



Fundação Educacional do Município de Assis
Instituto Municipal de Ensino Superior de Assis - IMESA

Caique Frioli Lopes

Garage Sale: Inovação em Sistemas para E-Commerce

Assis
2014

Caique Frioli Lopes

Garage Sale: Inovação em Sistemas para E-Commerce

Trabalho de Conclusão de Curso apresentado ao Instituto Municipal de Ensino Superior de Assis, como requisito do Curso de Análise e Desenvolvimento de Sistemas.

Orientador: Drº Almir Rogério Camolesi

Área de Concentração: Desenvolvimento de Sistemas

Assis
2014

FICHA CATALOGRÁFICA

Lopes, Caique

Garage Sale: Inovação em Sistemas para E-Commerce / Caique Lopes.
Fundação Educacional do Município de Assis – FEMA – Assis, 2014.
43p.

Orientador: Dr. Almir Rogério Camolesi
Trabalho de Conclusão de Curso – Instituto Municipal de Ensino Superior de Assis
– IMESA.

1. Sistemas de Informação; 2. Comércio Eletrônico; 3. Análise Orientada a Objetos;
4. Desenvolvimento em Camadas; 5. Desenvolvimento .Net

DEDICATÓRIA

Dedico este trabalho à minha família, amigos ao meu orientador e todas as pessoas que me apoiaram, em especial meu pai que sempre me apoiou.

AGRADECIMENTOS

Agradeço primeiramente a Deus, pois sem Ele, eu nada seria possível.

Aos meus amigos que sempre estão comigo nas horas boas e ruins.

Agradeço também aos familiares que sempre me deram apoio para que concluísse a faculdade.

Ao meu orientador, Drº Almir Rogério Camolesi, pela orientação, durante todo o período deste trabalho e por toda a caminhada acadêmica.

Nossa maior fraqueza está em desistir. O caminho mais certo de vencer é tentar mais uma vez.

Thomas Edison.

RESUMO

Atualmente, a tecnologia da informação vem crescendo cada vez mais, no Brasil e no mundo, um dos fatores que tem contribuído para esse crescimento, é uso de softwares específicos, com esse crescimento veio também a comodidade, coisas como efetuar uma compra acabam se tornando mais fáceis, devido a procura constante por compra e venda, acabam surgindo *sites* que efetuem essa ponte entre o comprador e fornecedor.

Neste trabalho serão apresentadas a especificação e a realização de um aplicativo Web, destinado a usuários que busquem diversos produtos com preço acessível.

Na página inicial do *site* tem o *menu* de navegação, podendo direcionar o usuário a página desejada, ainda na página inicial temos uma área para cadastro de usuários e *login*, caso o usuário possua cadastro, nesta página os usuários podem visualizar todas as garagens cadastradas, com diversos tipos de produtos, porém para comprar um produto devem estar *logado* no *site*, o usuário poderá também buscar uma garagem pelo nome através do campo “Buscar”.

Para cadastrar-se, o usuário deverá acessar a área “cadastre-se”, e preencher os campos obrigatórios corretamente, lembrando que o mesmo deverá aceitar os termos de uso. Assim que cadastrado o usuário poderá efetuar compras, ele também terá acesso à “área do usuário”, nesta contem todos os seus dados informados no seu cadastro, caso necessário uma atualização de dados, o mesmo poderá fazê-la por esta área.

Para que se torne um vendedor, o usuário terá de “Inaugurar sua garagem”. Através do botão “garagem” ou da área “Inaugure sua garagem”, o mesmo terá acesso à área de cadastro de garagens, tendo que nomeá-la e selecionar uma foto. Assim a garagem criada já estará na página inicial e pronta para colocar os produtos à venda.

Na tela inicial, temos todas as garagens criadas por usuários vendedores, nela podemos encontrar os produtos, e adiciona-los ao carrinho. No carrinho podemos finalizar a compra, continuar comprando ou até mesmo excluir o produto caso desejado.

Para o desenvolvimento deste trabalho foi efetuado um estudo sobre tecnologias a serem utilizadas para a realização do sistema, como linguagem de programação ASP.Net, com o banco de dados Microsoft SQL Server 2012.

ABSTRACT

Currently, information technology is growing increasingly in Brazil and the world, a factor that has contributed to this growth is the use of specific software, this growth also came to comfort, things like making a purchase eventually becoming more easy, due to the constant demand for buying and selling websites end up popping them to arrange this bridge between buyer and supplier.

In this paper the specification and the implementation of a Web application, intended for users who seek different products with affordable price will be displayed.

On the homepage of the site is the navigation menu, and can direct the user to the desired page, the home page still have an area for user registration and login, if the user has joined, this page users can view all registered garages, with various types of products, but to buy a product should be logged on the site, the user can also get a garage by name through the "Search" field.

To register, the user must access the area "cadaster-se" and fill in the required fields correctly, remembering that it must accept the terms of use. Once registered the user can make purchases, he will also have access to the "user area", this contains all of your knowledgeable in your registration data if necessary an update of data, it can do so is by area.

To become a vendor, the user must "Inaugurar sua garagem." Through the "garagem" button or the "inaugurate his garage" area, the same area will have access to records of garages, having to name it and select a photo. So the garage is already created in the early and ready to get products for sale page.

In the initial screen, we all garages created by user's vendors, it can find the products and add them to the cart. In the cart can finalize the purchase, continue shopping or even delete the product if desired.

To develop this work a study of technologies to be used for the realization of the system, such as language ASP.Net programming with Microsoft SQL Server 200 data was performed.

LISTA DE ILUSTRAÇÕES

Figura 1. Funcionamento do padrão de projetos MVC.**Erro! Indicador não definido.**

Figura 2. Mapa Mental**Erro! Indicador não definido.**

Figura 3. Caso de uso geral comum entre os atores.8

Figura 4. Caso de uso geral do Sistema9

Figura 5. Caso de uso geral do Administrador.**Erro! Indicador não definido.**

Figura 6. Caso de uso geral do Usuário.....**Erro! Indicador não definido.**

Figura 7. Controle Acesso.....**Erro! Indicador não definido.**0

Figura 8. Diagrama de Atividades Cadastro.....**Erro! Indicador não definido.**1

Figura 9. Diagrama de Atividades Compra.**Erro! Indicador não definido.**1

Figura 10. Diagrama de Classes – *Model*.**Erro! Indicador não definido.**2

Figura 11. Modelo Entidade-Relacionamento.**Erro! Indicador não definido.**3

Figura 12. Work Breakdown Structure**Erro! Indicador não definido.**4

Figura 13. Sequenciamento de Atividades.....**Erro! Indicador não definido.**5

Figura 14. Camadas do projeto**Erro! Indicador não definido.**7

Figura 15. CSS3 e HTML5**Erro! Indicador não definido.**8

Figura 16. Java Script e JQuery.....**Erro! Indicador não definido.**9

Figura 17. Código pagina inicial.**Erro! Indicador não definido.**0

Figura 18. Código cadastro de fotos.**Erro! Indicador não definido.**0

Figura 19. Pasta das fotos cadastradas.**Erro! Indicador não definido.**1

Figura 20. Camada BLL.**Erro! Indicador não definido.**1

Figura 21. Camada DAL.....**Erro! Indicador não definido.**2

Figura 22. Código camada DAL.**Erro! Indicador não definido.**2

Figura 23. Camada MODEL.....**Erro! Indicador não definido.**3

Figura 24. Código camada MODEL.	Erro! Indicador não definido.	3
Figura 25. Pagina inicial do <i>Site</i>	Erro! Indicador não definido.	4
Figura 26. Área de <i>login</i>	Erro! Indicador não definido.	4
Figura 27. Tela de cadastro de usuário.	Erro! Indicador não definido.	5
Figura 28. Campo de pesquisa.	Erro! Indicador não definido.	5
Figura 29. Carrinho de compras.	Erro! Indicador não definido.	5
Figura 30. Banner do <i>site</i>	Erro! Indicador não definido.	6
Figura 31. Menu do <i>site</i>	Erro! Indicador não definido.	6

LISTA DE TABELAS

Tabela 1 Efetuar Controle Acesso.	10
--	----

Sumário

1	INTRODUÇÃO	1
1.1	OBJETIVO	1
1.2	JUSTIFICATIVA	2
1.3	PUBLICO-ALVO.....	2
1.4	ESTRUTURA DO TRABALHO.....	2
2	TECNOLOGIAS E FERRAMENTAS DE DESENVOLVIMENTO	3
2.1	Tecnologia de Análise	3
2.1.1	UML	3
2.2	Tecnologias de Desenvolvimento	4
2.2.1	CSS3	4
2.2.2	jQuery.....	4
2.2.3	.Net.....	4
2.2.4	Visual Studio.....	4
2.2.5	HTML5.....	5
2.3	MICROSOFT SQL SERVER 2008 R2	5
2.4	PADRÃO DE PROJETO MVC	5
2.5	CRYSTAL REPORTS	7
3	ANÁLISE E ESPECIFICAÇÃO DO SISTEMA.....	7
3.1	MAPA MENTAL	7
3.2	LISTA DE EVENTOS	8
3.3	CASOS DE USO	8
3.4	DIAGRAMA DE ATIVIDADES.....	12
3.5	DIAGRAMA DE CLASSES.....	13
3.6	MODELAGEM DE ENTIDADE E RELACIONAMENTO.....	14

4	ESTRUTURA DO PROJETO	16
4.1	ESTRUTURA ANALÍTICA DE TRABALHO.....	16
4.2	SEQUENCIAMENTO DE ATIVIDADES.....	16
4.3	Orçamento	17
4.4	Cronograma.	Erro! Indicador não definido.
5	Implementação do <i>site</i>	18
5.1	Camadas de Interface	18
5.1.1	Lista de Garagens	20
5.1.2	Cadastrar Foto.....	21
5.2	Organização do Projeto BLL	22
5.3	Organização do Projeto DAL.....	23
5.4	Organização do Projeto Model.....	24
5.5	Interfaces do Sistema.....	24
5.5.1	Interface Principal	24
5.5.2	Interface <i>Login</i>	25
5.5.3	Campo de Busca	26
5.5.4	Área de compra (carrinho).....	26
5.5.5	Banner Interativo	27
5.5.6	Menu.....	27
6	Conclusão	28
6.1	Trabalhos Futuros	28
	REFERENCIAS BIBLIOGRAFICAS	29
	REFERENCIAS DIGITAL	29

1 INTRODUÇÃO

De acordo com os dados divulgados pelo jornal Folha de São Paulo publicado no dia 20 de Setembro de 2013 “O comércio eletrônico mantém um ritmo de forte expansão no Brasil...”.

Segundo o *site* globo.com¹ “As vendas no comércio eletrônico brasileiro neste ano devem atingir R\$ 28 bilhões, crescimento de cerca de 24% sobre os R\$ 22,5 bilhões faturados em 2012”.

Baseado nestes dados é proposto este trabalho que tem por objetivo o desenvolvimento do portal o E-Commerce “Garage Sale” um *site* no qual o usuário poderá criar sua garagem e expor seus produtos a venda tornando-o assim um E-Commerce interativo.

Existem outros E-commerce relacionados a venda de produtos e serviços como Bomnegocio², Mercado livre³, porém estes *sites* se limitam somente a anúncios e vendas de produtos, já o “Garage Sale ” será um *site* interativo onde você poderá montar sua garagem com seus produtos.

Tais recursos possibilitarão que o usuário selecione um plano de fundo a sua garagem, deixe a garagem com promoção, moldando a assim de acordo com gosto do usuário.

Os recursos disponibilizados pelo *site* proporcionarão ao usuário facilidades que contribuem para o sucesso na venda.

1.1 OBJETIVO

O objetivo principal deste trabalho é aprender sobre as novas tecnologias emergentes no mercado, propondo assim um *site* utilizando as mesmas. Um *site*, que permita aos usuários criar espaços para expor os seus produtos.

Os usuários do *site* terão uma cota livre, para divulgação de seus produtos podendo adquirir diante de remuneração mais cotas ou mediante os contatos que ele

¹ <http://www.globo.com/>

² <http://www.bomnegocio.com/>

³ <http://www.mercadolivre.com.br/>

disponibilizar. A ideia central é privilegiar o usuário que tiver mais acesso e disponibilizar mais contatos e assim possibilitar ao *site* mais visitas.

1.2 JUSTIFICATIVA

Será feito um *site* para que possa explorar sobre as novas tecnologias no mercado.

Diferente de vários *sites* de hoje que oferecem poucos recursos para que o usuário possa interagir e fazer mais que um simples anúncio.

O “Garage Sale” é um *site* que se expande pelo número de acessos, ou seja, com sua utilização de forma progressiva, facilitando o acesso às informações e trazendo grandes vantagens para seus usuários.

As empresas sempre buscam por *sites* seguros, modernos e de confiança para auxiliarem seus negócios, elevando a qualidade dos serviços e reduzindo custos e aumentando suas vendas portanto o *site* “Garage Sale” será em C# web (**Barkol; Vavilala; 2009**) uma linguagem de fácil modelagem e com uma aparência agradável ao usuário.

1.3 PUBLICO-ALVO

O *site* é voltado a pessoas interessadas em trocar ou vender objetos.

1.4 ESTRUTURA DO TRABALHO

Este trabalho está dividido em capítulos que serão apresentados a seguir.

No primeiro capítulo foi apresentada a justificativa para o desenvolvimento da proposta de trabalho.

Em seguida, no segundo capítulo serão abordados os conceitos de fundamentação teórica das tecnologias utilizadas para o desenvolvimento do *software*.

O terceiro capítulo apresenta as etapas e especificações do *software* e os principais diagramas UML (classe, sequência e atividade).

Já no quarto capítulo descreverei a WBS – *Work Breakdown Structure*, o sequenciamento das atividades e o orçamento do *software*.

No, o quinto capítulo exhibe os processo de implementação do *site*. Como foi feito e os métodos utilizados.

Por fim, no sexto capítulo temos a conclusão sobre o trabalho.

.

2 TECNOLOGIAS E FERRAMENTAS DE DESENVOLVIMENTO

Neste capítulo serão descritas as tecnologias e ferramentas utilizadas para o desenvolvimento do Garage Sale.

O Garage Sale será um sistema desenvolvido com a tecnologia Microsoft .NET, que possui uma linguagem de alto nível orientada a objetos. O ambiente utilizado para o desenvolvimento do sistema será o Visual Studio 2012.

Quanto a base de dados, será utilizado o banco de dados SQL Server 2012 R2⁴.

Para uma interface dinâmica e agradável será utilizado o HTML 5 que é a quinta versão da linguagem HTML junto com CSS3 que é uma linguagem para estruturação e apresentação de conteúdo para a *World Wide Web*.

Abaixo serão descritas as principais tecnologias utilizadas neste trabalho e suas características:

2.1 Tecnologia de Análise

2.1.1 UML

A UML (*Unified Modeling Language*) é uma linguagem de modelagem não proprietária de terceira geração. A UML não é uma metodologia de desenvolvimento, o que significa que ela não diz para você o que fazer primeiro e em seguida ou como projetar seu sistema, mas ela lhe auxilia a visualizar seu desenho e a comunicação entre objetos.

Basicamente, a UML permite que desenvolvedores visualizem os produtos de seus trabalhos em diagramas padronizados. Junto com uma notação gráfica, a UML também especifica significados, isto é, semântica. É uma notação independente de processos, embora o RUP (*Rational Unified Process*) tenha sido especificamente desenvolvido utilizando a UML.⁵

⁴ <https://www.microsoft.com/pt-br/download/details.aspx?id=1695>

⁵ <http://www.targettrust.com.br/blog/cursos/curso-de-uml/o-que-e-uml/>

2.2 Tecnologias de Desenvolvimento

2.2.1 CSS3

CSS3 é a mais nova versão das famosas CSS (*Cascading Style Sheets*), onde se define estilos para páginas web com efeitos de transição, imagem e outros.

A principal função do CSS3 é abolir as imagens de plano de fundo, bordas arredondadas, apresentar transições e efeitos para criar animações de vários tipos, como um simples relógio de ponteiros.

2.2.2 jQuery

jQuery foi desenvolvida para tornar mais simples a navegação do documento HTML.

jQuery é uma biblioteca *JavaScript* rápida, pequena e rica em recursos. Isso torna as coisas como travessia documento HTML e manipulação, manipulação de eventos, animação e Ajax muito mais simples com uma API fácil de usar, que funciona através de uma infinidade de navegadores. Com uma combinação de versatilidade e capacidade de expansão, *jQuery*⁶ mudou a maneira que milhões de pessoas escrever *JavaScript*.⁷

2.2.3 .Net

O .NET⁸ é um ambiente de tempo de execução gerenciado, que proporciona uma variedade de serviços para os diversos aplicativos em execução. Ele consiste em dois principais componentes, que são: o *Common Language Runtime* (CLR), que é o mecanismo de execução e manipula os aplicativos que estão em execução e a biblioteca de classes do .NET, que fornece uma biblioteca de código testado e, reutilizável que os desenvolvedores podem chamar a partir dos seus próprios aplicativos.

2.2.4 Visual Studio

O Visual Studio⁹ é um conjunto abrangente de ferramentas e serviços que ajuda você a criar uma ampla variedade de aplicativos, para a plataforma Microsoft e além. O Visual Studio também se conecta a todos os seus projetos, equipe e participantes. Agora sua equipe pode trabalhar com mais agilidade de praticamente qualquer

⁶ <http://jquery.com/>

⁷ <http://portaldesenvolvedor.blogspot.com.br/2014/03/jquery.html>

⁸ <http://www.asp.net/>

⁹ <http://www.visualstudio.com/>

lugar, independentemente da ferramenta de desenvolvimento, incluindo o Eclipse e o Xcode. Se estiver desenvolvendo aplicativos .NET essenciais à missão, escrevendo código rápido conhecidos com o C++ AMP ou testando e depurando um aplicativo HTML/JavaScript conectado por nuvem que é executado em muitos dispositivos, junte-se a milhares de desenvolvedores do mundo todo na escolha do Visual Studio como seu principal ambiente de desenvolvimento.¹⁰

2.2.5 HTML5

O HTML é considerado uma linguagem sólida e de confiança pelos programadores e desenvolvedores, e por isso mesmo tornou-se o padrão de uso em todos os navegadores atuais. Com o advento do HTML5 em 2007, certas melhorias foram introduzidas, mas foi apenas em 2010 que esses novos recursos começaram a ser utilizados com mais frequência.¹¹

2.3 MICROSOFT SQL SERVER 2008 R2

O Microsoft SQL Server é um conjunto completo de tecnologias e ferramentas que ajudam a gerar o máximo de informações possíveis a curto espaço de tempo, com altos níveis de desempenho, disponibilidade e segurança de dados, sendo possível empregar ferramentas de gerenciamento e desenvolvimento de aplicações **(MICROSOFT, 2012)**.

Esse Sistema Gerenciador de Banco de Dados Relacional (SGBD), foi criado pela Microsoft em parceria com a Sybase em 1988 e inserido como produto complementar do Windows NT. Em 1994, a parceria foi rompida e a Microsoft continuou aperfeiçoando o produto.

É uma plataforma de dados confiável, produtiva e inteligente que permite a execução de aplicações de missão crítica mais exigentes, reduzindo o tempo e o custo com o desenvolvimento e o gerenciamento de aplicações, além de ser um banco de dados robusto e usado por sistemas corporativos dos mais diversos portes **(PORTAL EDUCAÇÃO, 2008)**.¹²

2.4 PADRÃO DE PROJETO MVC

Um padrão de projeto tem como função nomear, abstrair e identificar os aspectos-chaves de uma estrutura de projeto comum, tornando-a útil para a criação de um projeto orientado a objetos de maneira reutilizável **(TAMIRES, 2011)**.

¹⁰ [http://msdn.microsoft.com/pt-br/library/hh425099\(v=vs.110\).aspx](http://msdn.microsoft.com/pt-br/library/hh425099(v=vs.110).aspx),

¹¹ <http://msdn.microsoft.com/pt-br/library/dn151487.aspx>

¹² <http://www.portaleducacao.com.br/microsoft-oficial/cursos/2304/curso-de-10142ac-instalacao-e-manutencao-do-microsoft-sql-server-2008>

O padrão de projeto Model View Controller (MVC) tem como objetivo separar a lógica de negócios (Model) da interface de usuário (View) e do fluxo da aplicação (Controller), permitindo o desenvolvimento, teste e manutenção isolada de ambos. Permite a reutilização dos códigos implementados nas camadas.

Abaixo podemos conferir quais são elas e para que servem (EDUARDO,2011):

Model ou Modelo: Responsável por grande parte do código escrito.

Nesta camada está presente a lógica de negócios, com o objetivo de definir e gerenciar toda a informação, bem como a notificação sobre possíveis mudanças ocorridas nos dados.

View ou Visão: Compreende a interface de usuário e é responsável pelo acesso aos dados contidos em Model e especifica como estes dados colhidos serão apresentados ao usuário.

Controller ou Controle: Responsável pelo fluxo da aplicação assume o mapeamento das ações efetuadas pelo usuário na camada View, por meio de eventos e, com isso, permite que a camada Model seja alterada.

A figura abaixo ilustra o funcionamento do padrão de projeto MVC:

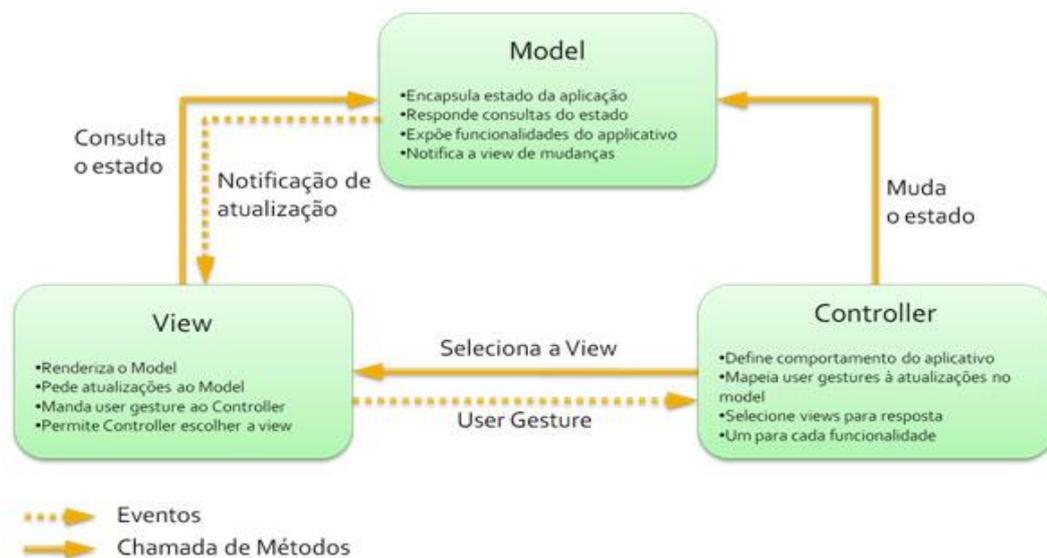


Figura 1. Funcionamento do padrão de projetos MVC.¹³

¹³ <http://www.blogomoura.com/2011/07/entendendo-o-padrao-de-projeto-mvc/>

2.5 CRYSTAL REPORTS

A ferramenta Crystal Reports permite a construção de relatórios, desde o mais simples ao mais complexo para aplicações .NET ou diretamente em aplicações web.

Pertence à SAP Business Objects e é um dos mais utilizados no mundo.

Permite a inserção de fórmulas nos relatórios e também o recebimento de dados de um sistema, através de uma conexão direta a um banco de dados relacional (Matheus, 2012).

3 ANÁLISE E ESPECIFICAÇÃO DO SISTEMA

Neste capítulo será apresentado a especificação do sistema proposto.

3.1 MAPA MENTAL

Para facilitar o entendimento e a visualização do sistema é apresentado o mapa mental que é um método utilizado para armazenar, organizar e priorizar informações através de diagramas elaborados a partir de uma ideia principal.

Sistematizado pelo inglês Tony Buzan, voltado a gestão de informações, de conhecimento e de capital intelectual (TRÍBOLI, 2004).

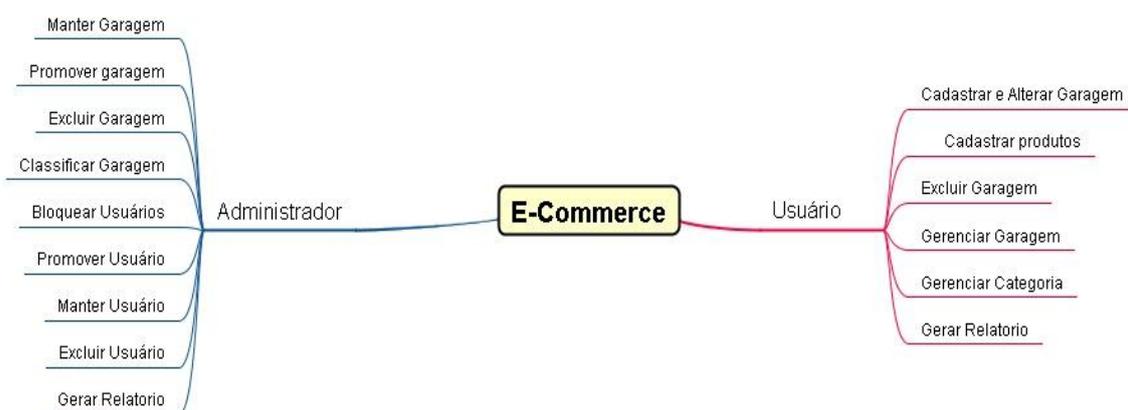


Figura 2. Mapa Mental

3.2 LISTA DE EVENTOS

Para ser realizada a visualização, especificação, construção e documentação dos artefatos de um sistema, utiliza-se a UML, pois é uma linguagem padrão para a elaboração de estrutura de projetos de *software*.

Para modelar o comportamento dos sistemas baseados em objetos, determinam-se quais eventos acontecem. Eventos fazem com que os sistemas tomem várias ações **(BOOCH; JACOBSON; RUMBAUGH, 2000)**.

Segue os principais eventos do sistema:

Cadastrar e Alterar Garagem

Cadastrar produtos

Excluir Garagem

Gerenciar Garagem

Gerenciar Categoria

Manter Garagem

Promover garagem

Excluir Garagem

Classificar Garagem

Bloquear Usuário

Promover Usuário

Gerar Relatório

Manter Usuário

Excluir Usuário

Efetua Venda

Efetua Compra

3.3 CASOS DE USO

O primeiro elemento da UML utilizado é o caso de uso. Caso de uso é um conjunto de cenários amarrados por um objetivo comum de um usuário, especifica o comportamento de um sistema ou de parte de um sistema e é uma descrição de um conjunto de sequencias de ações incluindo variantes realizadas pelo sistema para

produzir um resultado observável do valor de um ator (**BOOCH; JACOBSON; RUMBAUGH, 2005**).

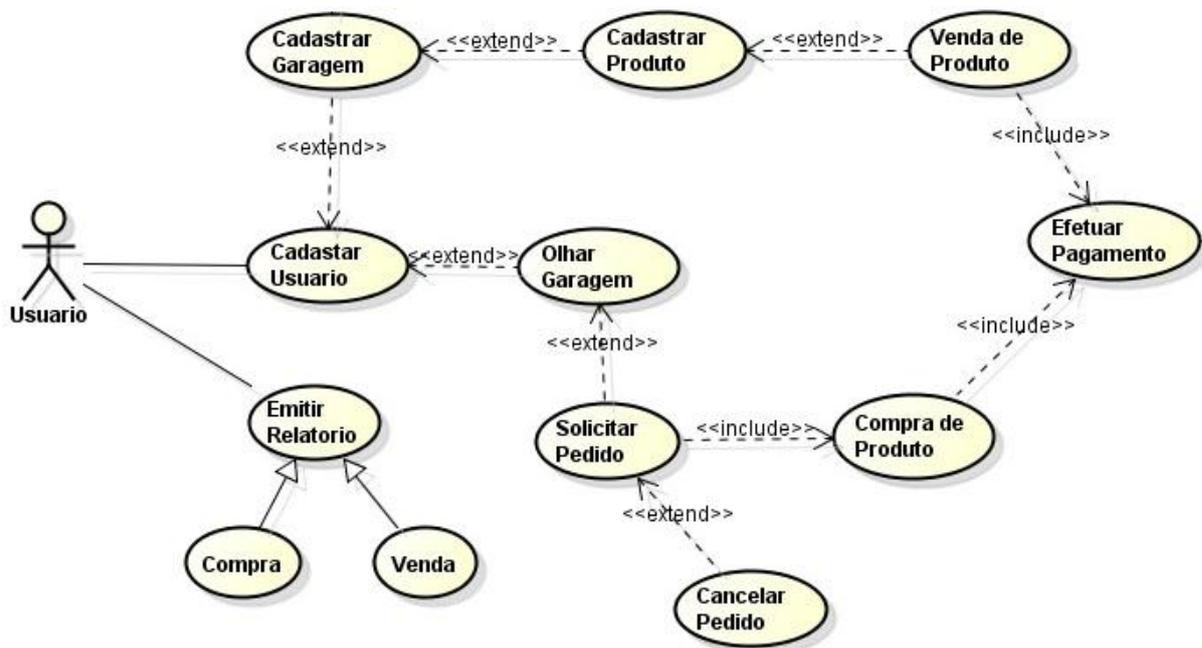


Figura 3. Caso de uso geral do Sistema.

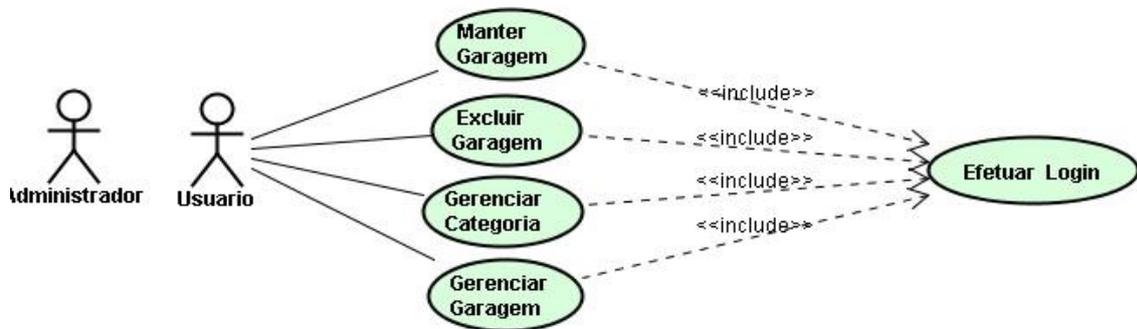


Figura 4. Caso de uso geral comum entre os atores.

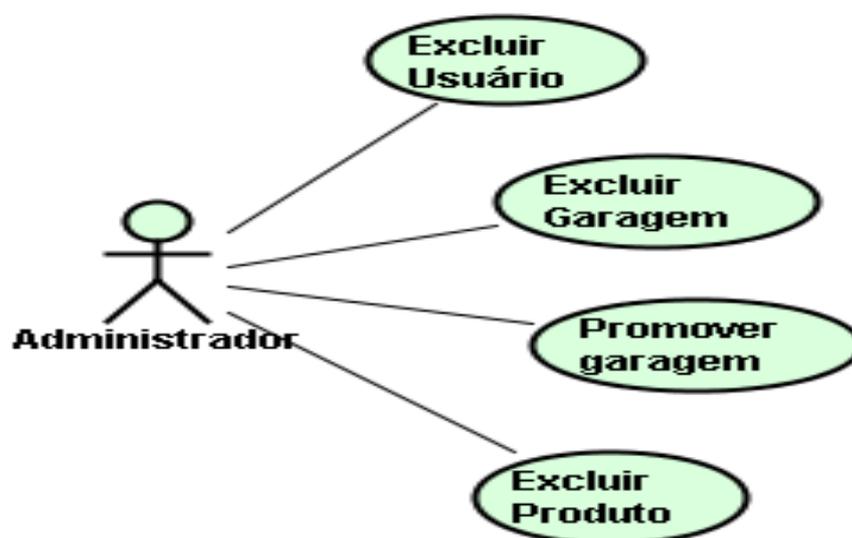


Figura 5. Caso de uso geral do Administrador.

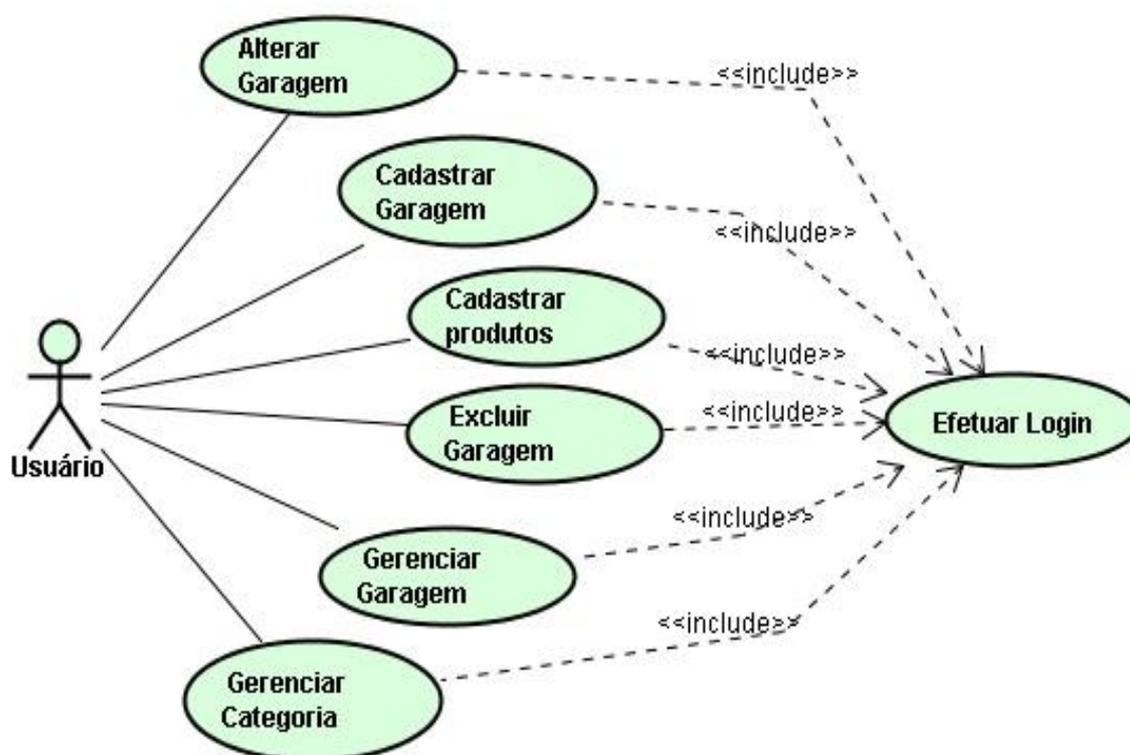


Figura 6. Caso de uso geral do Usuário.



Figura 7. Controle Acesso.

Nome do Caso de Uso	Controle de Acesso
Atores	Usuário
Pré- Condições	Efetuar <i>login</i>
Cenário Principal	<ol style="list-style-type: none"> 1- O Administrador terá uma garagem. 2- Ele deverá acessar sua conta. 3- Após o acesso deverá ir até um botão. 4- O botão o direcionará a outra pagina. 5- Nesta pagina o Administrador terá um gráfico com quantos usuários visitarão sua garagem.
Cenário Alternativo	Não existe.
Casos Teste	Não existe.

Tabela 1. Controle de Acesso.

3.4 DIAGRAMA DE ATIVIDADES

O diagrama de atividades representa os fluxos conduzidos por processamentos. É essencialmente um gráfico de fluxo, mostrando o fluxo de controle de uma atividade para outra (BOOCH; JACOBSON; RUMBAUGH,2000).

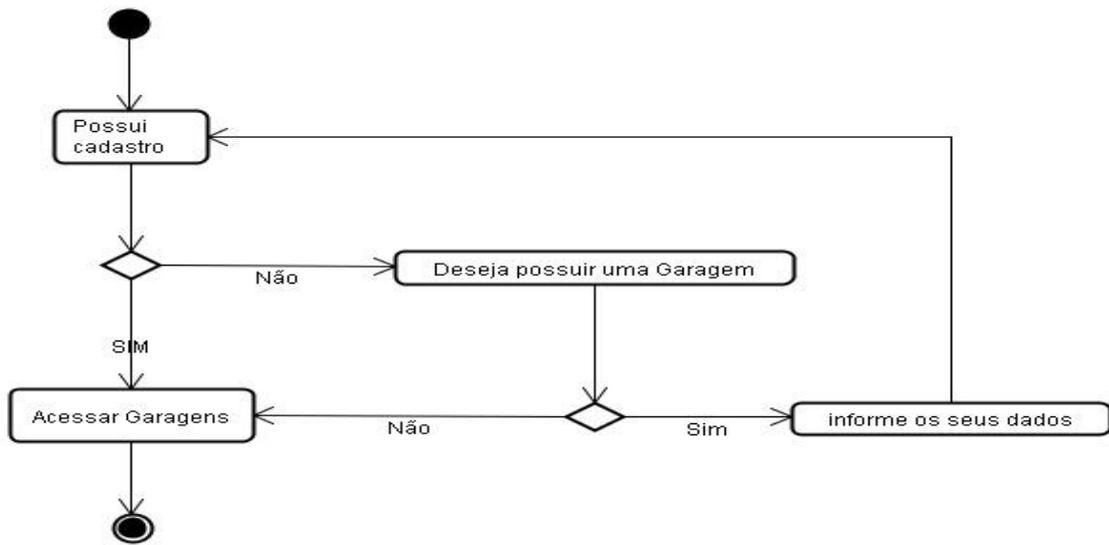


Figura 8. Diagrama de Atividades Cadastro.

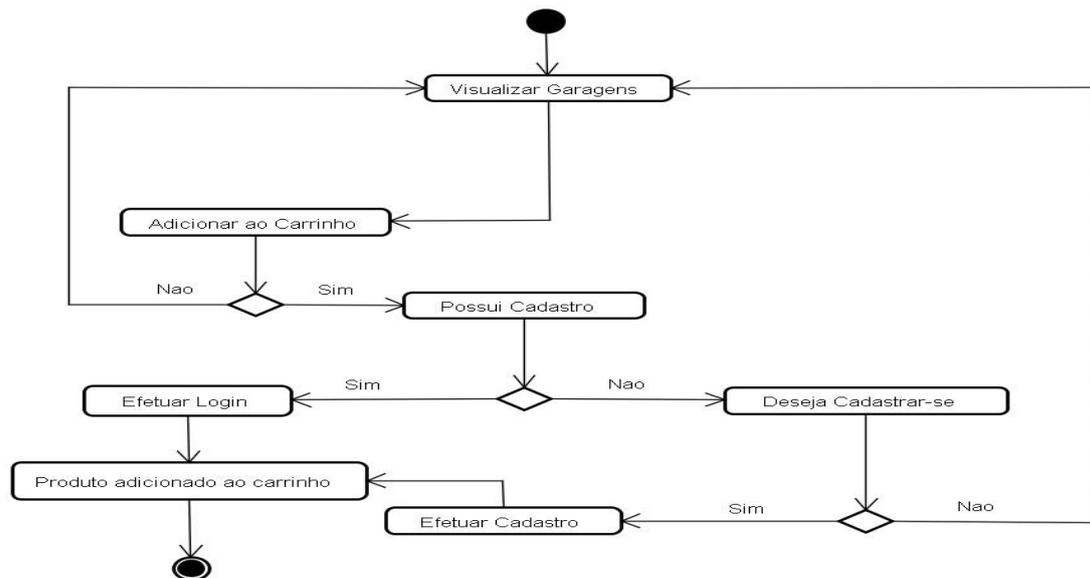


Figura 9. Diagrama de Atividades Compra.

3.5 DIAGRAMA DE CLASSES

Um diagrama de classes é uma representação da estrutura e relações das classes que servem de modelo para objetos. É muito útil para o sistema, pois define todas as classes que o sistema necessita possuir (MACORATTI, 2012).

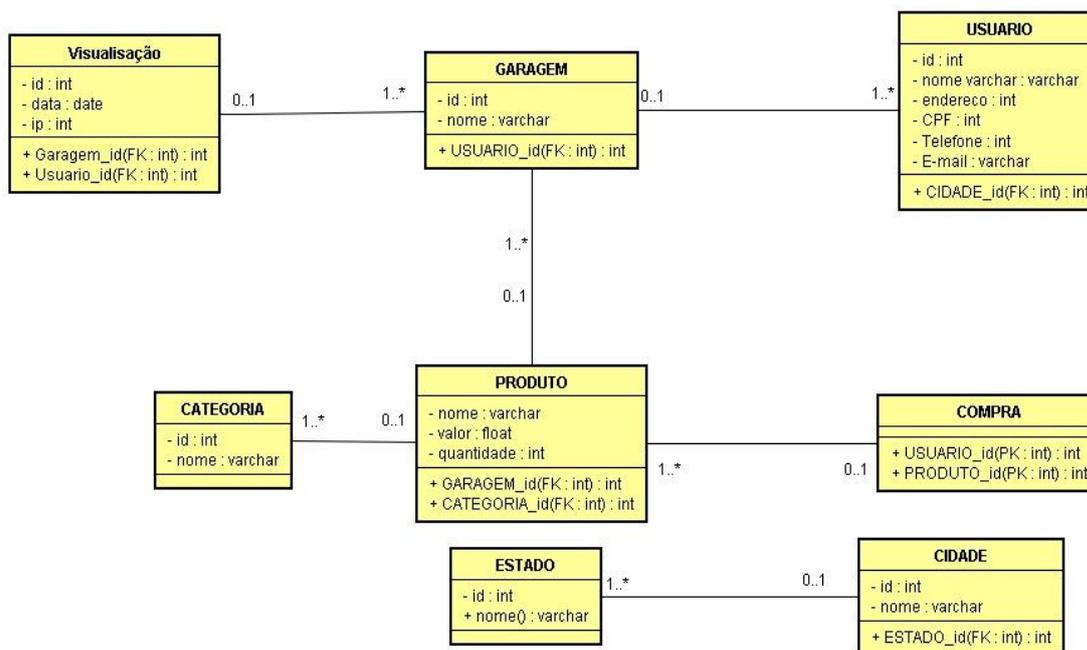


Figura 10. Diagrama de Classes – Model.

3.6 MODELAGEM DE ENTIDADE E RELACIONAMENTO

Um diagrama Entidade-Relacionamento é um modelo que descreve o modelo de dados de um sistema com alto nível de abstração. Foi desenvolvido para facilitar o projeto de banco de dados, permitindo a especificação de um esquema de negócio, onde tal esquema representa a estrutura lógica geral do banco de dados (REZENDE, 2005).

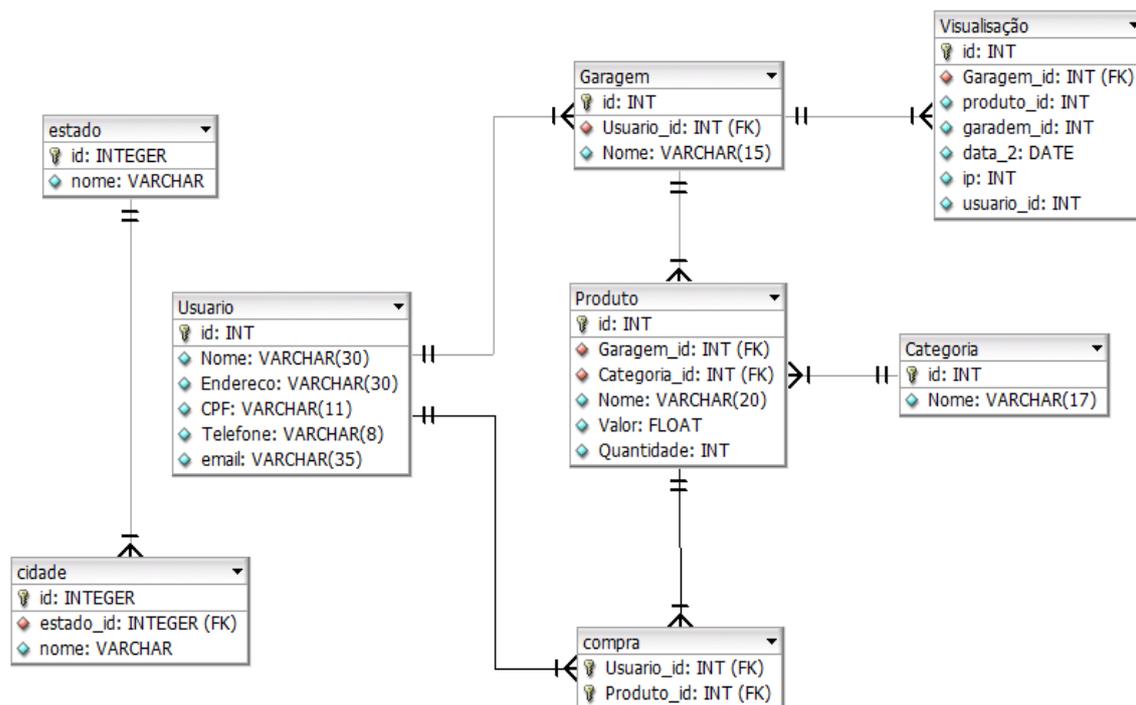


Figura 11. Modelo Entidade-Relacionamento.

4 ESTRUTURA DO PROJETO

Neste capítulo apresentarei a metodologia de desenvolvimento utilizada para o trabalho.

4.1 ESTRUTURA ANALÍTICA DE TRABALHO

A WBS é uma ferramenta utilizada para subdividir o trabalho de um projeto em partes menores que podem ser gerenciadas com maior facilidade, fornecendo uma ilustração detalhada do escopo do projeto, auxiliando na montagem da equipe e distribuição de trabalho, facilitando a identificação de riscos (TAMIRES, 2011).

É a peça central no planejamento de um projeto, uma vez que ela permite definir o conjunto de atividades que precisa ser executado. É com base na WBS que todos os elementos do projeto são planejados (MARTINS, 2010).

Portanto para a realização deste projeto, serão desenvolvidas as seguintes tarefas.

Work Breakdown Structure

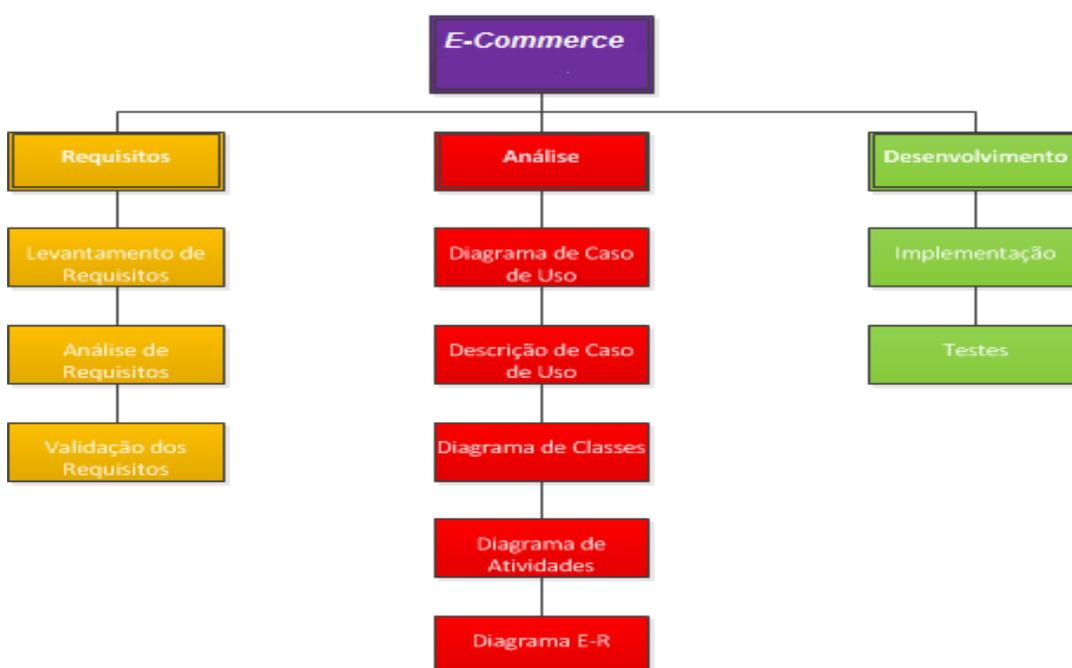


Figura 12. *Work Breakdown Structure*

4.2 SEQUENCIAMENTO DE ATIVIDADES

O diagrama de sequenciamento de atividades mostra o tempo de duração para a realização de cada atividade desenvolvida no decorrer do projeto.

Sequenciamento de Atividades

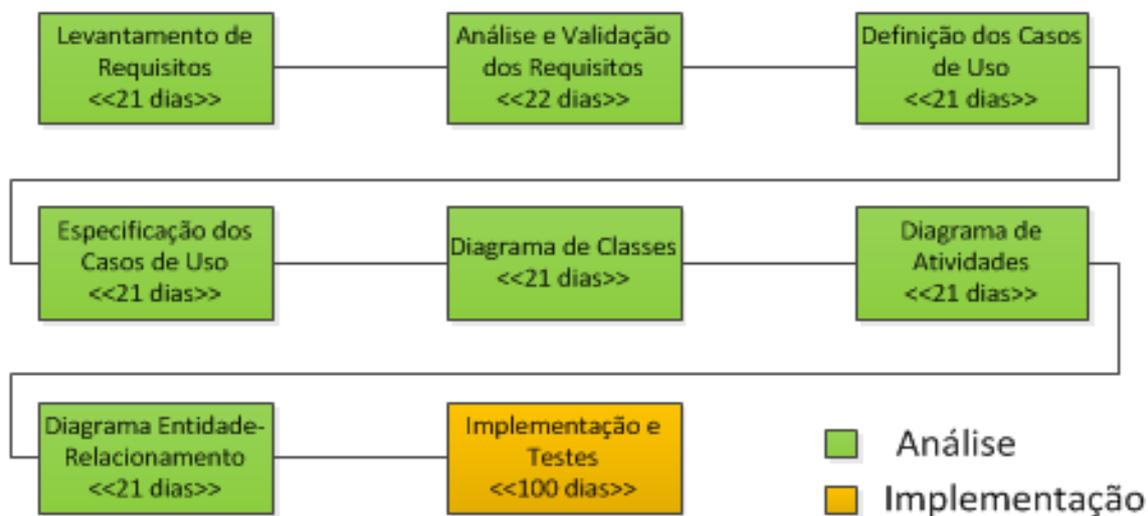


Figura 13. Sequenciamento de Atividades.

4.3 Orçamento

Orçamento Garage Sale			
Analista – Programador			
Analista – Programador	Quantidade de Horas	Valor Hora	Total
Caique F. Lopes	180	R\$ 50	R\$ 9.000,00
Equipamentos			
Equipamentos	Valor	Depreciação Diária	
Notebook	R\$ 3.000,00	R\$ 3,00	
Microsoft Visual Studio 2010	R\$ 1.317,00	R\$ 1,50	
Total			R\$ 13,32

5 Implementação do site

Para a implementação do sistema foi utilizado o ambiente de desenvolvimento Visual Studio 2012 com a linguagem de programação Asp.net. Os códigos-fontes foram divididos em três camadas, sendo estas:

- **Camadas Interface** - Projeto responsável pela camada de visualização do sistema, nela foi desenvolvido todas as telas do sistema, utilizando HTML5, CSS3, JS, JQUERY e programação Asp.net.
- **BLL** – Projeto responsável pela camada da regra de negócio, verifica os erros do formulário e só depois passa os dados para a camada DAL.
- **DAL** – Projeto responsável pela camada de comunicação com o banco, mapeamento de cada classe criada, métodos assessores e modificadores.
- **Model** – Projeto responsável pela get e set dos atributos.

5.1 Camadas de Interface

Neste projeto foi criada toda a parte de interface do sistema.

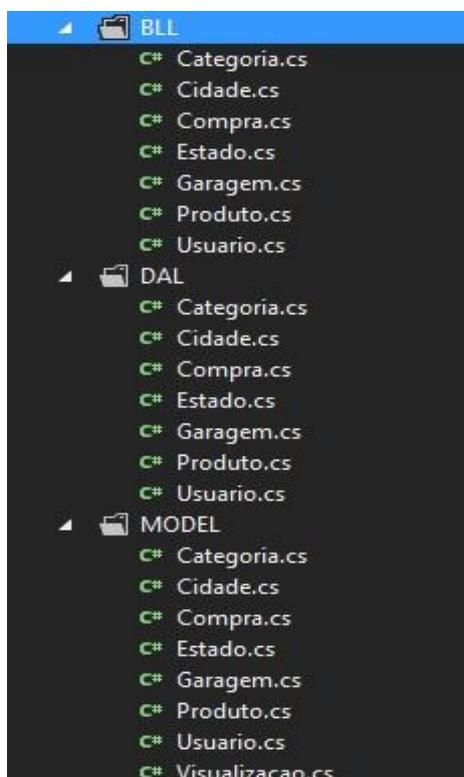


Figura 14. Camadas do projeto

No projeto HTML5 e CSS3, foram criadas as interfaces gráficas junto com os efeitos obtidos por estas.

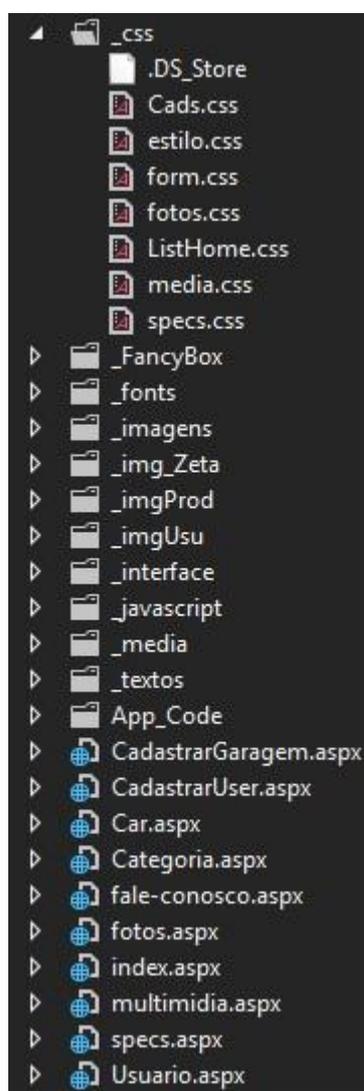


Figura 15. CSS3 e HTML5

Foi utilizado o Java Script e o JQuery, para criar alguns efeitos no ambiente do *site*.

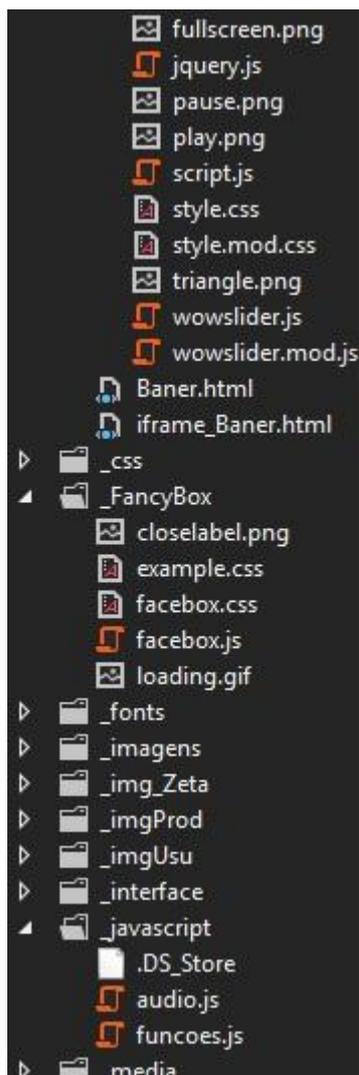


Figura 16. Java Script e JQuery.

5.1.1 Lista de Garagens

Para que cada garagem ao ser criada, seja formada uma ao lado da outra, de formato retangular foi utilizado uma lista não ordenada (), dentro desta lista possui os itens da lista (). As características foram atribuídas ao “list” item, ou seja, tudo o que for “list” item possuirá a mesma característica.

```

<!-- INICIO DO ARTIGO -->
<article id="principal">
  <header id="cabecalho-art">
    <hgroup>
      <h3>Home</h3>
      <h2>por Caique Lopes</h2>
      <h3 class="dir">Atualizado em 03/Jul/2014</h3>
    </hgroup>
  </header>

  <div id="Busc">
    <asp:TextBox ID="txtbusca" runat="server"></asp:TextBox>
    <asp:Button ID="Button3" runat="server" Text="Buscar" OnClick="Button3_Click" ForeColor="#999999" BackColor="White" BorderColor="Wh
  </div>

  <ul id="List">
    <% foreach(MODEL.Garagem garagem in listaGaragens) { %>
      <% if (garagem != null) { %>
        <% Image1.ImageUrl = "~/\" + garagem.Foto; %>
        <% if(garagem.Foto.Equals("")) %>
          <% Image1.ImageUrl = "~/_img_Zeta/Logo.jpg"; %>
        <li><asp:Image ID="Image1" runat="server" width="170" height="200" ImageUrl="~/_img_Zeta/Logo.jpg"/><span><%= garagem.nome %>
      <% } %>
    <% } %>
  </ul>

```

Figura 17. Código pagina inicial.

5.1.2 Cadastrar Foto

Ao cadastrar uma garagem, é solicitada a seleção de uma foto para a mesma.

```

protected void Button1_Click(object sender, EventArgs e)
{
    DAL.Garagem garagemDal = new DAL.Garagem();

    if (garagemDal.garagemjaExiste(txtnome.Text) == false)
    {
        if (garagemDal.usuarioComGaragem(Convert.ToInt32(Session["usuario_id"])) == false)
        {
            MODEL.Garagem garagem = new MODEL.Garagem();
            garagem.nome = txtnome.Text;
            garagem.usuario = new MODEL.Usuario();
            garagem.usuario.id = Convert.ToInt32(Session["usuario_id"]);

            if (FileUpload1.HasFile)
            {
                FileUpload1.SaveAs("D:\\Desktop\\projeto_zeta\\_imgUsu\\" + garagem.id.ToString() + FileUpload1.FileName);
                garagem.Foto = "_imgUsu\\" + garagem.id.ToString() + FileUpload1.FileName;
            }

            garagemDal.Insert(garagem);
        }
        else {
            System.Web.HttpContext.Current.Response.Write("<SCRIPT LANGUAGE='JavaScript'>alert('Nome indisponivel')</SCRIPT>");
        }
    }
}

```

Figura 18. Código cadastro de fotos.

Para que a imagem seja exibida é necessário que o usuário escolha uma imagem de sua preferência, assim esta imagem será salva em uma pasta local, tornando assim acessível sua exibição.

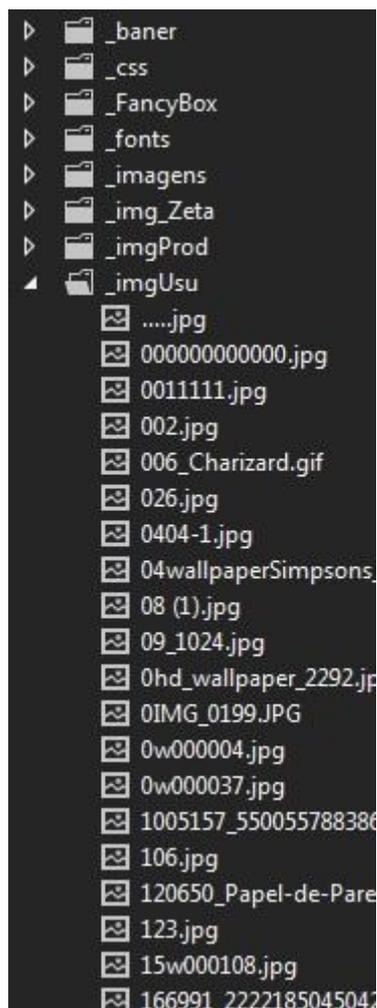


Figura 19. Pasta das fotos cadastradas.

5.2 Organização do Projeto BLL

Projeto onde se localiza todas as regras do negocio.

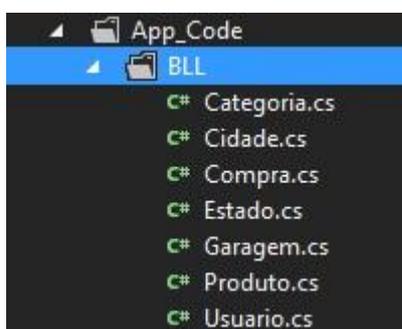


Figura 20. Camada BLL.

Essa classe verifica alguns erros, evitando dados incorretos no sistema e logo após faz uma chamada a classe DAL passando os dados para comunicação com o banco.

5.3 Organização do Projeto DAL

Projeto onde se localiza todas as classes DAL.

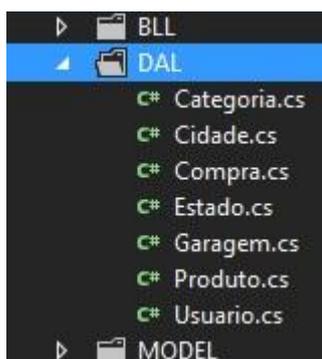


Figura 21. Camada DAL.

Nesta classe contém o endereço do banco, além de serem codificados todos os métodos utilizados do sistema, métodos como SELECT, INSERT, DELET, UPDATE entre outros. O DAL recebe os dados digitados no formulário pelo usuário, e grava no banco, essa ação é feita pela passagem de parâmetro.

```

con.Open();
SqlDataAdapter da = new SqlDataAdapter(cmd);
cmd.Parameters.AddWithValue("@id", id);
SqlDataReader reader = cmd.ExecuteReader(CommandBehavior.CloseConnection);
MODEL.Produto oProd = null;
DAL.Categoria categoriaDal = new DAL.Categoria();
DAL.Garagem garagemDal = new DAL.Garagem();
try
{
    if (reader.Read())
    {
        oProd = new MODEL.Produto();
        oProd.id = Convert.ToInt32(reader["id"]);
        oProd.nome = reader["nome"].ToString();
        oProd.categoria = categoriaDal.SelectId(Convert.ToInt32(reader["categoria_id"]));
        oProd.garagem = garagemDal.SelectId(Convert.ToInt32(reader["garagem_id"]));
        oProd.valor = Convert.ToInt32(reader["valor"]);
        oProd.foto1 = reader["Foto01"].ToString();
        oProd.descricao = reader["descricao"].ToString();
    }
}
finally
{
    con.Close();
}
return oProd;
}

public void Insert(MODEL.Produto oProd)
{
    SqlConnection con = new SqlConnection(strCon);
    string sql = "insert into produto values (@categoria_id,@garagem_id,@nome,@valor,@Foto01,@descricao)";
    SqlCommand cmd = new SqlCommand(sql, con);

```

Figura 22. Código camada DAL.

5.4 Organização do Projeto Model

Projeto onde se localiza todas as classes MODEL.

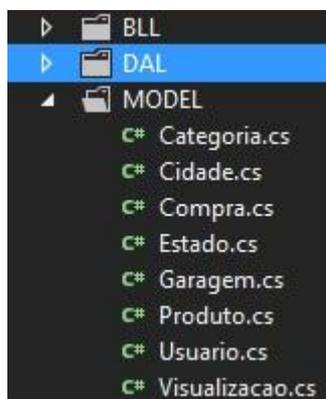


Figura 23. Camada MODEL.

Nesta classe foram realizados todos os métodos “get” e “set” das variáveis. Tais métodos permitem a manipulação da mesma. Pois são métodos de entrada e saída.

```
using System;
using System.Collections.Generic;
using System.Linq;
using System.Web;

namespace MODEL
{
    public class Garagem
    {
        public int id { get; set; }
        public Usuario usuario { get; set; }
        public string nome { get; set; }
        public string Foto { get; set; }
    }
}
```

Figura 24. Código camada MODEL.

5.5 Interfaces do Sistema

5.5.1 Interface Principal

Essa é a tela inicial do *site*. Nela se encontra a lista de todas as garagens existentes, uma área de *Login*, um campo de Busca, um Banner Interativo, um menu, e uma área para Compras.



Figura 25. Pagina inicial do Site.

5.5.2 Interface *Login*

O usuário deve criar seu próprio *login* para efetuar compras ou para obter o seu produto à venda, anunciado no *site*.



Figura 26. Área de *login*.

Para criar o cadastro é necessário fornecer informações pessoais como: Nome, Apelido, CPF, Logradouro, Cidade, Estado, Telefone, E-mail. O usuário deverá criar um nome de usuário e senha, e aceitar os termos de uso do *site*.

Figura 27. Tela de cadastro de usuário.

5.5.3 Campo de Busca

O campo de busca fará a pesquisa das garagens já criadas, por nome.

Figura 28. Campo de pesquisa.

5.5.4 Área de compra (carrinho)

Área onde armazena os produtos escolhidos pelo usuário para compra.

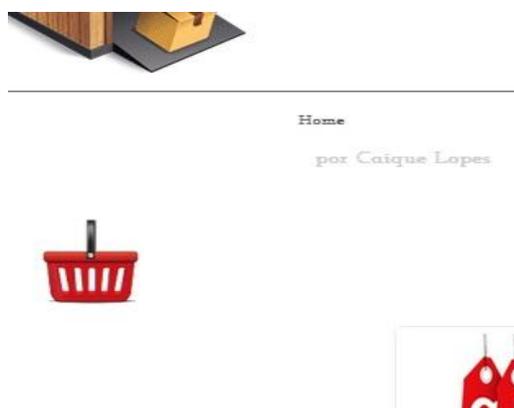


Figura 29. Carrinho de compras.

5.5.5 Banner Interativo

Área de propaganda do *site*. O banner foi feito em JQUERY e foi usado o programa WOWSlider para a sua criação.



Figura 30. Banner do *site*.

5.5.6 Menu

Nele contém as páginas de Home, Categorias e Fale conosco.

- Home: Nesta área o usuário poderá visualizar as garagens existentes juntamente com os seus produtos a serem vendidos.
- Categorias: Nesta área o usuário poderá pesquisar a categoria do produto escolhido para a compra.
- Fale conosco: Nesta área o usuário terá as informações necessárias para contato.



Figura 31. Menu do *site*.

6 Conclusão

Conclui que a qualificação foi uma parte fundamental no trabalho, pois me auxiliou na incorporação de conhecimentos teóricos, técnicos e operacionais.

Pude obter uma visão ampla do *site* e das ferramentas que foram utilizadas na programação do mesmo ficar pronto.

Para o desenvolvimento foram utilizados os conceitos de programação em camadas, o que ajudou muito a entender melhor o código e deixa-lo mais enxuto, pois tal padrão divide o sistema em etapas, separando métodos de acesso ao banco, regra de negócios e interfaces do sistema.

Durante a implementação do sistema foi utilizado a ferramenta Visual Studio 2012, junto ao banco de dados SQL Server. Surgiram varias dificuldades durante o desenvolvimento do sistema, as quais foram sanadas por meio de conversas com o orientador.

6.1 Trabalhos Futuros

Como todo E-commerce o “Garage Sale”, ainda não possui as formas de pagamentos de produtos, como cartões ou boleto. Porém estou trabalhando para inclui-las.

REFERENCIAS BIBLIOGRAFICAS

BOOCH, Grady; JACOBSON, Ivar; RUMBAUGH, James. UML Guia do Usuário. 2º Edição. Tradução Fábio Freitas da Silva e Cristiana de Amorim

BOOCH, Grady; JACOBSON, Ivar; RUMBAUGH, James. UML Essencial – Um breve guia para a linguagem-padrão de modelagem de objetos. 2º Edição. Tradução de Vera Pezerico e Christian Thomas. Porto Alegre: Borkman, 2000.

GALLO, Alessandro; BARKOL, David; KRISHNA, Rama. Asp.Net em Ação; 1ª Edição, 2009;

GUEDES, Gilleanes T. A; UML2 Uma Abordagem Prática; 2ª Edição;

Head First Ajax REBECCA M. RIORDAN A BRIAN-FRIENDLY GUIDE

REFERENCIAS DIGITAL

HTML5. Disponível em:

<<http://msdn.microsoft.com/pt-br/library/dn151487.aspx>> Acesso em: 26 de Fevereiro de 2014.

Asp. Net. Disponível em:

<[http://technet.microsoft.com/pt-br/Library/cc775831\(v=ws.10\).aspx](http://technet.microsoft.com/pt-br/Library/cc775831(v=ws.10).aspx)> Acesso em: 26 de Fevereiro de 2014.

JQuery. Disponível em:

<<http://jquery.com/>><<http://pt.wikipedia.org/wiki/JQuery>> Acesso em: 27 de Fevereiro de 2014.

CSS3. Disponível em:

<<http://revistaw.com.br/galeria/sites-que-abusam-de-css3/>><<http://pt.wikipedia.org/wiki/CSS3>> Acesso em: 27 de Fevereiro de 2014.

Ajax. Disponível em:

<<http://www.w3schools.com/ajax/default.ASP>><[http://pt.wikipedia.org/wiki/AJAX_\(programa%C3%A7%C3%A3o\)](http://pt.wikipedia.org/wiki/AJAX_(programa%C3%A7%C3%A3o))> Acesso em: 28 de Fevereiro de 2014.

