mais simples suas aplicações para a web. De acordo com *The Php Group* (2012), o PHP é uma linguagem de programação de ampla utilização, interpretada, que é especialmente interessante para desenvolvimento para a Web e pode ser mesclada dentro do código HTML.

A sintaxe da linguagem lembra C e Java. O objetivo principal da linguagem é permitir a desenvolvedores escreverem páginas que serão geradas dinamicamente rapidamente, permitindo aplicações avançadas utilizando conexões com os SGBDs mais comuns do mercado atualmente como o *MySQL*, *Oracle* e *PostGre*.

2.5 MYSQL

O *MySQL* é um banco de dados relacional com grandes vantagens para as aplicações Web, além de suportar muitos tipos de dados, é um *software* livre com base na GPL (entretanto, se o programa que acessar o *Mysql* não for GPL, uma licença comercial deverá ser adquirida). Ele utiliza seu próprio servidor de banco de dados para programar os acessos aos bancos de dados e tabelas criados e garantir sua segurança. Ao se instalar pela primeira vez o *MySQL*, um banco de dados chamado “*mysql*” é criado, que contém cinco tabelas: “db”, “host”, “user”, “tables_priv” e “columns_priv”. Essas tabelas são utilizadas para acessos destinados aos diferentes usuários e privilégios registrados no SGBD.

O banco de dados *MySQL* pode ser considerado um dos projetos de *software* de origem mais quente desde o Linux. É um sério competidor para os maiores sistemas de banco de dados existentes para aplicações de pequeno a médio porte. No seu lançamento, era apenas um mero substituto para o ultrapassado sistema de banco de dados *mSQL*. Como o *mSQL* mostrava falhas em formas de estabilidade e incapacidade de atender a grande

demanda, o *MySQL* mostrou-se mais robusto e eficaz que seu antecessor, superando as expectativas iniciais (FERRARI, 2007).

--protocol	Value	Connection Protocol	Allowable Operating Systems
tcp		TCP/IP connection to local or remote server	All
socket		Unix socket file connection to local server	Unix only
pipe		Named-pipe connection to local server	Windows only
memory		Shared-memory connection to local server	Windows only

Figura 8 – Protocolos de Comunicação utilizados em Windows e Linux (Fonte: Ferrari, 2007)

A figura 8 apresenta os protocolos de comunicação que são utilizadas em diferentes plataformas como o Windows e Linux. Para ambas as plataformas, o protocolo TCP é suportado, sendo amplamente utilizado para aplicações na internet.

2.6 NetBeans IDE 7.2

Uma IDE (Ambiente Integrado de Desenvolvimento) é um *software* que reúne ferramentas de apoio ao desenvolvimento de *software* com o objetivo de agilizar este processo, e o *software* NetBeans IDE é uma ferramenta completa para o desenvolvimento de sistema que se melhor se adéqua aos requisitos deste sistema. De acordo Oracle (2012), é um ambiente de desenvolvimento - uma ferramenta para programadores, que permite escrever, compilar, depurar e instalar programas. O IDE é completamente escrito em Java, mas pode suportar qualquer linguagem de programação. Existe também um grande número de módulos para estender as funcionalidades do IDE NetBeans. O NetBeans IDE é um produto livre, sem restrições à sua forma de utilização.

A quantidade de plugins disponíveis para o NetBeans e a quantidade de documentação disponível em português tornou a IDE mais apropriada para o desenvolvimento do protótipo do SE, além dos conhecimentos sobre a IDE já adquiridos em aulas.

O NetBeans suporta todas as linguagens necessárias para o desenvolvimento do protótipo como o suporte a PHP, XHTML, HTML5, CSS, *JavaScript* e facilidade de conexão com o banco de dados.

2.7 Servidor Apache

O servidor *Apache* é um dos servidores web mais utilizados no mundo. De acordo com Marcelo (2005), atualmente o *Apache* chega a 60% de utilização dos servidores internet. O Autor Marcelo (2005) relaciona algumas vantagens que este servidor web possui:

- a) Suporte a HTTP 1.1 para criação de virtual hosts baseados em DNS;
- b) Suporte a Secured Socket Layer (SSL) para transações seguras;
- c) Suporte a CGI's, Perl e PHP;
- d) Suporte a autenticação baseada em HTTP;
- e) Suporte a Server Side Includes (SSI);
- f) Suporte a Servlets Java;
- g) Logs Customizáveis;
- h) Configuração rápida e simples;

Além das vantagens citadas pelo autor, o *software* do Apache é um *software* livre, código aberto, e pode ser instalado em vários servidores diferentes.

2.8 jQuery

Muitos estão usando jQuery incluindo páginas de grandes portais na internet como Amazon, Google, Twitter, Bank of America, Netflix, Dell, Oracle, e muitos outros. Mesmo desenvolvedores web iniciantes que criam páginas web para se divertir, utilizam os últimos efeitos interessantes para seus sites, e jQuery é rápido e flexível. Assim como CSS (*Cascading Style Sheets*) separa características de visualização da estrutura de HTML, jQuery separa as características do comportamento da estrutura HTML, efeitos visuais, controles especiais e Ajax.

jQuery é uma biblioteca de JavaScript, popularmente o recurso simplifica a manipulação de eventos, animação e interações utilizando Ajax para desenvolvimento rápido de aplicações web. (HOLZNER, 2009).

O recurso *javascript* jQuery é de uso livre e está disponível para download na internet gratuitamente, atualmente na versão 1.7.2.

2.8.1 *jQueryUI*

O pacote jQueryUI – *jQuery User Interface* está disponível na forma de código livre para ser implementado na utilização do jQuery em páginas web com o objetivo de tornar as aplicações web mais agradável e dinâmicas. Algumas das funções disponíveis com este pacote envolvem caixas de diálogo internas móveis, abas de conteúdo dinâmico, botões estilizados, menus no formato “*accordion*” para web, e muitas outras funcionalidades que possibilitam um visual agradável, elegante para a página com sombras e realces.

2.9 Empreendedorismo na Internet

O consumidor definitivamente adotou a compra on-line e hoje cerca de um quarto da população brasileira compra regularmente pela Internet, devido principalmente à conveniência e a economia obtida. Isso significa uma excelente oportunidade para quem deseja montar um empreendimento neste novo canal de comercialização e tem o conhecimento necessário para fazê-lo (FELIPINI, 2013).

Em princípio, vender pela Internet não é muito diferente do que vender por meio de outro canal de comercialização. O fundamental é conhecer muito bem o seu público alvo e oferecer um produto que atenda efetivamente as necessidades desse público. Deverão ser consideradas na implantação e gestão do empreendimento e que podem fazer a diferença no final. Vamos ver três fatores críticos de sucesso para o empreendedor da Internet (FELIPINI, 2013).

a) Conhecer o consumidor online: o consumidor online típico tem melhor escolaridade que o consumidor tradicional. Ele é crítico, exigente e sabe que tem em suas mãos uma arma poderosa que é a possibilidade de usar a própria Internet para manifestar a sua insatisfação com um eventual serviço. Além disso, esse consumidor tem outra arma poderosíssima a sua disposição: o poder da informação. Na Internet é muito mais fácil e rápido conseguir informações sobre qualquer produto desejado e inclusive comparar o valor pelo qual esse produto esta sendo ofertado pelos diferentes players (FELIPINI, 2013).

b) Planejamento nunca foi tão importante: é claro que pesquisar, analisar e colocar no papel todas as variáveis relevantes sobre o projeto sempre foi um aspecto determinante para o empreendedor de sucesso, uma vez que isso previne gastos desnecessários, evita atrasos e amplia perspectivas de aproveitamento de oportunidades. No entanto, o que ocorre hoje na Internet é que o grau de competição se ampliou enormemente, diminuindo as possibilidades

de correção de erros, assim como as margens de lucro, e aumentando os riscos do negócio (FELIPINI, 2013).

c) A escolha do Nicho de Mercado, decisão crucial: pode-se dizer sem medo de errar que um dos principais determinantes de sucesso é a definição do que será vendido, e para quem. Um empreendedor desavisado que entra num nicho carregado de grandes players já estabelecidos tem pouquíssimas chances. Da mesma forma, um empreendedor que tentar um nicho com poucos concorrentes, mas cuja demanda pelos produtos vem decaindo gradativamente, ano após ano, está fadado ao fracasso (FELIPINI, 2013).

Com o sistema de apoio a decisão para compra de automóveis, o mesmo trará uma eficiente busca para o que realmente o cliente precisa, com isso trará o benefício de fazer uma compra correta focando naquilo que mais se adapta à sua necessidade.

3 METODOLOGIA

No desenvolvimento do trabalho foi realizada uma pesquisa nos principais sites de compra e venda de automóveis para levantar os dados necessários para o desenvolvimento da ferramenta do Sistema Especialista filtrando os campos mais importantes das páginas para facilitar o entendimento do usuário e melhores opções de escolha.

Para a realização deste trabalho foi efetuado pesquisa bibliográfica em livros de Sistemas Especialistas de Apoio à Decisão, Inteligência Artificial, livros de vendas pela internet, sites, artigos e materiais sobre a criação de *softwares* para atender tais necessidades levando em consideração ergonomia, eficiência, eficácia para que todos os usuários da página possam utilizar a ferramenta auxiliando a compra correta.

Conforme a autora Maria Isabel Faria, pesquisa bibliográfica é a operação com vista a obter por meios manuais ou informatizados referências bibliográficas específicas.

4 PROJETO “AUXILI CARS”

4.1 Definição da Tecnologia

A aplicação utilizada para a criação e desenvolvimento da página em *PHP* foi o *NetBeans 7.2.1*.

A aplicação *EasyPHP 14.1 VC11*, consiste em um pacote que inclui o servidor de linguagem script *PHP*, servidor Web Apache e o *MySQL*, assim como ferramentas de desenvolvimento: como o gerenciador de banco e o depurador *Xdebug*, onde a mesma gerencia toda a estrutura que envolve a criação de páginas e armazenamento de dados.

4.2 Implementação do Projeto

Para a implementação do Projeto foi utilizado a IDE para a codificação o *NetBeans 7.2.1*, banco de dados *MySQL Workbench 5.2* para armazenamento das tabelas, Microsoft Office Visio 2007 para criação de diagramas e a linguagem para criação da página foi a codificação em *PHP*.

A base de dados do projeto para a criação da página está localizada no seguinte endereço local da máquina:

C:\Program Files (x86)\EasyPHP-DevServer-14.1VC11\data\localweb\auxilicars
--

4.2.1 Estruturação do projeto

Model-view-controller (MVC), em português modelo-visão-controlador, padrão de arquitetura de software.

- Modelo (*model*) consiste nos dados da aplicação, regras de negócios, lógica e funções.
- Visão (*view*) pode ser qualquer saída de representação dos dados, como uma tabela ou um diagrama.
- O controlador (*controller*) faz a mediação da entrada, convertendo-a em comandos para o modelo ou visão. A utilização deste padrão para o desenvolvimento com mais legibilidade de código, reutilizando métodos e separando conceitos.

A figura abaixo mostra a estrutura de arquivos do projeto do *NetBeans*:

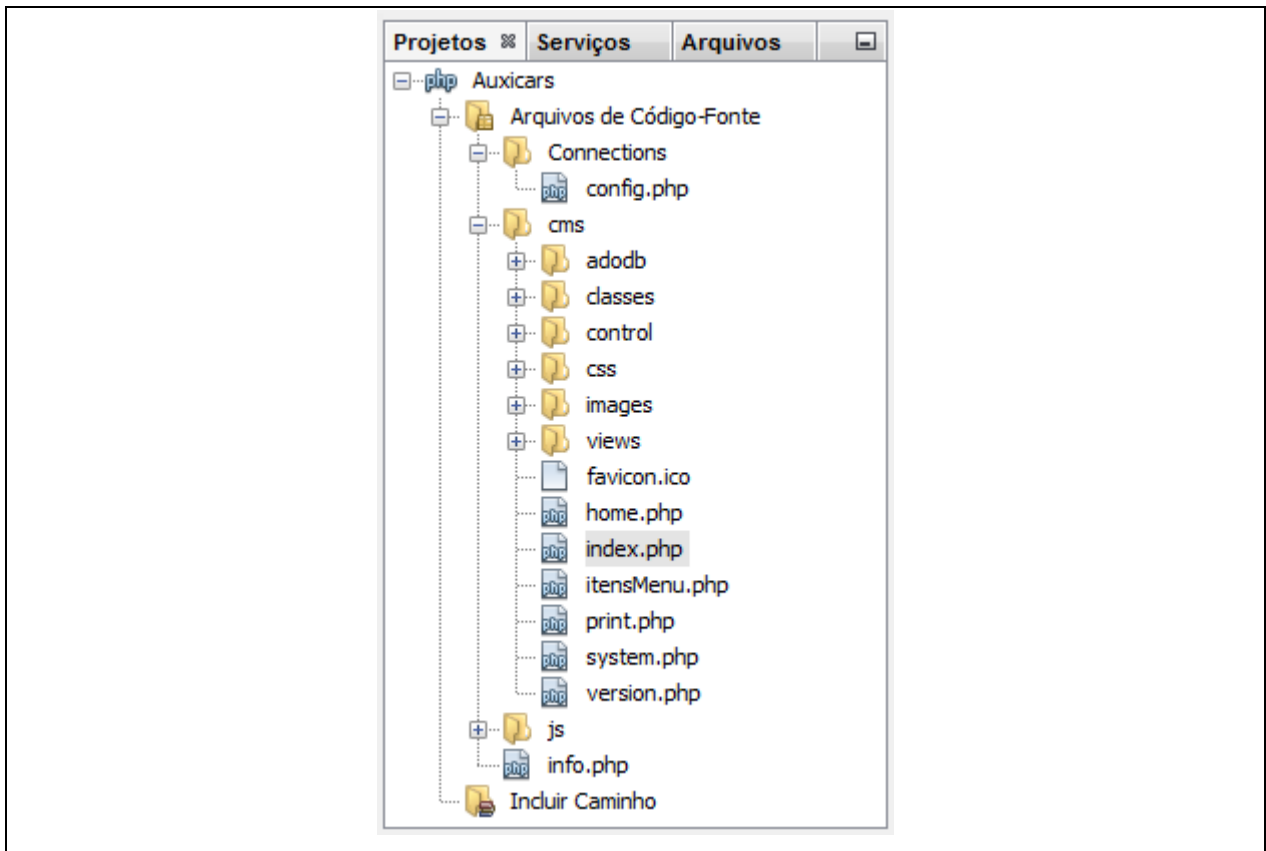


Figura 9 – Estrutura MVC do projeto *NetBeans 7.2.1* (Próprio autor)

A figura abaixo mostra a área de programação principal do projeto no *NetBeans*:

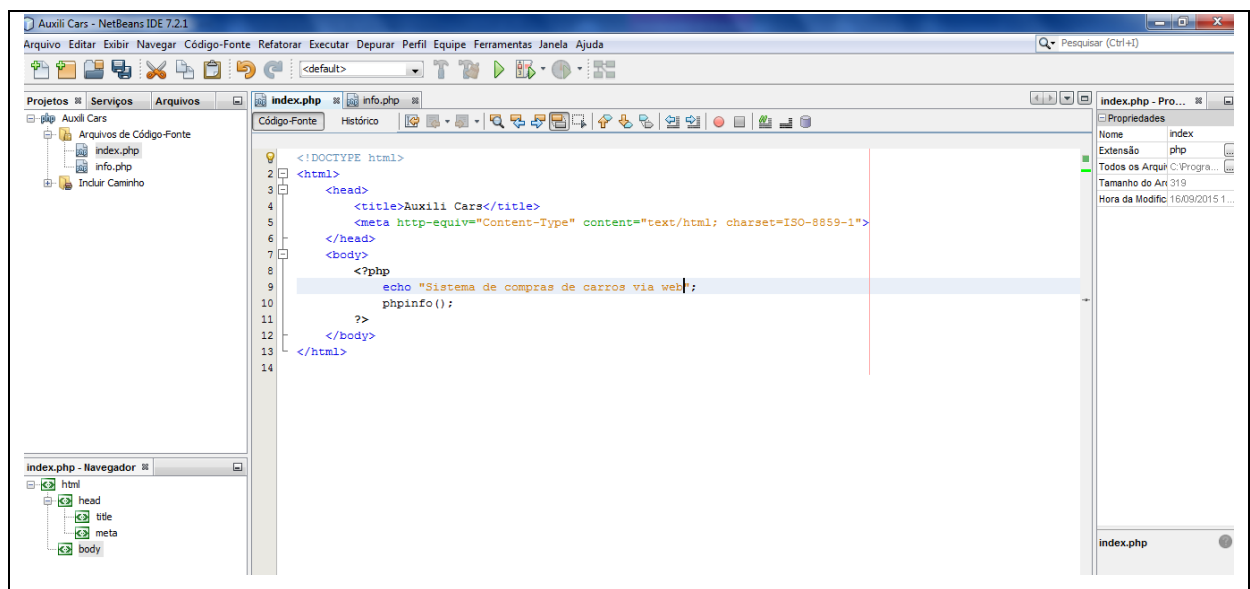


Figura 10 – Tela do projeto *NetBeans 7.2.1* (Próprio autor)

A figura a seguir apresenta estruturação do banco de dados do SGBD *MySQL*:

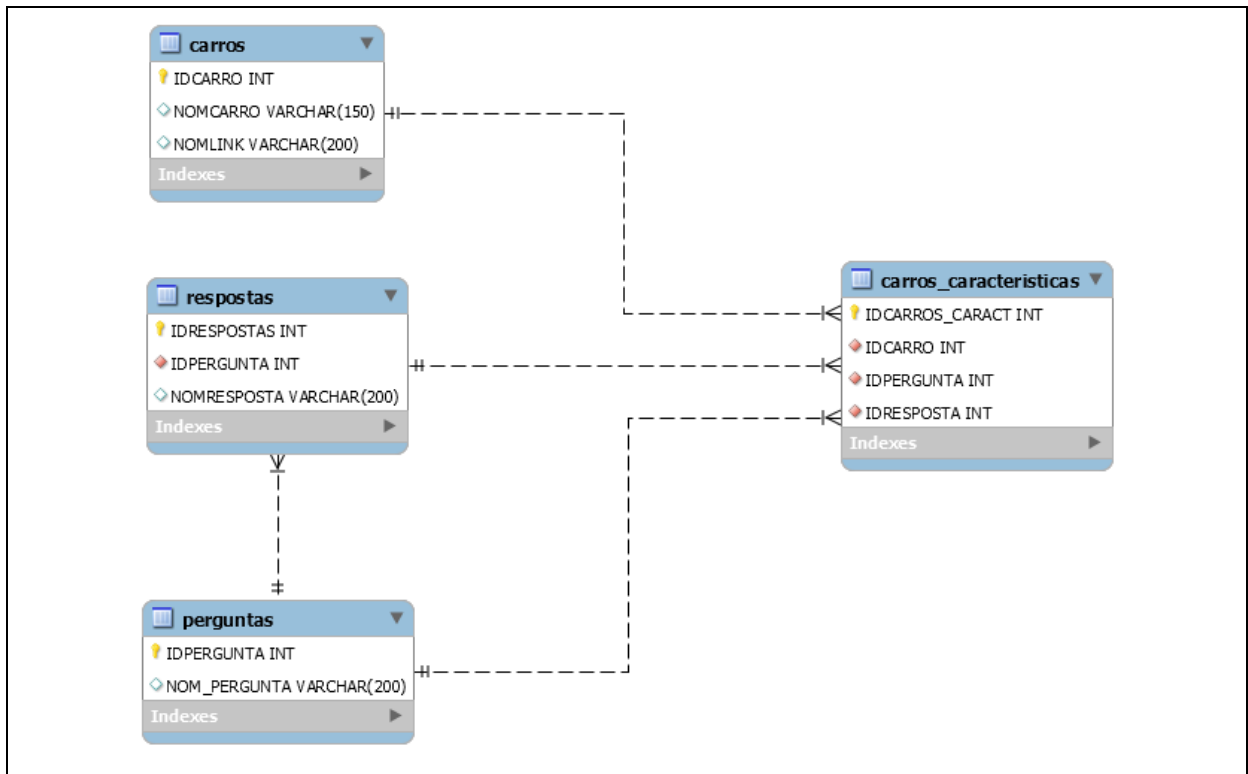


Figura 11 – Tela do projeto *MySql Workbench* (Próprio autor)

A figura a seguir mostra a aplicação *EasyPHP* rodando os servidores *Apache* e *MySQL*:

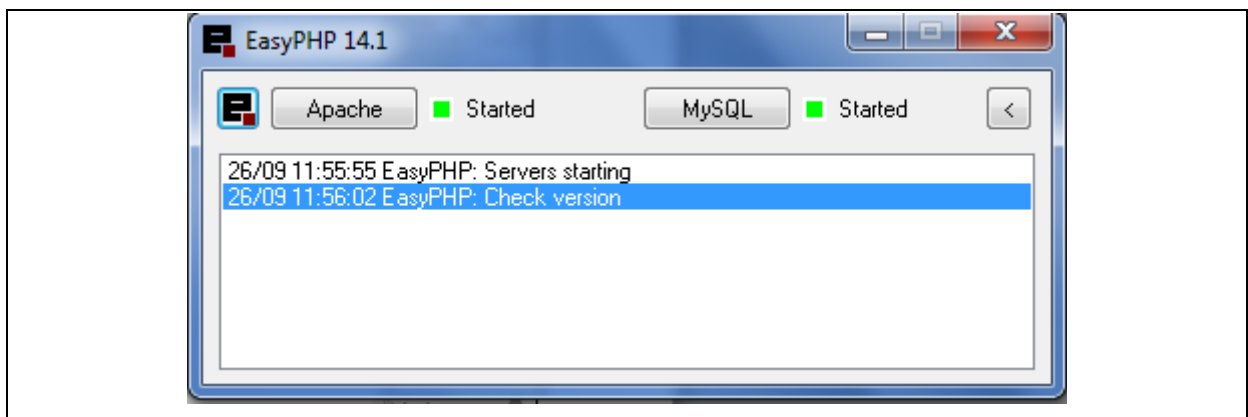


Figura 12 – Aplicação *EasyPHP* (Próprio autor)

A próxima figura apresenta a página da aplicação *EasyPHP*, através da qual direcionará para o servidor de aplicação Apache e banco de dados *MySQL*.

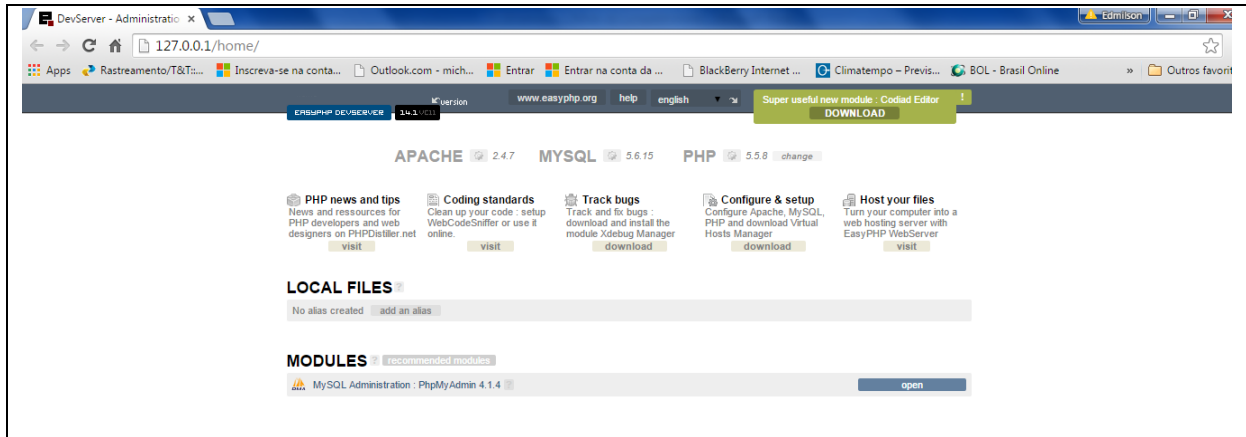


Figura 13 – Aplicação *EasyPHP* (Próprio autor)

A figura abaixo mostra o servidor *phpMyAdmin* com o banco de dados *MySQL* do projeto em execução:

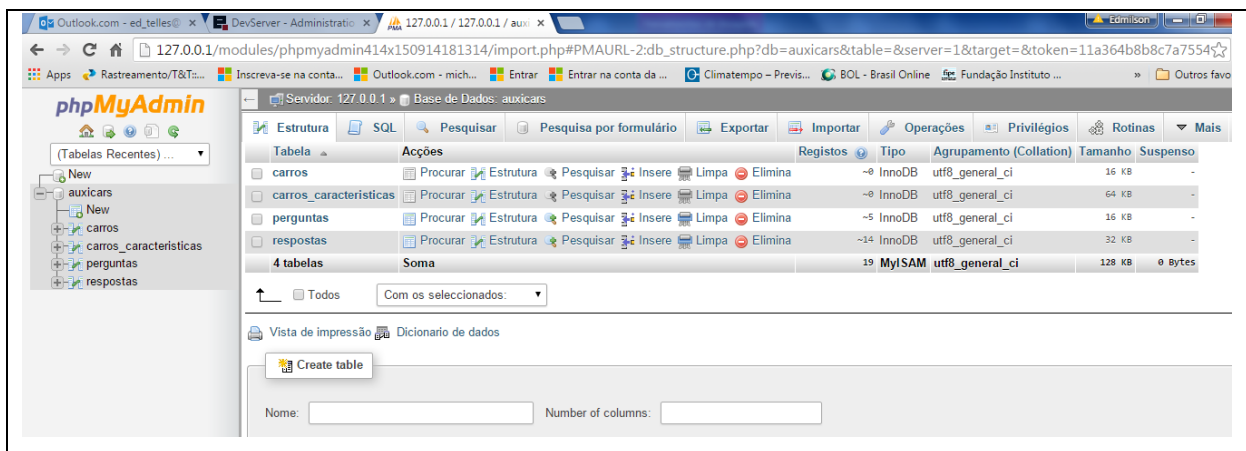


Figura 14 – Banco de dados *MySQL* (Próprio autor)

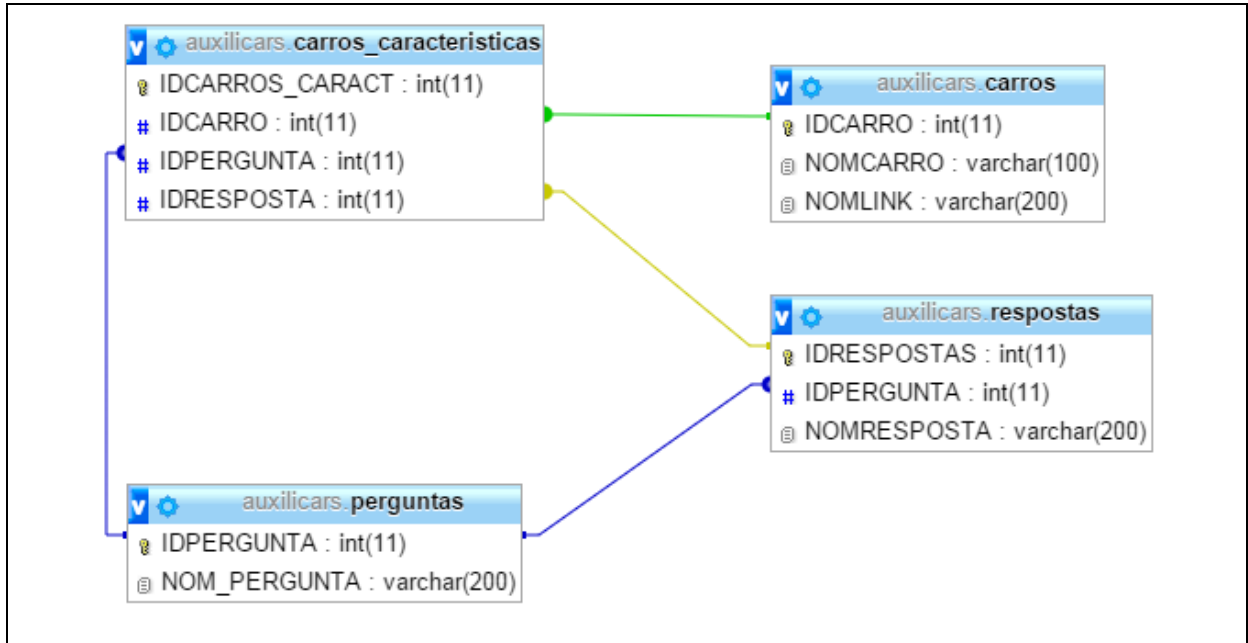


Figura 15 – Estrutura do banco de dados *MySQL* (Próprio autor)

Hardware utilizado no projeto:

- Notebook para implementação e documentação;
- Processador Intel Core I3 M380, 2,53 GHZ;
- 4 GB de memória RAM;
- HD 500 GB;

4.3 Modelagem de dados

Foi efetuada a estruturação do banco de dados do SGBD MySQL.

Usando a interface cliente MySQL Workbench 5.2 CE, foi modelada a estrutura com suas restrições, chave e dependências funcionais.

4.4 Diagramas

Será apresentado nessa parte do projeto os diagramas de Estudo de Caso, Classe e Fluxo de Dados.

4.4.1 Estudo de Caso

O administrador da página do sistema possui um acesso para cadastro geral e manutenção do sistema, realizando as inserções dos dados ao auxílio da tomada de decisão.

O usuário acessa a página e responde os questionamentos, retornando o resultado para o mesmo direcionando à tomada de escolha de acordo com seu perfil, conforme segue o diagrama a seguir.

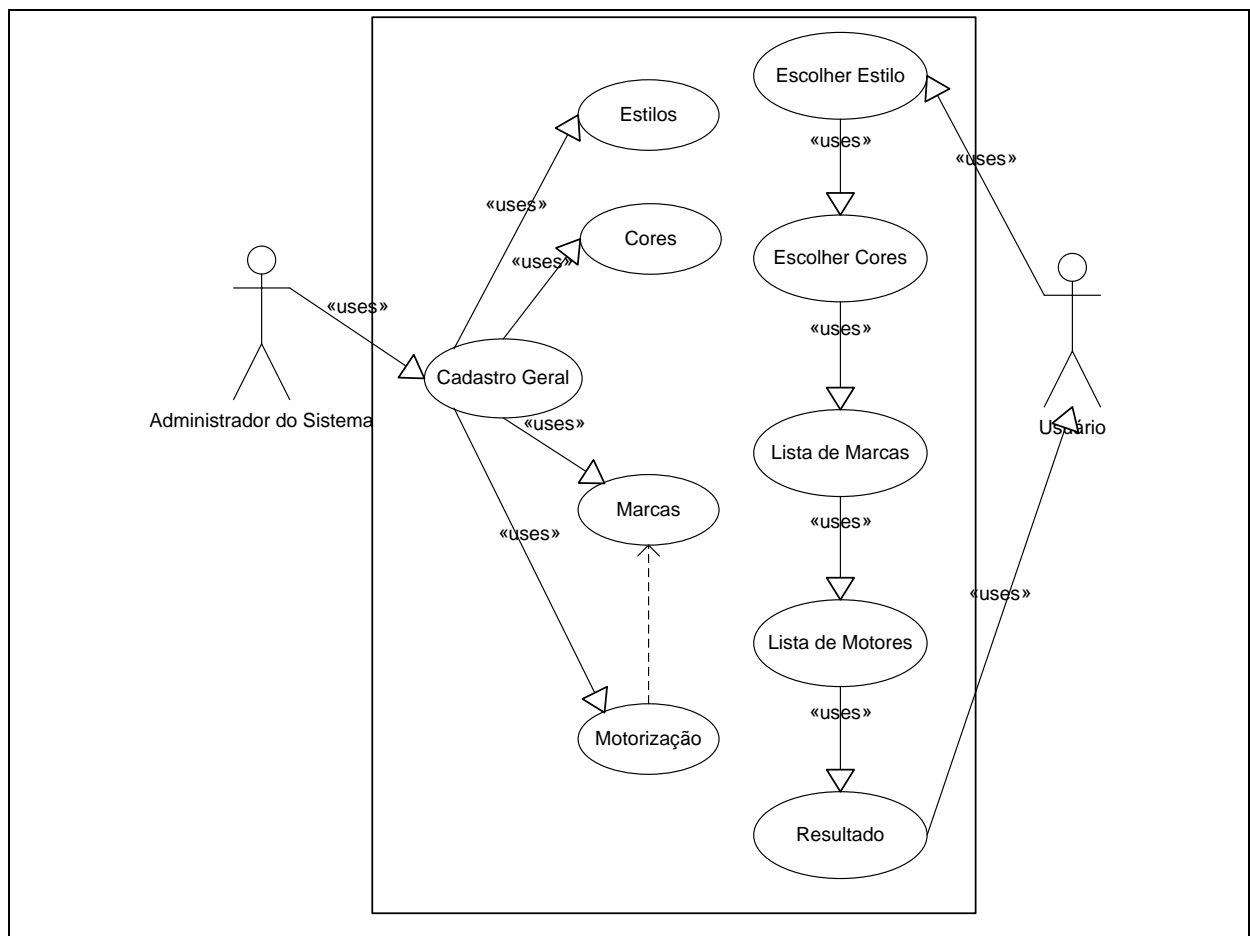


Figura 16 – Diagrama de Estudo de caso (Fonte: Próprio Autor)

4.4.2 Classe

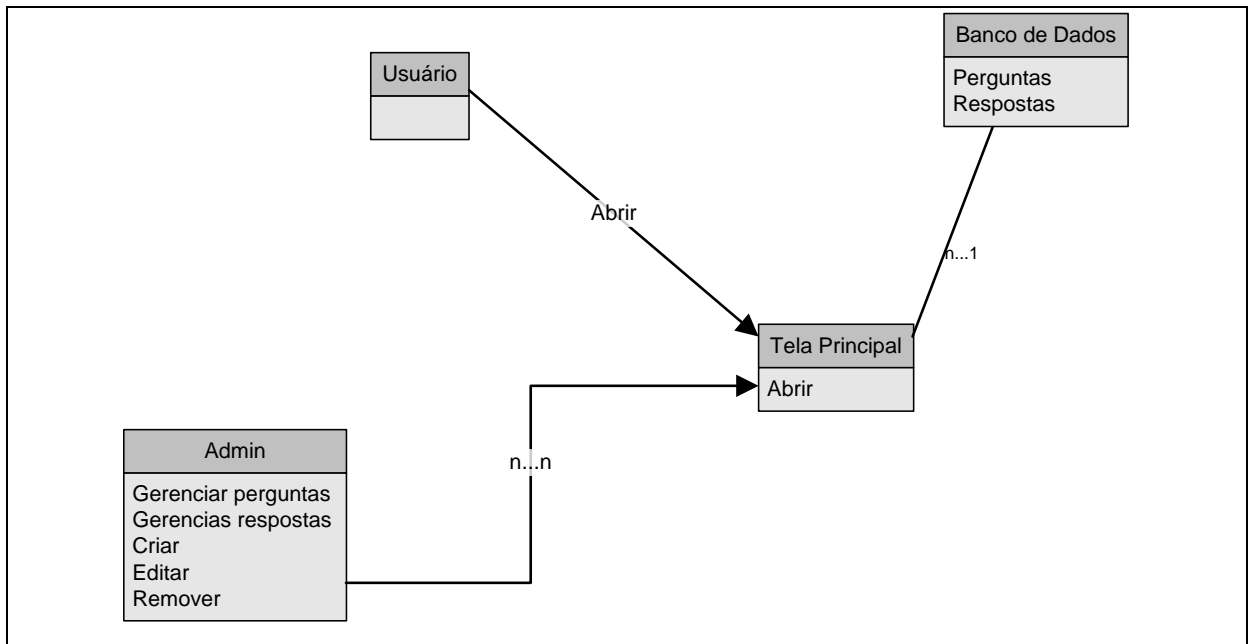


Figura 17 – Diagrama de Classe (Fonte: Próprio Autor)

4.4.3 Fluxo de Dados

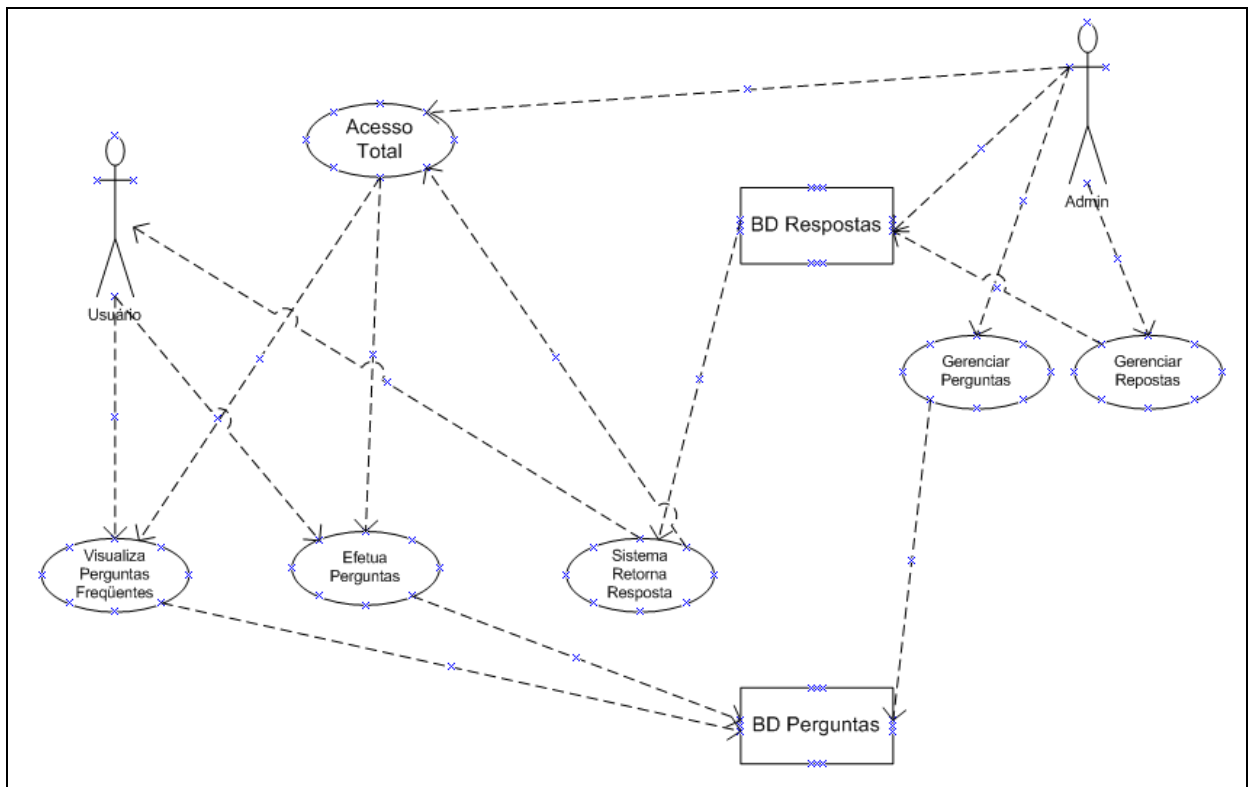


Figura 18 – Diagrama de Fluxo de Dados (Fonte: Próprio Autor)

4.5 Telas do Projeto

4.5.1 Página Principal

Nessa parte do projeto, será apresentada a tela principal da página do sistema especialista de apoio à decisão com as questões divididas em abas, cada uma contendo as questões para o auxílio à compra de automóveis das principais montadoras comercializadas no Brasil (Audi, Chevrolet, Fiat, Ford, Honda, Hyundai, Kia, Nissan, Peugeot, Renault, Toyota e Volkswagen).

O sistema permite cadastrar perguntas, respostas, modelos de carros bem como suas características e marcas o quanto for necessário.

As questões, respostas e características dos veículos são cadastradas através da área administrativa onde são armazenadas no banco de dados *MySQL*.

De acordo com as respostas do usuário, o sistema de apoio à decisão fará a busca, trazendo o resultado ao usuário, informando qual veículo comprar.

Ao acessar a página o usuário visualizará a tela conforme mostra a figura 19, na qual possui as abas com as questões que auxiliaram na tomada de decisão para a compra do automóvel.

O usuário poderá clicar diretamente na pergunta a ser respondida, ou em apenas uma delas, escolhendo uma ou mais respostas de cada. O usuário também pode iniciar os questionamentos clicando no botão “iniciar”.

As questões poderão ser alteradas a qualquer momento, somente se o mesmo ainda não tenha clicado nos botões “Encontrar o veículo mais adequado” ou “Efetuar pesquisa”, as repostas são divididas e de múltiplas escolhas na forma de *checkbox*.

A página possui um cabeçalho, coluna de tutorial, e o corpo que traz as perguntas e repostas e logo a baixo o rodapé com links para consulta de carros e da tabela fipe.

O corpo da página é composto por um arquivo que varre o banco de dados buscando as perguntas e repostas cadastradas na área administrativa.



Figura 19 – Página Principal do Projeto (Fonte: Próprio Autor)

A primeira questão que o usuário terá que responder, será referente ao estilo do veículo, tendo opções de livre escolha por um ou mais estilos ou todos ao mesmo tempo:

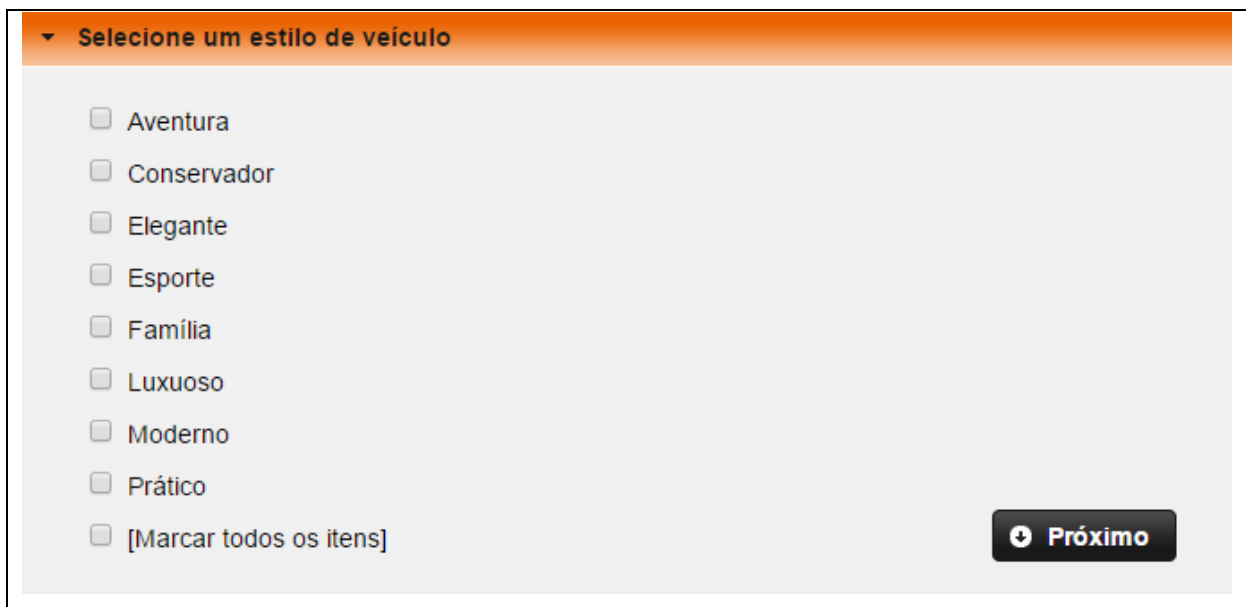
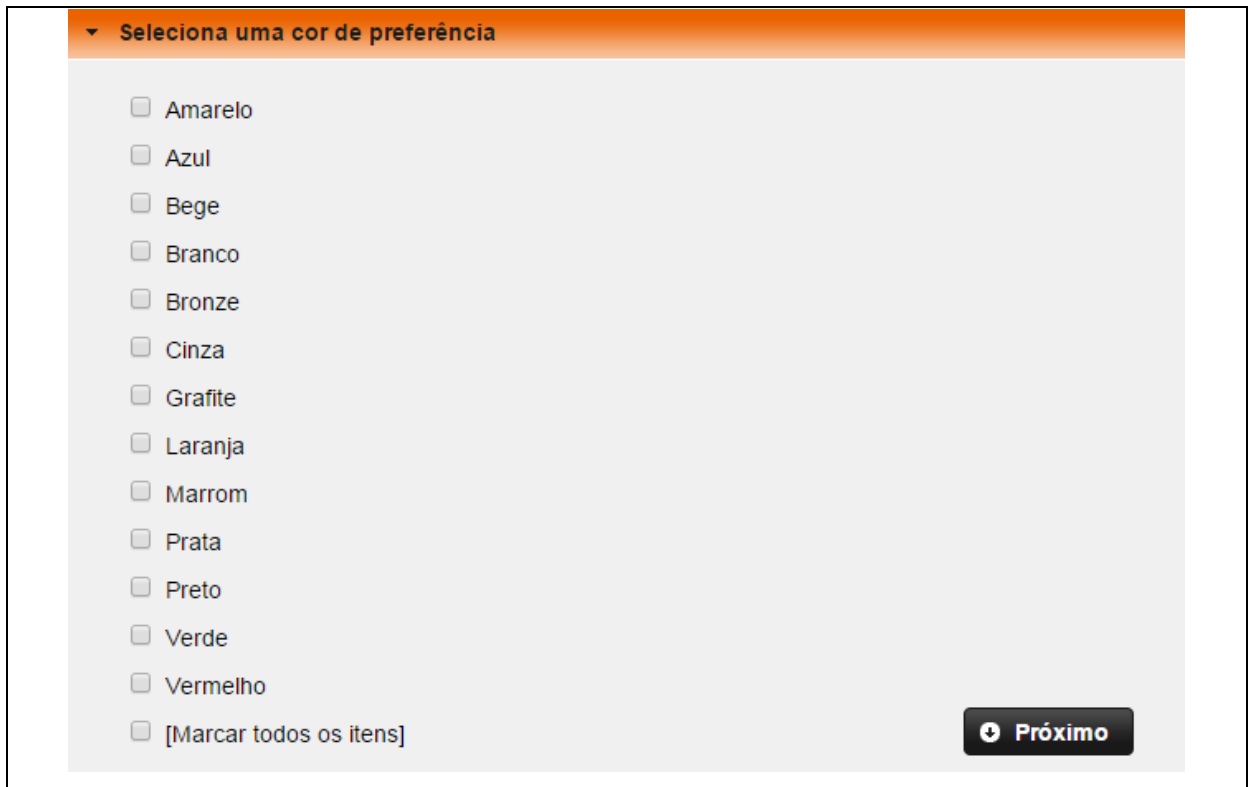


Figura 20 – Questões do sistema, opções de estilos (Fonte: Próprio Autor)

A segunda questão, conforme figura 21 que o usuário terá que responder, será referente à cor do veículo, tendo opções de livre escolha por uma ou mais cores ou todas ao mesmo tempo.



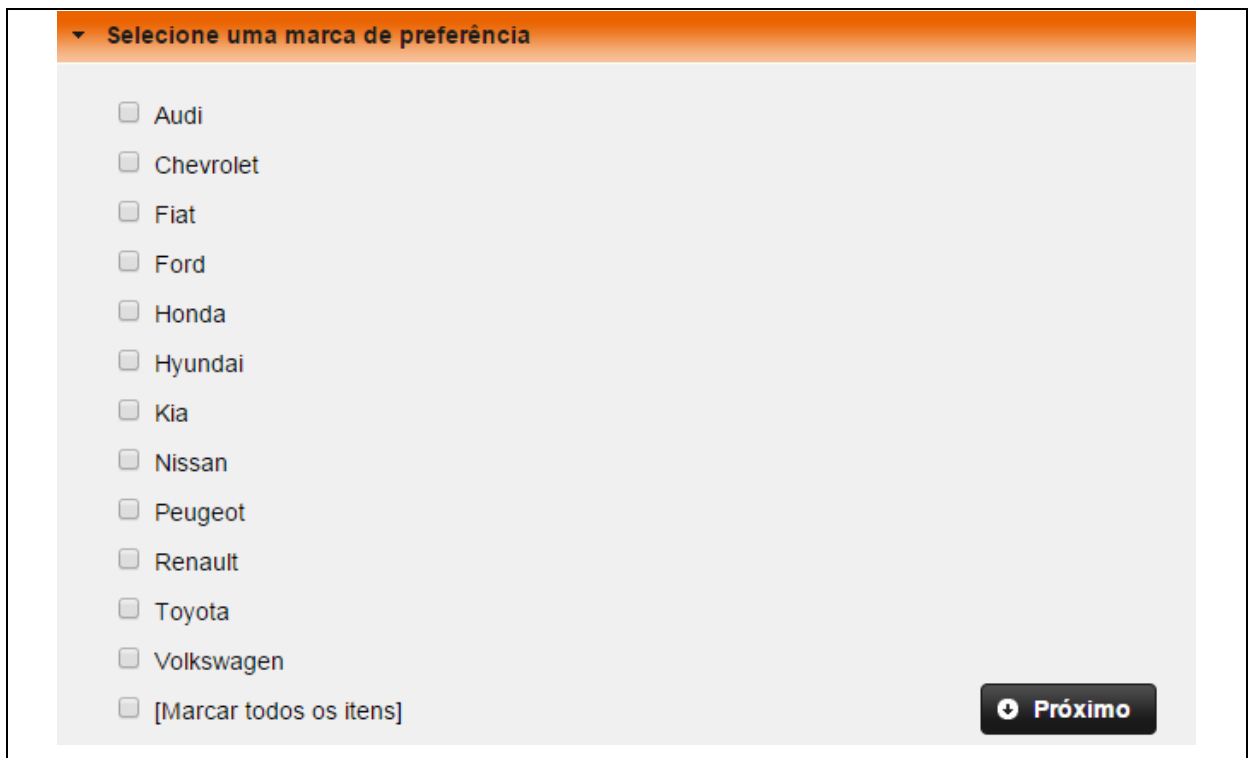
▼ **Selecione uma cor de preferência**

- Amarelo
- Azul
- Bege
- Branco
- Bronze
- Cinza
- Grafite
- Laranja
- Marrom
- Prata
- Preto
- Verde
- Vermelho
- [Marcar todos os itens]

➔ Próximo

Figura 21 – Questões do sistema, opções de cores (Fonte: Próprio Autor)

A terceira questão que o usuário terá que responder, será referente à marca do veículo, tendo opções de livre escolha por uma ou mais marcas ou todas ao mesmo tempo:



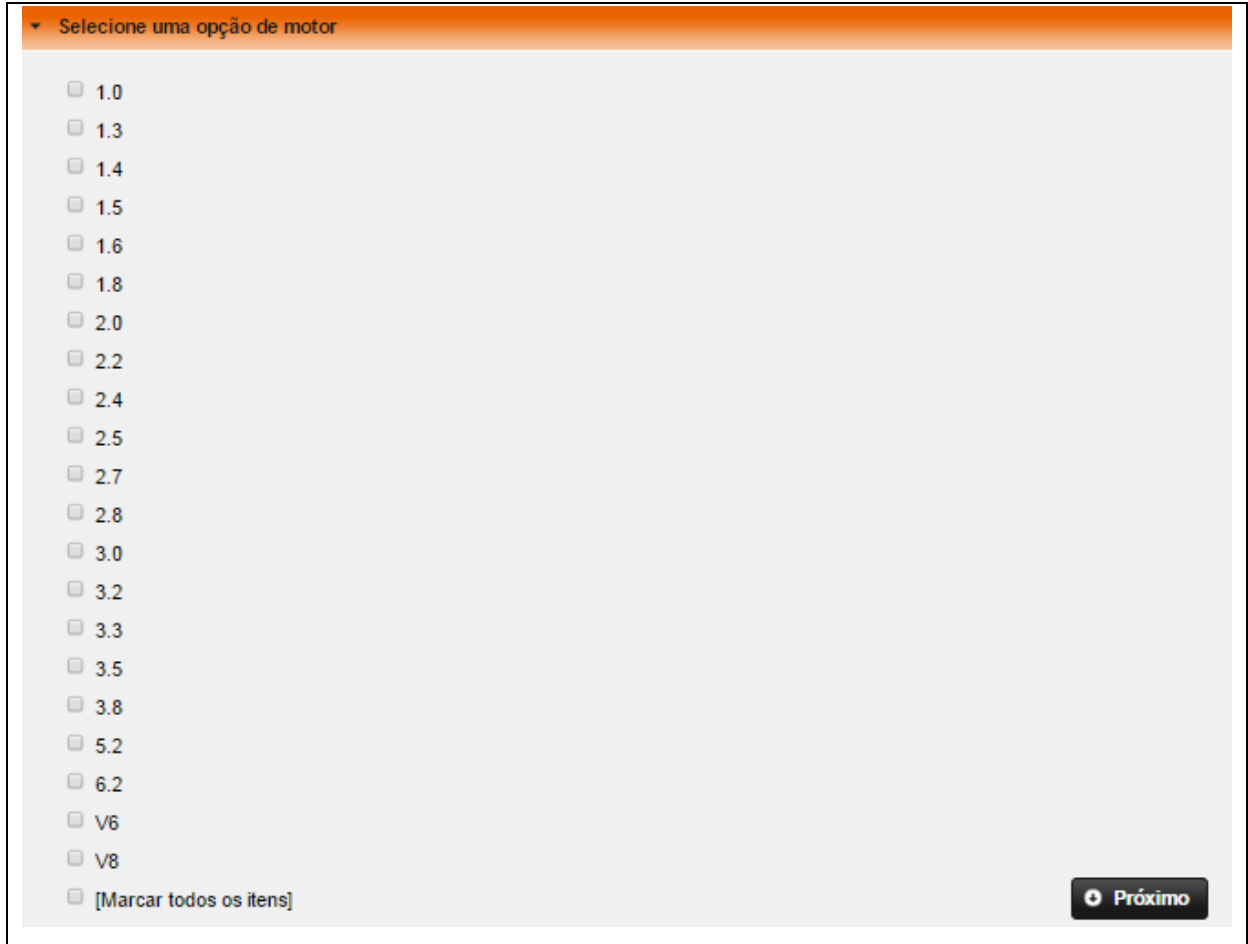
▼ **Selecione uma marca de preferência**

- Audi
- Chevrolet
- Fiat
- Ford
- Honda
- Hyundai
- Kia
- Nissan
- Peugeot
- Renault
- Toyota
- Volkswagen
- [Marcar todos os itens]

➔ Próximo

Figura 22 – Questões do sistema, opções de marcas (Fonte: Próprio Autor)

A quarta questão que o usuário terá que responder, será referente ao motor do carro, tendo opções de livre escolha por um ou mais motores ou todos ao mesmo tempo:



The screenshot shows a user interface for selecting an engine option. At the top, there is a header bar with the text "Selecione uma opção de motor" and a downward arrow. Below this, a list of engine options is displayed, each with an unchecked checkbox: 1.0, 1.3, 1.4, 1.5, 1.6, 1.8, 2.0, 2.2, 2.4, 2.5, 2.7, 2.8, 3.0, 3.2, 3.3, 3.5, 3.8, 5.2, 6.2, V6, and V8. At the bottom left of the list, there is a checkbox labeled "[Marcar todos os itens]". At the bottom right, there is a dark button with a right-pointing arrow and the text "Próximo".

Figura 23 – Questões do sistema, opções de motores (Fonte: Próprio Autor)

A quinta e última questão que o usuário terá que responder, será referente quantidade de portas do carro, tendo três opções possíveis:



The screenshot shows a user interface for selecting the number of doors. At the top, there is a header bar with the text "Selecione a quantidade de portas" and a downward arrow. Below this, three options are listed, each with an unchecked checkbox: "4 portas", "2 portas", and "3 Portas". At the bottom right, there is a dark button with a right-pointing arrow and the text "Encontrar veículo mais adequado".

Figura 24 – Questões do sistema, opções de portas (Fonte: Próprio Autor)



Figura 25 – Botão efetuar pesquisa (Fonte: Próprio Autor)

Ao término dos questionamentos o usuário irá clicar no botão “Encontrar o veículo mais adequado”, se o usuário responder uma única pergunta ou não finalizar todas as questões, poderá clicar no botão “Efetuar pesquisa” à qualquer momento tendo o resultado com a opção ou as opções de um ou mais modelos de veículos, conforme mostra figura 25.

Clicando na opção desejada do modelo que o sistema realizou o filtro da pesquisa, a página irá direcionar para o link do modelo selecionado.

Após o sistema mostrar o resultado o usuário poderá reiniciar o teste, clicando no botão “Fazer nova pesquisa”.

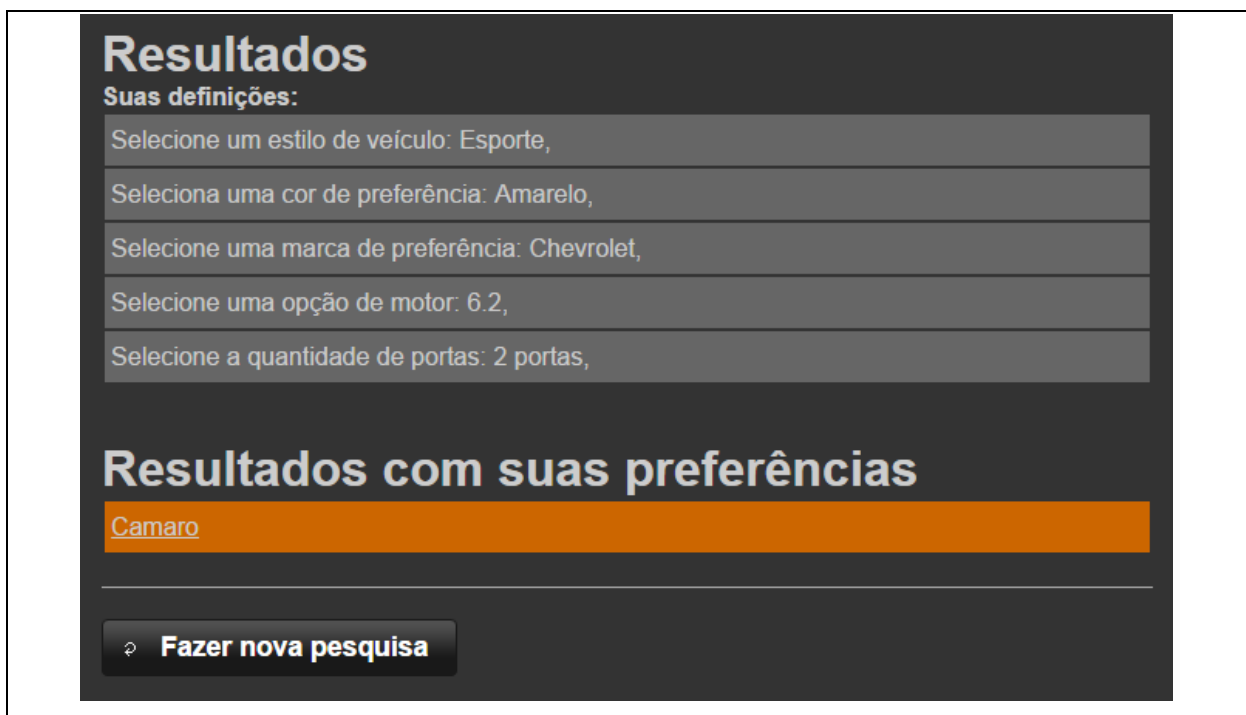


Figura 26 – Resultados (Fonte: Próprio Autor)

4.5.2 Área Administrativa

O administrador da página terá acesso através da página principal para cadastrar as perguntas, repostas e carros bem como suas características, conforme mostra figura 27.

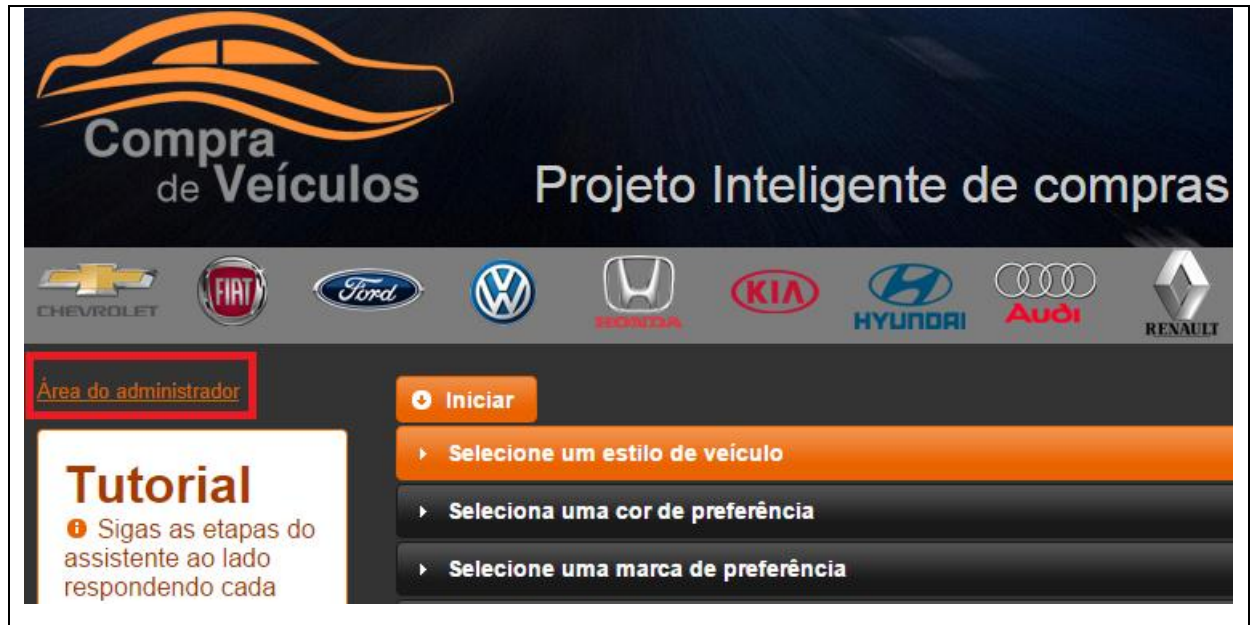


Figura 27 – Acesso da área Administrativa (Fonte: Próprio Autor)

Para acessar a área administrativa será necessário realizar autenticação de usuário e senha cadastrada no sistema pelo administrador, conforme figura 28.

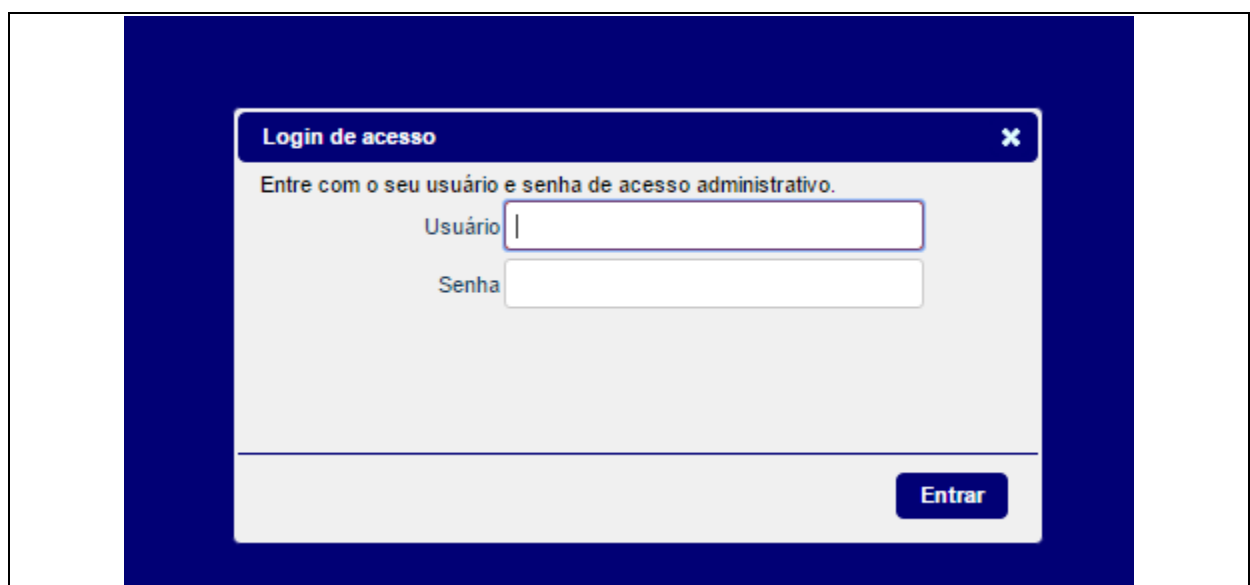


Figura 28 – Tela inicial de autenticação do administrador do sistema (Fonte: Próprio Autor)

A área de gestão de conteúdo da página ou administrativa contém um cabeçalho, uma coluna que traz o cadastro das perguntas, cadastro das respostas, a opção de registro de carros e o rodapé contendo o botão “Encerrar” para efetuar o logoff da área administrativa, conforme mostra figura 29.

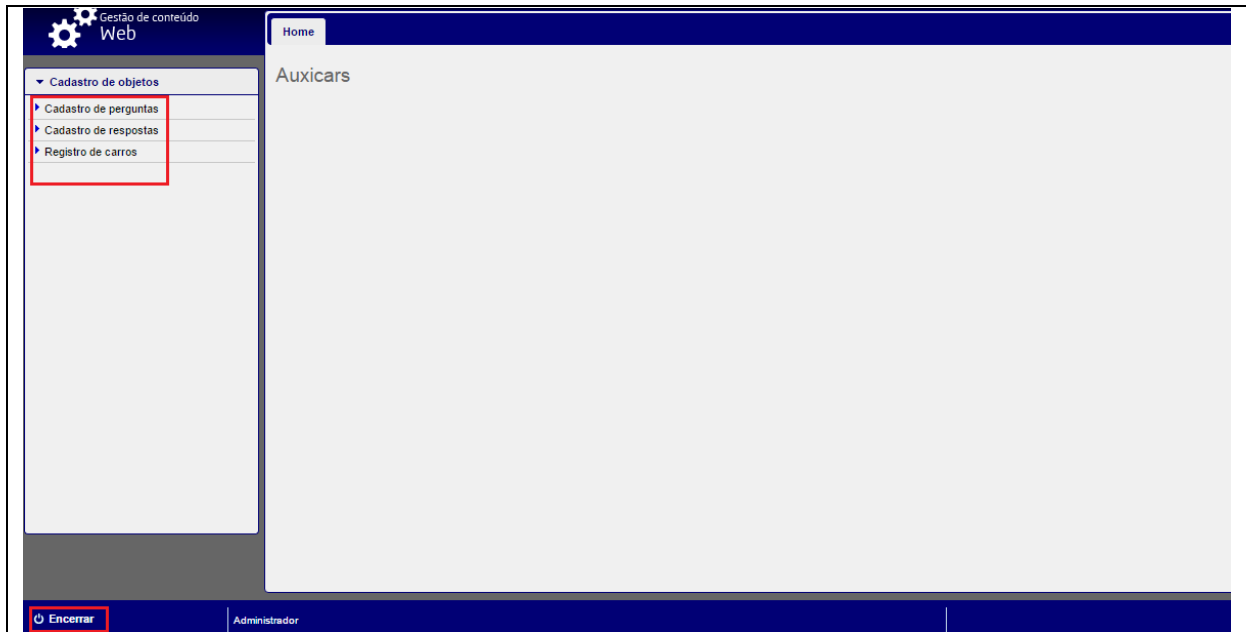


Figura 29 – Página Principal da área administrativa (Fonte: Próprio Autor)

O cadastro é realizado a partir do clique em uma das opções de acordo com a necessidade do administrador.

Ao clicar em um das opções de cadastro, será exibida uma aba no corpo da página para os devidos cadastros e alterações necessárias, após o cadastro realizado, o administrador clicará no botão “salvar” para que os dados inseridos sejam guardados no banco de dados, se a aba for fechada e o botão “salvar” não tenha sido clicado, as informações serão perdidas. O botão “cancelar” também interrompe a inserção dos dados.

A figura abaixo mostra a aba de cadastro de perguntas:

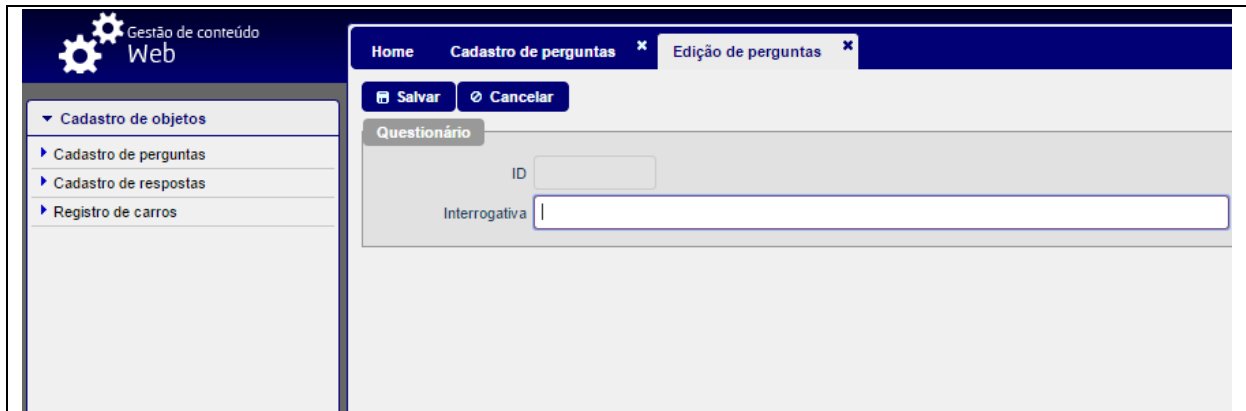


Figura 30 – Cadastro de perguntas (Fonte: Próprio Autor)

Após o cadastro de uma pergunta, a mesma aparecerá na aba de cadastros de perguntas, a opção pode ser removida através do botão “Remover” e também editada após o cadastro pelo botão “Editar”, conforme figura abaixo.



Figura 31 – Perguntas Cadastradas (Fonte: Próprio Autor).

Na tela de cadastro de respostas, o administrador irá cadastrar a resposta de acordo com a pergunta cadastrada, após o cadastro clicando no botão “Salvar” ou cancelando o cadastro clicando no botão “Cancelar”, conforme figura 32.

Figura 32 – Cadastro de respostas (Fonte: Próprio Autor).

Após o cadastro de uma resposta, a mesma aparecerá na aba de cadastros de respostas, a opção pode ser removida selecionando a opção desejada e clicando no botão “Remover” e também editada após o cadastro pelo botão “Editar”, conforme figura abaixo.

	ID	Pergunta	Resposta	Opções
<input type="checkbox"/>	1	Selecione um estilo de veículo	Aventura	
<input type="checkbox"/>	2	Selecione um estilo de veículo	Conservador	
<input type="checkbox"/>	3	Selecione um estilo de veículo	Elegante	
<input type="checkbox"/>	4	Selecione um estilo de veículo	Esporte	
<input type="checkbox"/>	5	Selecione um estilo de veículo	Família	
<input type="checkbox"/>	6	Selecione um estilo de veículo	Luxuoso	
<input type="checkbox"/>	7	Selecione um estilo de veículo	Moderno	
<input type="checkbox"/>	8	Selecione um estilo de veículo	Prático	
<input type="checkbox"/>	9	Selecione uma cor de preferência	Preto	
<input type="checkbox"/>	10	Selecione uma cor de preferência	Vermelho	
<input type="checkbox"/>	15	Selecione uma cor de preferência	Verde	

Figura 33 – Respostas Cadastradas (Fonte: Próprio Autor).

Após as perguntas e respostas serem cadastradas, os registros dos carros deve ser inseridos e vinculados às respostas nos mesmos de acordo com a característica de cada um, sendo definida pelo administrador da página. A figura 34 mostra a aba com os veículos cadastrados.

	ID	Veículo	Opções
<input checked="" type="checkbox"/>	2	A1	
<input type="checkbox"/>	3	A3	
<input type="checkbox"/>	4	A4	
<input type="checkbox"/>	5	Q3	
<input type="checkbox"/>	6	Q7	
<input type="checkbox"/>	7	TT	

Figura 34 – Registro de carros (Fonte: Próprio Autor)

Após o registro do carro concluído, o administrador tem a opção de remover o carro cadastrado se assim for necessário, ao contrário, o mesmo irá clicar no botão “Editar” e inserir as características de cada carro.

Ao clicar no botão “Editar” da coluna de opções no canto direito da página com o desenho de um “lápiz”, o administrador da página terá que inserir as características dos carros clicando no botão “Adicionar Característica”, na edição de um carro, será inserido também o link com o caminho de acordo com o site que o mesmo se encontra, onde será direcionado no momento que o usuário clicar na última questão do sistema, conforme figura 24.

Caso o administrador tenha cadastrado uma característica errada, o mesmo poderá excluir clicando no botão “deletar” da coluna de opções no canto direito da página com o desenho de uma “lixeira”, conforme mostra figura abaixo.

ID	Característica	Valor	Opções
0	Selecione um estilo de veículo	Esporte	
1	Selecione uma cor de preferência	Preto	
2	Selecione uma cor de preferência	Azul	
3	Selecione uma cor de preferência	Branco	
4	Selecione uma cor de preferência	Prata	
5	Selecione uma marca de preferência	Audi	
6	Selecione uma opção de motor	1.4	
7	Selecione uma opção de motor	1.8	
8	Selecione a quantidade de portas	4 portas	

Figura 35 – Características dos carros (Fonte: Próprio Autor)

5 CONCLUSÕES

Conclui-se através desse projeto, que um sistema especialista de apoio à decisão para compra de automóveis pela internet, é de fato uma ideia permanente no mundo multidirecionado as mais variadas marcas de automóveis. O avanço das compras pela internet não atingem patamares de grandes valias, como automóveis, porém, as pessoas estão cada vez mais procurando ferramentas que auxiliam na hora da compra de um automóvel, facilitando o acesso e principalmente a comodidade de não precisar sair de casa para sua aquisição.

As ferramentas utilizadas para o desenvolvimento foram importantes para a conclusão do trabalho, tendo um bom acervo bibliográfico nas áreas estudadas, como Inteligência Artificial e Sistemas Especialistas.

REFERÊNCIAS

- AZEVEDO, THIAGO DOS ANJOS. Seattrc: **Sistema especialista de apoio a decisão dos tipos de trabalhadores e rescisões de contratos**, Monografia do Curso de graduação em Ciência da Computação, Universidade Federal da Bahia, 2005. Disponível em <http://www.egov.ufsc.br/portal/sites/default/files/anexos/6508-6507-1-PB.pdf>. Acesso em 10 de abril de 2014.
- COPPIN, BEN. **Inteligência Artificial**, tradução e revisão técnica Jorge Duarte Pires Valério. Rio de Janeiro: LTC, 2012.
- DEITEL, PAUL J. & HARVEY M. DEITEL. **Java como programar**, São Paulo: Pearson Prentice Hall, 2010.
- EASYPHP, EasyPHP. **Aplicação PHP**, Disponível em: <http://www.easyphp.org>. Acesso em 05 de agosto de 2015.
- FARIA, MARIA ISABEL, **Novo dicionário do livro**. Circuito de leitores 1999. Disponível em: http://www.sdoc.ipleiria.pt/portal/sdoc/?p_id=96901. Acesso em 01 de maio de 2014.
- FELIPINI, DAILTON. **Lojas Virtuais Eficazes**: Artigo ecommerce. 2004. Disponível em: http://www.e-commerce.org.br/artigos/lojas_virtuais_eficazes.php. Acesso em 20 de abril de 2014.
- FELIPINI, DAILTON. **Empreendedor da Internet: Fatores Críticos de Sucesso**. 2013. Disponível em: <http://www.e-commerce.org.br/artigos/empreendedor-internet.php>. Acesso em 20 de abril de 2014.
- FERRARI, F. A. **Crie banco de dados em MySQL**. São Paulo: Digerati Books, 2007.
- HOLZNER, S. **jQuery: Visual QuickStart Guide**. [S.l.]: Peachpit Press, 2009.
- KNIGHT, RICH E. **Inteligência Artificial**. São Paulo: Editora Makron Books, 1993.
- LEE, ROSINA WEBER. **“Pesquisa Jurisprudencial Inteligente”**. Florianópolis, 1998. Tese (Doutorado em Engenharia de Produção) – Centro tecnológico, Univ. Federal de Santa Catarina.
- MARCELO, A. **Apache: Configurando o servidor Web para Linux**. 3ª. ed. Rio de Janeiro: Brasport, 2005.
- ORACLE. **Mysql Workbench 5.2.43**, 2012.
- ORACLE. NetBeans. **NetBeans**, 2012. Disponível em: <<http://netbeans.org>>. Acesso em 25 março 2014.
- RABUSKE, RENATO ANTONIO. **Inteligência Artificial**. Editora da UFSC, 2000.

RABUSKE, RENATO ANTONIO. **Inteligência Artificial**. Editora da UFSC, 1995.

RAMOS, R. F. **Sistemas especialistas**. Dissertação de mestrado, UFSC, 1995.

REZENDE, SOLANGE O. **Sistemas Inteligentes**. Barueri: Manole, 2003.

SAVARIS, SILVANA VALDEMARA APARECIDA MICHELOTTO, **Sistema Especialista para Primeiros Socorros para Cães**,. Disponível em:

<http://www.inf.ufsc.br/~barreto/teses/savaris.pdf>. Acesso em 15 de março de 2014.

SOUZA, CLEIDSON. **Conceito de orientação à objetos**. UFPA, 2007. Disponível em:

<http://www.ufpa.br/cdesouza/teaching/es/3-OO-concepts.pdf>. Acesso em 25 de março de 2014.

THE PHP GROUP. **PHP Hypertext Preprocessor**, 2012. Disponível em:

<<http://www.php.net/>>. Acesso em: 02 abril 2014.

WINSTON, P. H. **Inteligência Artificial**. Addison Wesley: Livros Técnicos e Científicos Editora Ltda, 1987.

ANEXOS

Descritiva do código de implementação desenvolvido dos principais métodos de controle do projeto incluindo conexão com o banco de dados, descritos abaixo.

Classe de conexão com o banco de dados: cms/classes/class_conexdb.php

```

1 function __construct() {
2     setlocale(LC_ALL, 'pt_BR');
3     date_default_timezone_set('America/Sao_Paulo');
4
5     global $DBServer, $DBUser, $DBPassword, $DBdatabase, $DBDrive;
6     try {
7         $DB = ADONewConnection($DBDrive);
8     } catch (Exception $e){
9         var_dump($e);
10        adodb_backtrace($e->gettrace());
11    }
12
13    $DB->debug = $this->debug;
14
15    $DB->PConnect($DBServer, $DBUser, $DBPassword, $DBdatabase);
16
17    if($DB->IsConnected()){
18        $this->DB = $DB;
19    } else {
20        die("Problema na conexão com o banco de dados");
21    }
22
23    $this->regxpagina = C_REGXPAGINA;
24    $this->COD_EMPRESA = $_SESSION["SES_COD_EMPRESA"];
25
26 }

```

Figura 36 – Código fonte da classe de conexão ao banco de dados (Fonte: Próprio Autor)

A classe “class_conexdb” possui a inclusão dos arquivos da biblioteca PHP adodb que possibilita a comunicação com diferentes bancos de dados, e esta classe possui o construtor que inicializa a conexão com o banco de dados ao ser instanciada.

Na figura 36, a linha 15 do método “PConnect” do adodb permite uma comunicação persistente com o banco MySQL.

As configurações parametrizadas no método descrito são chamados pela linha 5, onde as variáveis definidas no arquivo de configuração se tornam de uso global.

```

1 public function getResultadoPesquisa($coinc=true){
2     $rsResult = $this->ProcessaPesquisa($coinc);
3
4     $arrayResult = array();
5     if($rsResult){
6         while(!$rsResult->EOF){
7
8             $arrayResult[] = $rsResult->Fields("IDCARRO");
9             $rsResult->MoveNext();
10        }
11    }
12
13    return $arrayResult;
14 }

```

Figura 37 – Código fonte do resultado da pesquisa (Fonte: Próprio Autor)

Uma das classes mais importantes do projeto é o objeto de controle “ia.php” que possui métodos de interação, tratamento e definição de cláusulas para as pesquisas no banco de dados.

O método “getResultadoPesquisa()” da figura acima é acionado após informados os parâmetros para o objeto no resultado recebido do formulário do usuário. Estes resultados são informados no método “setRespostasQuestionarios(\$_POST);” que trata os dados provenientes do método POST do formulário e filtra em variáveis de *array* multidimensional.

Com estes parâmetros, a linha 2 da figura 37 informa a chamada do método “ProcessaPesquisa()” que retorna os valores possíveis do banco de dados.

```

1 private function ProcessaPesquisa($coincidente=true){
2     $x = " and ";
3     $coincidente==true ? $x = " and " : $x = " or ";
4     $where = " where ";
5
6     $itensP = count($this->arrayPeR);
7     $p = 0;
8     foreach($this->arrayPeR as $respostas){
9         $p++;
10        $numR = count($respostas);
11        $ir = 0;
12        $subQuery = "(";
13        foreach ($respostas as $idresposta){
14            $subQuery .= "IDRESPOSTA = ".$idresposta;
15            $ir++;
16            $ir < $numR ? $subQuery .= " or " : "";
17        }
18        $subQuery .= ")";
19        $where .= "cc.IDCARRO in (select IDCARRO from carros_caracteristicas cc
20 where ".$subQuery." ";
21        $p < $itensP ? $where .= $x : "";
22    }
23    for($i=0;$i<$itensP;$i++){
24        $where .= ")";
25    }
26
27    $SQL = "select "
28        . " car.IDCARRO"
29        . ", car.NOMCARRO"
30        . " from carros car"
31        . " inner join carros_caracteristicas cc on (car.IDCARRO =
32 cc.IDCARRO)"
33        . " ".$where." group by car.IDCARRO order by car.NOMCARRO";
34
35    return $this->SQL($SQL, __METHOD__);
36 }

```

Figura 38 – Código fonte da função pesquisa (Fonte: Próprio Autor)

Na figura acima, é apresentado o método de tratamento das variáveis informadas pelo usuário para a concatenação das cláusulas de pesquisa que o sistema efetua após o envio de seleção das respostas do internauta.

O parâmetro “\$coincidente” define se a pesquisa será idêntica ou similar às preferências do internauta definidas nas linhas 1 e 2.

Na linha 8 as perguntas respondidas são sequenciadas e respondidas uma a uma na linha 13.

As respostas com mais de uma seleção para cada pergunta são incrementadas na pesquisa utilizando o operador lógico “or” enquanto para cada resposta da linha 8 é incrementado o operador “e”.

A linha 19 concatena a cláusula “where” conforme as variáveis incrementando os filtros de *subquerys*. O resultado final é agrupado por cada tipo de carro e ordenado alfabeticamente pelo nome na linha 33.