

**CENTRO UNIVERSITARIO UNIVESC
CURSO DE CIÊNCIA DA COMPUTAÇÃO
TRABALHO DE CONCLUSÃO DE CURSO**

E-COMMERCE – ANGEL E RE ACESSÓRIOS

Renata Hoefling

Lages (SC), Novembro de 2012.

**CENTRO UNIVERSITARIO UNIVESC
CURSO DE CIÊNCIA DA COMPUTAÇÃO
TRABALHO DE CONCLUSÃO DE CURSO**

E-COMMERCE – ANGEL E RE ACESSÓRIOS

Área de Programação Web

Renata Hoefling

Projeto apresentado à Banca
Examinadora do Trabalho de Conclusão
do Curso de Ciência da Computação
para análise e aprovação.

Lages (SC), Novembro de 2012.

EQUIPE TÉCNICA

Acadêmico

Renata Hoefling.

Professor Orientador

Prof. Marcio Jose Sembay, Msc.

Coordenador de TCC

Prof. Marcio Jose Sembay, Msc.

Coordenador do Curso

Marcio Sembay, Msc.

AGRADECIMENTOS

Hoje vivo uma realidade que parece um sonho, mais para chegar até aqui foi preciso muito esforço, determinação, paciência, e nada disso eu conseguiria sozinha. Com certeza há muito o que agradecer, pois sem o auxílio e compreensão de todos, este trabalho não chegaria ao término.

À DEUS, primeiramente, pelo dom da vida, pelo seu amor infinito, que sempre está ao meu lado e por privilegiar-me de cumprir essa missão em minha vida;

A minha mãe, que me deu toda a estrutura para que me tornasse a pessoa que sou hoje, pelo importante apoio, amor incondicional, compreensão, confiança que me fortalece todos os dias. Esta caminhada não seria a mesma sem você;

Aos familiares e amigos pelo companheirismo, e por sempre acreditarem no meu trabalho, me apoiando nos momentos de angústias, alegrias e decepções, contribuindo para enfrentar as situações difíceis.

A todos os meus professores, pelo esforço, determinação e dedicação que desempenharam as aulas ministradas;

Ao meu melhor amigo, namorado e companheiro de todas as horas, Orlen Camargo Arruda, que desde o início me auxiliou sem medir esforços na busca desse sonho. Obrigada pela paciência, pelo sorriso, pelo abraço, pela mão que sempre estendia quando eu precisava, sempre apoiando em todas as minhas decisões;

Aos meus colegas de classe, a quem constitui laços eternos. Obrigada por todos os momentos, por todo apoio e cumplicidade;

A minha eterna gratidão a todos aqueles que colaboraram para que este sonho pudesse ser concretizado.

SUMÁRIO

EQUIPE TÉCNICA	I
AGRADECIMENTOS	II
SUMÁRIO	III
LISTA DE FIGURAS	VI
ÍNDICE DE TABELAS	VII
RESUMO	VIII
ABSTRACT	IX
I – INTRODUÇÃO	I
1. APRESENTAÇÃO	X
2. JUSTIFICATIVA	XI
3. IMPORTÂNCIA.....	XII
4. OBJETIVOS	XII
4.1. Objetivo Geral	xii
4.2. Objetivos Específicos.....	xii
5. METODOLOGIA.....	XIII
6. CRONOGRAMA.....	XIV
II - REVISÃO BIBLIOGRÁFICA.....	XV
1. PROGRAMACAO WEB	XV
1.1. Internet	xvi
1.1.1. Internet 1.0	Erro! Indicador não definido.
1.1.2. Internet 2.0	Erro! Indicador não definido.
1.1.3. Internet 3.0	8
1.2. Comercio Eletronico.....	xv
1.2.1. Ambiente de compra	8
1.3 Banco de Dados.....	xviii
1.3.1 <u>Sistema de Gerenciamento de Banco de Dados</u>.....	xviii
1.3.1.1 <u>Vantagens e Desvantagens do uso de um SGBD</u>	10
1.3.1.1.1 <u>Controle de reduncia</u>.....	10
1.3.1.1.2. <u>Representacao de relacionamentos complexos entre dados</u>.....	xix

1.4. UNIFIED MODELING LANGUAGE (UML)	xx
1.4.1 <u>Conceituando a UML</u>	xx1
1.4.2. Fases do Desenvolvimento de um Sistema em UML	xx
1.4.2.1. <u>Análise de Requisitos</u>	12
1.4.2.2. <u>Análise</u>	xxi2
1.4.2.3. <u>Design (Projeto)</u>	xxi
1.4.2.4. <u>Programação</u>	xxii3
1.4.2.5. <u>Testes</u>	xxii3
2. LINGUAGENS E FERRAMENTAS	xxiii4
2.1 . Rational Software Modeler 7.0	xxiii4
2.2. HTML	xxiv5
2.2.1 CSS	Erro! Indicador não definido.
2.3. PHP	Erro! Indicador não definido.6
2.4. Dreamweaver	Erro! Indicador não definido.6
2.5. MySQL	xxv
III - MODELAGEM DO SISTEMA	19
1. Diagramas	19
1.1. <u>Diagrama de Caso de Uso</u>	19
1.1.1 <u>Caso de Uso</u>	19
1.1.2 <u>Ator</u>	20
1.1.3 <u>Relacionamento</u>	20
1.2. <u>Diagrama de Classe</u>	21
1.2.1 <u>Classe</u>	21
1.2.2 <u>Relacionamento</u>	22
1.3. <u>Diagrama de Seqüência</u>	23
1.4. <u>Diagrama de Componentes</u>	25

1.4. <u>Diagrama de Atividades</u>.....	26
V - CONSIDERAÇÕES FINAIS	27
VI - REFERÊNCIAS	28
II	

LISTA DE FIGURAS

Figura 1	Etapas do desenvolvimento do TCC I	4
Figura 2	Tela principal do <i>Rational Software Modeler</i>	14
Figura 3	Tela do <i>Dreamweaver</i>	17
Figura 4	Diagrama de Caso de Uso do Sistema Proposto.....	21
Figura 5	Diagrama de Classe do Sistema Proposto	23
Figura 6	Diagrama de Seqüência do Sistema Proposto	24
Figura 7	Diagrama de Componentes do Sistema Proposto.....	25
Figura 8	Diagrama de Atividades do Sistema Proposto.....	26

ÍNDICE DE TABELAS

Tabela 1	Cronograma do TCC	5
----------	-------------------------	---

RESUMO

O trabalho proposto tem como objetivo apresentar o desenvolvimento do módulo de personalização de um sistema de comércio eletrônico. O objetivo deste módulo é implementar as funcionalidades referentes aos cadastros e à vitrine virtual de um comércio eletrônico com a personalização dos produtos oferecidos, além de fornecer recursos de segurança. O desenvolvimento deste projeto foi motivado visando a uma melhor integração da empresa na comercialização on line.

Uma ferramenta que facilitará a interação entre cliente e a empresa. Na modelagem deste sistema será utilizado a linguagem UML – Linguagem Unificada de Modelagem, e será implementado em HTML, utilizando Banco de Dados.

ABSTRACT

The proposed work aims to present the development of a personalization module e-commerce system. The purpose of this module is to implement the functionalities relating to registrations and storefront with an e-commerce personalization of products offered, in addition to providing security features. The development of this project was motivated in order to better integrate the company's online marketing.

A tool that will facilitate the interaction between customer and company. In this modeling system will be used UML - Unified Modeling Language, and is implemented in HTML using Database.

I – INTRODUÇÃO

1. APRESENTAÇÃO

O *e-commerce* é um eficiente canal de vendas e marketing. Diariamente as empresas aproximam-se mais de seu público via web aumentando significamente o crescimento do varejo sem loja, com baixo custo, fazendo com que os consumidores possam receber suas mercadorias, muitas vezes sem taxa de entrega, facilitando a negociação.

Churchill (2003), o processo de compra é influenciado por fatores situacionais, sociais e de *marketing*. Para ele, deve-se entender intrinsecamente a necessidade específica de cada consumidor e supri-la.

Com o crescimento expressivo da tecnologia, o uso da internet possibilita as empresas dominarem todo o processo do marketing, venda, pagamento através da internet suprimindo as necessidades da empresa de se relacionar diretamente com o seu cliente maximizando os negócios, com milhares de pessoas de diversos lugares em um mesmo momento. Assim o consumidor poderá sentir-se em uma loja, porém não saberá exatamente a forma, a cor dos produtos a menos que tenha testado no varejo físico. Com a internet o cliente poderá dispor de mais tempo para suas compras.

Maslow (1976), as necessidades humanas se apresentam segundo uma hierarquia. Uma escala de valores, que uma vez atendidos, ascende-se à níveis superiores para satisfação de novos anseios.

2. JUSTIFICATIVA

Este trabalho teve como objetivo desenvolver um sistema de comércio eletrônico, responsável pelo cadastro de categorias, características, produtos e promoções, responsável também pela vitrine virtual, proporcionando a personalização das informações apresentadas, além da abordagem sobre segurança nas transações comerciais que necessitam de confiabilidade e integridade das informações.

3. IMPORTÂNCIA

A empresa para se destacar necessita integrar-se a tecnologia. O “E-commerce Angel e Re acessórios ” apresenta produtos aos usuários, fazendo com que ele sintasse motivado para realizar a compra. Partindo-se da premissa que os produtos são de interesse do usuário e estão sendo oferecidos a um preço justo, transmissão de informações precisas ao comprador; além da confiabilidade, segurança e facilidade de navegação encontrada.

O e-commerce, ferramenta útil, possibilita que qualquer um visualize produtos disponíveis em qualquer lugar do mundo, fazendo com que as mercadorias sejam conhecidas em diversos lugares, antes nem imaginados, criando novas possibilidades de público e assim, gerando mais receita.

4. OBJETIVOS

4.1. Objetivo Geral

O e-commerce “ Angel e Re acessórios” pode reduzir todos os custos do ciclo comercial, aproximar clientes de mercados, despertando o interesse pelo produto, permitindo uma compra rápida e fácil. Criando assim novas possibilidades de público e uma maior produtividade.

4.2. Objetivos Específicos

Os objetivos específicos deste trabalho são:

- Mostrar a relação entre logística e *E-commerce* como desenvolvimento de novas técnicas para o alcance de metas e atendimento ao mercado, satisfazendo as necessidades de pessoas;
- Analisar a segurança e a facilidade para o consumidor nas formas de pagamento desta nova ferramenta tecnológica;

- Analisar a viabilidade para a implantação de um *e-commerce*, verificando as vantagens e desvantagens para o consumidor e a organização.

5. METODOLOGIA

Trate-se de uma pesquisa de abordagem, tendo por finalidade a implantação de um *e-commerce*.

A Figura 1 demonstra as etapas de desenvolvimento do TCC (Trabalho de Conclusão de Curso).

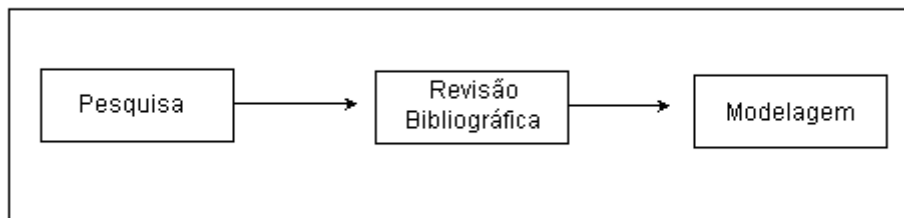


Figura 1: Etapas do desenvolvimento do TCC

Este estudo iniciou-se com o levantamento bibliográfico para alcançar os objetivos do TCC (Trabalho de Conclusão de Curso), indicados na figura 1. Este material é proveniente, basicamente, de livros, apostilas e de páginas *Web*, sendo este material encontrado em diversas fontes.

Durante a revisão bibliográfica, baseando-se em todo material pesquisado anteriormente, serão efetuadas todas as fundamentações teóricas do trabalho, desde conceituação básica até a conclusão final do mesmo.

A modelagem da ferramenta proposta será a última etapa para a conclusão do trabalho. Nesta, foi utilizada a linguagem de modelagem UML (*Unified Modeling Language*), onde é desenvolvido principalmente um escopo inicial do sistema.

Segundo as autoras Marconi e Lakatos (2005), o objetivo da pesquisa bibliográfica é colocar o pesquisador em contato com tudo o que já foi escrito e publicado sobre o referido tema de estudo.

6. CRONOGRAMA

A metodologia apresentada anteriormente delimita os passos para o desenvolvimento do TCC.

O cronograma é definido na Tabela 01.

ATIVIDADES	JUL	AGO	SET	OUT	NOV	DEZ
Pesquisa	X	X	X	X		
Revisão Bibliográfica			X	X	X	X
Modelagem do Software				X	X	X
Entrega e defesa do TCC II a banca avaliadora					X	X

Tabela 01. Cronograma do TCC

II - REVISÃO BIBLIOGRÁFICA

1. PROGRAMAÇÃO WEB

Programação WEB é a programação de sites para a internet, e tem como finalidade disponibilizar informações e/ou aplicações para o usuário. A programação WEB pode variar desde simples páginas estática, comércio eletrônico (*e-commerce*) ou redes sociais.

Na programação WEB temos de um lado o cliente web, ou browser, que solicita dados ao servidor web, recebe respostas, formata a informação e a apresenta ao usuário. Do outro lado está o servidor web que recebe as requisições, lê os dados (páginas HTML) do disco e as retorna para o cliente. Os programas que executam no browser normalmente são scripts que tem a capacidade de perceber os eventos causados pelo usuário e responder de forma apropriada. Os scripts interagem com todos os elementos que formam uma página HTML.

1.1. Internet

Segundo Hortinha (2002), a Internet foi criada para fins acadêmicos e militares. Porém a partir de 1993, quando foi utilizada para fins comerciais, passou a ser uma rede global de redes interligadas, incluindo redes privadas, de governos, de empresas e outras organizações.

Para os próprios internautas que escrevem através da *wikipedia* (enciclopédia livre da Internet) a definição de Internet é um conglomerado de redes em escala mundial de milhões de computadores interligados.

Podemos classificar a internet através de suas fases: 1.0, 2.0 e 3.0. Esta classificação é a nomenclatura dada para as diferentes fases da Internet. Elas ocorrem de acordo com inovações tecnológicas e novas funcionalidades para a Internet, demarcando o início de uma nova fase. Pode-se dizer que as próximas fases são as somas das anteriores com diferenciais.

1.1.1. Internet 1.0

A Internet 1.0, como foi classificada, era estática. Os usuários da mesma não produziam conteúdo, eles apenas recebiam as informações postadas pelos sites. Foi a fase de criação dos grandes portais.

1.1.2. Internet 2.0

Para O'REILLY (2005), pode ser entendida como uma perda da passividade visualizada pela Internet 1.0 e obtenção da interatividade. Há nessa fase da Internet, um maior controle por parte do usuário, maior personalização de conteúdos e serviços disponíveis, a participação desse mesmo usuário no processo de troca de informações na Internet.

No caso da Internet 2.0, instaura-se a interatividade. Onde os usuários são também geradores de conteúdo. O maior expoente da Internet 2.0 são as tão aclamadas redes sociais e o comércio eletrônico (e-commerce). As redes sociais, são redes de relacionamento que integram seus usuários, fazendo com que eles compartilhem informações, fotos, interesses, músicas e etc. Seus principais exemplos são o *Face book*, *Orkut*, *Twitter*, *Myspace*, *LinkedIn*, entre outras. As redes sociais mantêm conectados jovens, adultos e crianças de todo o mundo. Quanto ao comércio eletrônico, este terá uma maior ênfase no tópico abaixo.

1.1.3. Internet 3.0

Essa nomenclatura foi dada pelo jornalista John Markoff do jornal americano *The New York Times*. Seria a terceira onda da Internet e considerada como a fase onde se teria um aproveitamento melhor das tecnologias já disponíveis.

Segundo a *wikipedia*, a fase 3.0 será uma fase de organização e uso de uma forma mais inteligente do conhecimento que já se encontra disponível na Internet e criado pela própria ou pelos usuários. Essa porém não provoca uma ruptura tão grande quanto percebe-se na passagem da primeira fase para a segunda.

1.2. Comércio Eletrônico (e-commerce)

O *e-commerce*, que em português significa comércio eletrônico, é uma modalidade de comércio que realiza suas transações financeiras por meio de dispositivos e plataformas eletrônicas,

como computadores e celulares. Um exemplo desse tipo de comércio é comprar ou vender produtos em lojas virtuais.

Segundo Kotler (2000), o termo e-commerce significa ser uma ampla variedade de transações eletrônicas, tais como o envio de pedidos de compra para fornecedores via EDI (troca eletrônica de dados). Ainda segundo Kotler, por trás dos negócios eletrônicos existem dois fenômenos: o da conectividade e da digitalização.

Historicamente o e-commerce nasceu nos Estados Unidos no ano de 1995 com o surgimento da Amazon.com e outras empresas. No Brasil esse processo começou cerca de cinco anos depois. No início (idos de 2001) o e-commerce era utilizado basicamente para vender bens tangíveis com valores modestos como livros e CDs. Hoje ele é utilizado para comercializar desde produtos que custam milhões até produtos que há pouco tempo eram inimagináveis pela sua incompatibilidade com esse tipo de comércio como roupas, perfumes e alimentos.

1.2.1. Ambiente de Compra

Diferentemente das lojas de varejo tradicionais, as lojas online não tem uma interface presencial com o cliente, o que faz com sejam requeridas adaptações e características diferentes. A forma como a marca se apresenta aos clientes deve ser diferente, bem como suas estratégias de retenção de atenção. Na loja online, o consumidor não pode tocar no produto, nem testá-lo. Devendo assim confiar no que está sendo apresentado, em suas funcionalidades, consistências e outros atributos que são tangíveis por natureza.

Para Alencar, Helena & Menezes (2008), “o ambiente da loja é virtual e a interação ocorre por meio da página de web, portanto o design é relevante para o consumidor desse segmento”. E para trazer esse consumidor à essa realidade criada, precisam-se de funcionalidades melhores do que simplesmente os “splashes” amarelos com letras vermelhas. Precisa-se de uma arquitetura harmônica, de forma que o consumidor não somente tenha interesse de utilizá-la para comparação de preço, mas sim, como fim de compra.

1.3. Banco de Dados

Segundo PEARSON (2005), Um “banco de dados” pode ser definido como um conjunto de “dados” devidamente relacionados. Por “dados” podemos compreender como “fatos conhecidos” que podem ser armazenados e que possuem um significado implícito. Porém, o significado do termo “banco de dados” é mais restrito que simplesmente a definição dada acima. Um banco de dados possui as seguintes propriedades:

- Um banco de dados é uma coleção lógica coerente de dados com um significado inerente; uma disposição desordenada dos dados não pode ser referenciada como um banco de dados.
- Um banco de dados é projetado, construído e populado com dados para um propósito específico; um banco de dados possui um conjunto pré-definido de usuários e aplicações.
- Um banco de dados representa algum aspecto do mundo real, o qual é chamado de “mini-mundo” ; qualquer alteração efetuada no mini-mundo é automaticamente refletida no banco de dados.

1.3.1. Sistema de Gerenciamento de Banco de Dados

Segundo KORTH (1995), um banco de dados pode ser criado e mantido por um conjunto de aplicações desenvolvidas especialmente para esta tarefa ou por um “Sistema Gerenciador de Banco de Dados” (SGBD). Um SGBD permite aos usuários criarem e manipularem banco de dados de propósito geral. O conjunto formado por um banco de dados mais as aplicações que manipulam o mesmo é chamado de “Sistema de Banco de Dados”.

1.3.1.1. Vantagens e desvantagens do uso de um SGBD

1.3.1.1.1. Controle de Redundância

No processamento tradicional de arquivos, cada grupo de usuários deve manter seu próprio conjunto de arquivos e dados. Desta forma, acaba ocorrendo redundâncias que prejudicam o sistema com problemas como:

- toda vez que for necessário atualizar um arquivo de um grupo, então todos os grupos devem ser atualizados para manter a integridade dos dados no ambiente como um todo;
- a redundância desnecessária de dados levam ao armazenamento excessivo de informações, ocupando espaço que poderia estar sendo utilizado com outras informações.

1.3.1.1.2. Compartilhamento de Dados

Um SGBD multi-usuário deve permitir que múltiplos usuários acessem o banco de dados ao mesmo tempo. Este fator é essencial para que múltiplas aplicações integradas possam acessar o banco. O SGBD multi-usuário deve manter o controle de concorrência para assegurar que o resultado de atualizações sejam corretos. Um banco de dados multi-usuário deve fornecer re para a construção de múltiplas visões.

1.3.1.1.3. Representação de Relacionamentos Complexos entre Dados

Um banco de dados pode incluir uma variedade de dados que estão inter-relacionados de várias formas. Um SGBD deve fornecer recursos para se representar uma grande variedade de relacionamentos entre os dados, bem como, recuperar e atualizar os dados de maneira prática e eficiente.

1.4. UNIFIED MODELING LANGUAGE (UML)

A modelagem do sistema foi realizada através do uso da *Unified Modeling Language* (UML), que em português significa Linguagem de Modelagem Unificada, e tem como finalidade demonstrar o funcionamento do próprio sistema, através dos diagramas de *use-case*, (caso de uso), e de classe.

1.4.1 Conceituando a UML

Segundo FURLAN (1998), UML é uma linguagem padrão para especificar, visualizar, documentar e construir artefatos de um sistema e pode ser utilizada com todos os processos ao longo do ciclo de desenvolvimento e através de diferentes tecnologias de implementação. A UML é usada no desenvolvimento dos mais diversos tipos de sistemas. Ela abrange sempre qualquer característica de um sistema em um de seus diagramas e é também aplicada em diferentes fases do desenvolvimento de um sistema, desde a especificação da análise de requisitos até a finalização com a fase de testes.

De acordo com Fowler (2000), a UML é uma linguagem de modelagem, e não um método. Onde a diferença entre método e linguagem de modelagem é que a maioria dos métodos consiste, de uma linguagem de modelagem e de um processo: a linguagem de modelagem é a notação utilizada por métodos para expressar projetos, e o processo é a seqüência de passos a ser seguida na elaboração de um projeto.

1.4.2. Fases do Desenvolvimento de um Sistema em UML

Segundo Deboni (1998), existem cinco fases no desenvolvimento de sistemas de software: análise de requisitos, análise, *design* (projeto), programação e testes.

1.4.2.1. Análise de Requisitos

Nesta fase, encontram-se as necessidades dos usuários do sistema a ser desenvolvido através do uso de determinadas funções chamadas *use-cases* (casos de uso). Com o desenvolvimento destas funções, as entidades externas ao sistema (atores), que interagem e possuem interesse no sistema, são modelados com relacionamentos que possuem comunicação associativa entre eles ou são desmembrados em hierarquia.

1.4.2.2. Análise

A análise está preocupada com as primeiras abstrações (classes e objetos) e mecanismos que estarão presentes no problema. É feita a ligação e a modelagem do problema através de relacionamentos com outras classes (Diagrama de Classe). As colaborações entre classes são mostradas no diagrama de classe para desenvolver os casos de uso modelados, e são criadas através de modelos dinâmicos em UML. Na análise, só é modelada classes que pertencem ao domínio principal do problema aplicado (classes que gerenciem banco de dados, *interface*, comunicação, concorrência e outros não estarão presentes neste diagrama).

1.4.2.3. Design (Projeto)

O design gera o detalhamento das especificações para a fase de programação do sistema. Nesta fase, novas classes serão adicionadas para prover uma infra-estrutura técnica: a *interface* do usuário e de periféricos, gerenciamento de banco de dados, comunicação com outros sistemas, entre outros. As classes do domínio do problema modeladas no diagrama de classe são mescladas tornando possível alterar tanto o domínio do problema quanto a infra-estrutura.

1.4.2.4. Programação

É na programação que as classes do *design* são convertidas para o código da linguagem orientada a objetos escolhida. Dependendo da capacidade da linguagem usada, essa conversão pode ser uma tarefa fácil ou muito complicada. Na programação os modelos criados são convertidos em código.

1.4.2.5. Testes

A fase de testes é formada por testes de unidade (classes individuais ou grupos de classes e são testados pelo programador), testes de integração (uso das classes e componentes integrados que verifica se as classes estão cooperando uma com as outras como descrito nos modelos) e testes de aceitação (observa-se o sistema e verificam se este está funcionando como o especificado nos primeiros diagramas de caso de uso)

2. LINGUAGENS E FERRAMENTAS

2.1. Rational Software Modeler 7.0

O Rational® Software Modeler (RSM) é uma ferramenta CASE comercial que nos permite criar diagramas e perfis UML. Atualmente o RSM faz parte do Rational® Software Architect que está sobre licença da IBM.

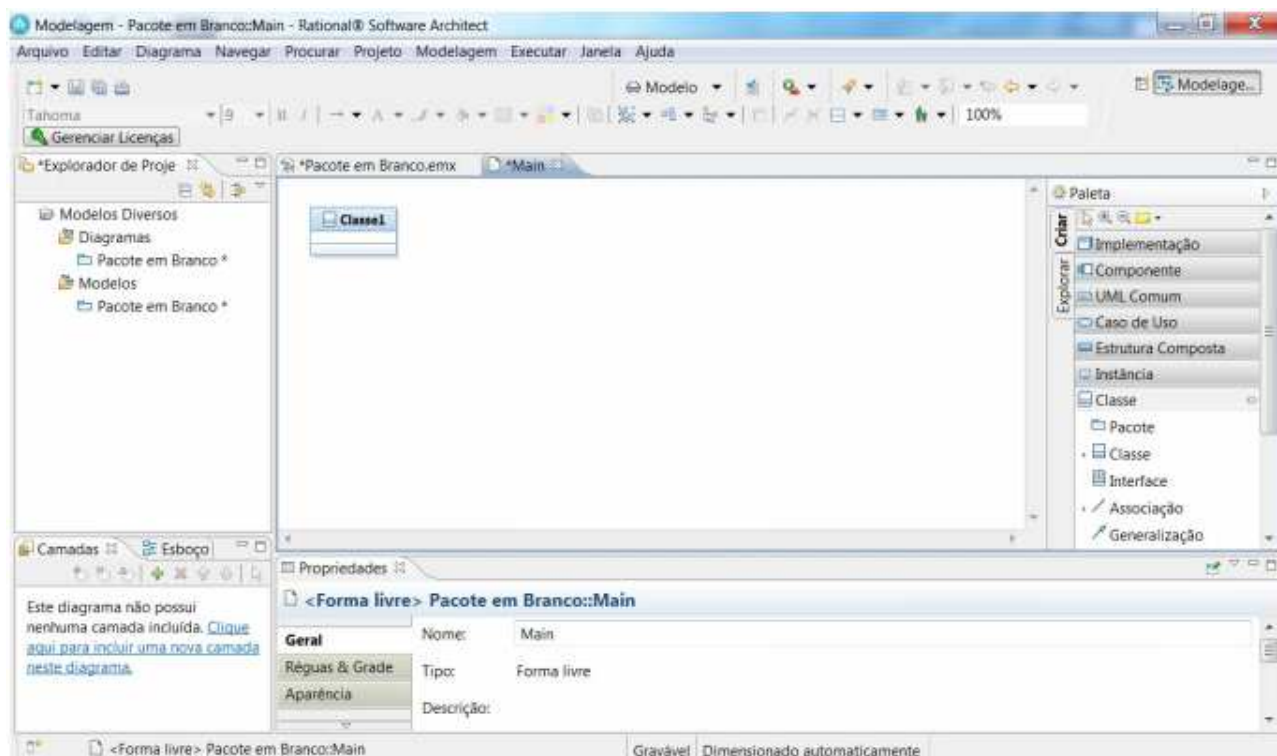


Figura 2: Tela do *Rational Software Modeler*

Fonte: *Rational_Software_Modeler_Tutorial*, 2012.

O *Rational Software Modeler*, é uma ferramenta de fácil utilização e sobretudo permite que se possa mostrar para o usuário uma prévia visualização do sistema além de uma vasta documentação que se pode fazer para cada detalhe da modelagem. E utilizando-se dessa ferramenta pode-se implementar em uma das várias linguagens que o software possui.

2.2. HTML

Segundo JORGE (2004), HTML é uma linguagem especializada, dedicada a construção e exibição de páginas *Web*. Consiste exclusivamente de linhas de programas em forma de texto comum e de códigos especiais (comandos) conhecidos como *tags* ou marcas. A linguagem HTML foi criada para simplificar a elaboração de textos mesclados com figuras, sons, animações e ligações (*links*) entre diversos textos.

Segundo CASTRO (2005), HTML significa '*HyperText Markup Language*', e é uma linguagem universal destinada à elaboração de páginas com hiper-texto (certos itens de um documento contêm uma ligação a outra zona do mesmo documento ou, como é mais vulgar, a outros documentos). A principal aplicação do HTML é a criação de páginas na *Web*, e convém esclarecer que não se trata de uma linguagem de programação. O HTML é uma espécie de linguagem de formatação, um arquivo de texto que é formatado através de uma série de comandos – definidos por '*tags*'. Podemos dizer no entanto, que o HTML é a linguagem usada pelos navegadores para mostrar as páginas *Web* ao usuário.

2.2.1. CSS

CSS, é a abreviação para o termo em inglês *Cascading Style Sheet*, traduzido para o português como folhas de estilo em cascata. Folha de estilo em cascata é um mecanismo simples para adicionar estilos (por exemplo: fontes, cores, espaçamentos) aos documentos web.

As CSS têm por finalidade devolver à marcação HTML/XML o propósito inicial da linguagem. A HTML foi criada para ser uma linguagem exclusivamente de marcação e estruturação de conteúdos. Isso significa que, segundo seus idealizadores, não cabe à HTML fornecer informações ao agente do usuário sobre a apresentação dos elementos. Por exemplo: cores de fontes, tamanhos de textos, posicionamentos e todo o aspecto visual de um documento não devem ser funções da HTML. Cabem às CSS todas as funções de apresentação de um documento, e essa é sua finalidade maior.

2.3. PHP

PHP, que significa "PHP: Hypertext Preprocessor", é uma linguagem de programação (linguagem script *Open Source* de uso geral) 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, Java e Perl, e é fácil de aprender. O objetivo principal da linguagem é permitir a desenvolvedores escrever páginas que serão geradas dinamicamente rapidamente. A Grande vantagem de se utilizar o PHP, é que o código é executado no servidor.

Neste trabalho, o PHP será o responsável pela comunicação da pagina WEB com os armazenados no banco de dados.

2.4. Dreamweaver CS5

O *Dreamweaver*, é uma ferramenta de criação e edição de HTML, que oferece diversos recursos ao seu usuario, como criar tabelas, editar quadros, criar paginas estaticas e dinamicos. O *Dreamweaver* CS5 se destaca dos demais editores por sua versatilidade e facilidade de uso, porém é de extrema importância o conhecimento de HTML para o melhor aproveitamento desta ferramenta.

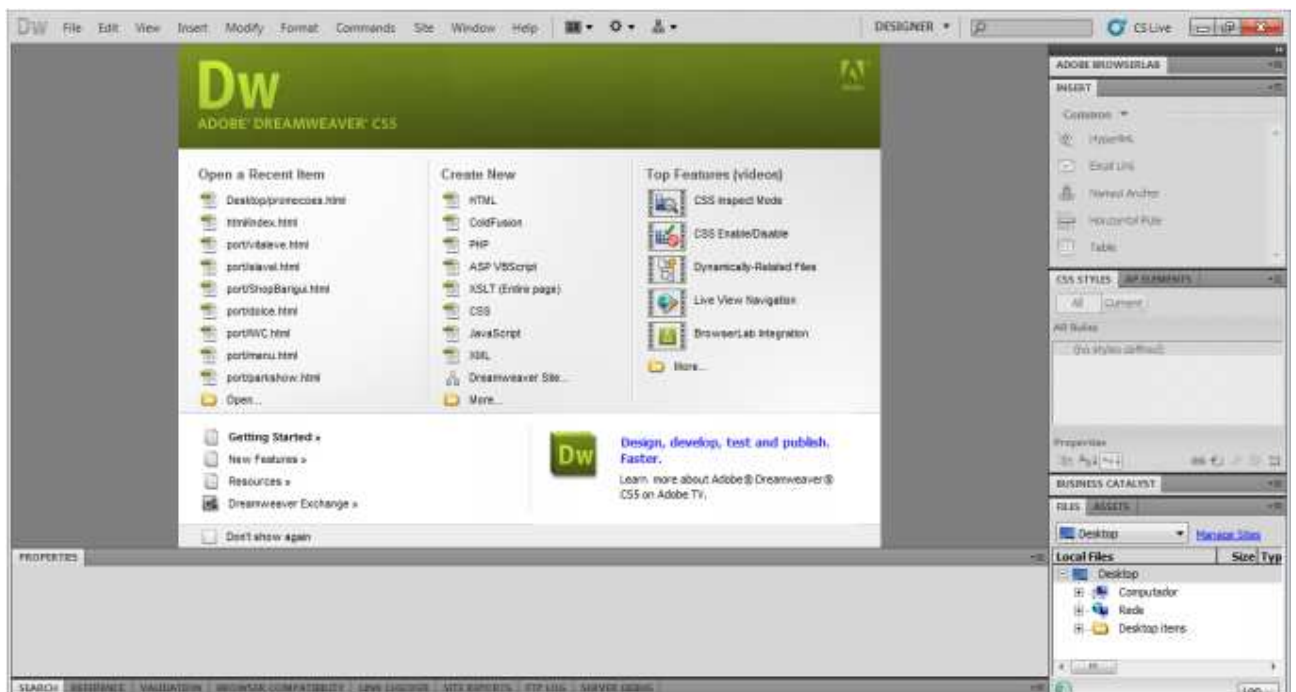


Figura 2: Tela do *Dreamweaver*

Fonte:

2.5. MySQL

O MySQL é um sistema de gerenciamento de banco de dados (SGBD), que utiliza a linguagem *SQL* (*Structured Query Language* - Linguagem de Consulta Estruturada) como interface. É atualmente um dos bancos de dados mais populares, com mais de quatro milhões de instalações pelo mundo.

Características do MySQL:

- Portabilidade (suporta praticamente qualquer plataforma atual);
- Compatibilidade (existem drivers *ODBC*, *JDBC* e *.NET* e módulos de interface para diversas linguagens de programação);
- Excelente desempenho e estabilidade;
- Pouco exigente em termos de recursos de hardware;
- Facilidade de uso;
- É um Software Livre;

A grande vantagem é a de ter código aberto e funcionar em, quase, qualquer plataforma e sistema operacional : *Windows*, *Linux*, *FreeBSD*, *BSDI*, *Solaris*, *Mac OS X*, *SunOS*, *SGI*, etc. É reconhecido pelo seu desempenho e robustez e também por ser multi-tarefa e multi-usuário. É frequentemente considerado um sistema mais "leve" e para aplicações menos exigentes, sendo preterido por outros am-se balanceadas ou equilibradas (JARDIM, 1976).

III - MODELAGEM DO SISTEMA

1. Diagramas

Segundo BOOCH (2000), a UML está baseada em diagramas que são representações gráficas de elementos de um sistema. São eles:

1. Diagrama de *use-case* (caso de uso);
2. Diagrama de classes;
3. Diagrama de Sequência;
4. Diagrama de Componentes.

1.1. Diagrama de Caso de Uso

Segundo Conallen (2003) um diagrama de caso de uso expressa os casos de uso do sistema em relação aos atores que os chamam.

Segundo Booch, Rumbaugh e Jacobson (2000), os diagramas de casos de uso são importantes para a organização e modelagem dos comportamentos de um sistema precisamente para visualizar, especificar e documentar o comportamento de um elemento.

1.1.1 Caso de Uso

O caso de uso é uma maneira formal de capturar e expressar a interação e o diálogo entre usuário do sistema, denominados atores e o próprio sistema (CONALLEN, 2003).

Para Guedes (2004), “os casos de uso referem-se aos serviços, tarefas ou funções que podem ser utilizados de alguma maneira pelos usuários do sistema”.

Segundo Larman (2000), um caso de uso é uma descrição de um processo relativamente grande, ou seja, não é normalmente um passo ou atividade individual em um processo.

Os casos de uso têm por objetivo descrever os requisitos funcionais do sistema, fornecer a descrição das funcionalidades do sistema, ou seja, deve expressar o que o sistema deve fazer, sem impor restrições de como isso será feito.

O caso de uso é representado por meio de uma elipse que contém um nome. Este nome deve dizer, de forma intuitiva, a finalidade específica ou funcionalidade deste caso (CONALLEN, 2003).

1.1.2 Ator

Os atores representam os papéis dos possíveis usuários que possam interagir com o sistema. Podem representar também, funcionalidades, módulos e entidades externos ao sistema (GUEDES, 2004).

O ator é representado por uma figura com formato de uma pessoa e um nome que ir tipo de papel que esse ator desempenha no sistema.

1.1.3 Relacionamento

Descreve a maneira com que o ator interage com o caso de uso e também a forma com que os casos de uso interagem entre si. Existem dois tipos principais de relacionamento. <<extend>> indica uma funcionalidade facultativa na execução de um determinado caso de uso. <<include>> indica a obrigatoriedade de execução de uma funcionalidade pertencente a um determinado caso de uso (GUEDES, 2004).

Os dois tipos de relacionamentos são representados por meio de uma linha tracejada com uma ponta em forma de seta.



Figura 4: Diagrama de Caso de Uso do Sistema Proposto.

1.2. Diagrama de Classe

Segundo Guedes (2004), o diagrama de classes é o mais usado e um dos mais importantes diagramas da UML. No diagrama de classes define-se a estrutura das classes utilizadas pelo sistema, determinando os atributos e os métodos de cada classe, além de estabelecer os relacionamentos e como as classes trocam informações entre si. O diagrama de classes serve também de base para a construção de outros diagramas da UML (GUEDES, 2004).

Pelo diagrama de classes, também é possível visualizar vários componentes da programação orientada a objetos, como polimorfismo e herança.

1.2.1 Classe

Segundo Booch, Rumbaugh e Jacobson (2000), “classe é a descrição de um conjunto de objetos que compartilham os mesmos atributos, operações, relacionamentos e semântica”.

Segundo Guedes (2004), uma classe é composta por:

- * Atributos: definem as propriedades e particularidades de cada classe, variando de objeto para objeto;

* Métodos: dizem respeito ao comportamento das classes. Cada método é uma funcionalidade que a classe pode executar. Os métodos podem ser definidos de maneira a receber ou não parâmetros e retornar ou não valores.

Além de atributos e métodos, a classe deve conter também um nome. Esse nome deve ser o mais adequado possível para a representação dessa classe, assim torna-se mais compreensão do diagrama de classes e também a identificação da classe no sistema.

1.2.2 Relacionamento

Os relacionamentos indicam o vínculo que ocorre entre duas ou mais classes. Estes relacionamentos podem indicar associação, composição e/ou agregação entre as classes.

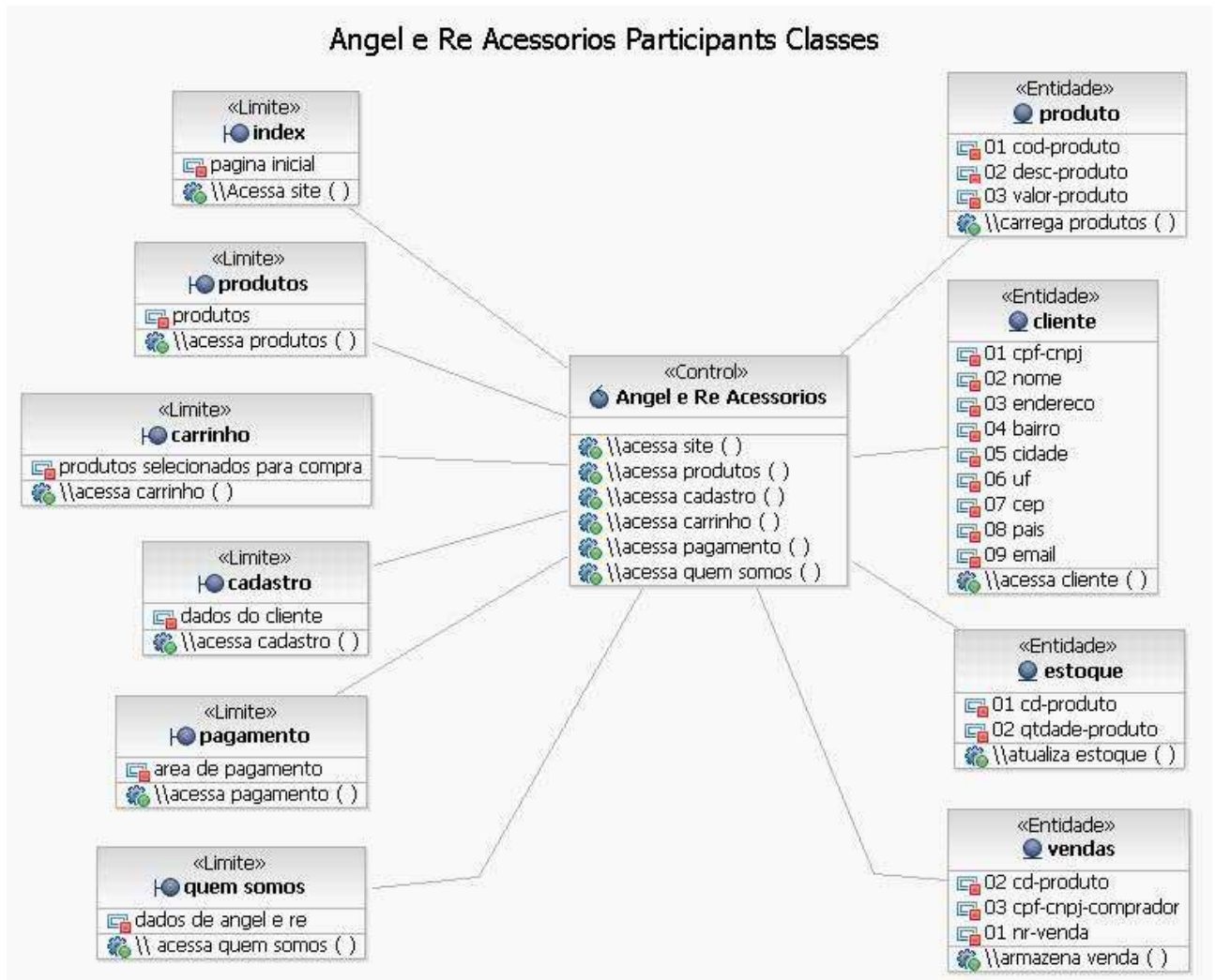


Figura 5: Diagrama de Classe do Sistema Proposto

1.3. Diagrama de Seqüência

Um diagrama de seqüência descreve o comportamento dos objetos do sistema. Esses objetos se relacionam entre si por meio de troca de mensagens em interações feitas de forma ordenada em uma linha de tempo que ocorre de cima para baixo.

Em geral, um diagrama de seqüência se baseia na definição de um caso de uso e apoia-se no diagrama de classes para determinar os objetos e sua iteração que serão envolvidas por um processo. Assim, o diagrama de seqüência se preocupa com a ordem temporal com que as mensagens são trocadas entre os objetos (GUEDES, 2004)..

Os principais componentes do diagrama de seqüência são os atores e as classes. Cada classe possui sua linha de vida. Esta linha de vida determina o tempo de existência de um objeto durante a execução de um determinado processo. Caso ocorra a destruição de um objeto, esta linha é finalizada com um “X” (GUEDES, 2004).

O foco de controle ou ativação determina o período em que o objeto participa de determinada ação no processo. Indica o momento em que determinado método de um objeto será executado. Os focos de controle são representados dentro da linha da vida de cada objeto (GUEDES, 2004).

As mensagens dentro de um diagrama de seqüência indicam a ocorrência de eventos dentro da execução dos processos, geralmente determinam à chamada de algum método de um determinado objeto participante do processo. No entanto, podem também ocorrer mensagens que indicam apenas a comunicação entre atores, neste caso, sem disparar nenhum método.

- usuario/usuario
- index/limite/usuario
- angel e re acessorios/controle/angel e re acessorios
- produtos/limite/produtos
- produtos/limite/produtos
- produto/limite/produto
- cadastro/limite/cadastro
- cliente/limite/cliente
- carrito/limite/carrito
- pagamento/limite/pagamento
- verais/limite/verais
- estoque/limite/estoque
- quem somos/limite/quem somos

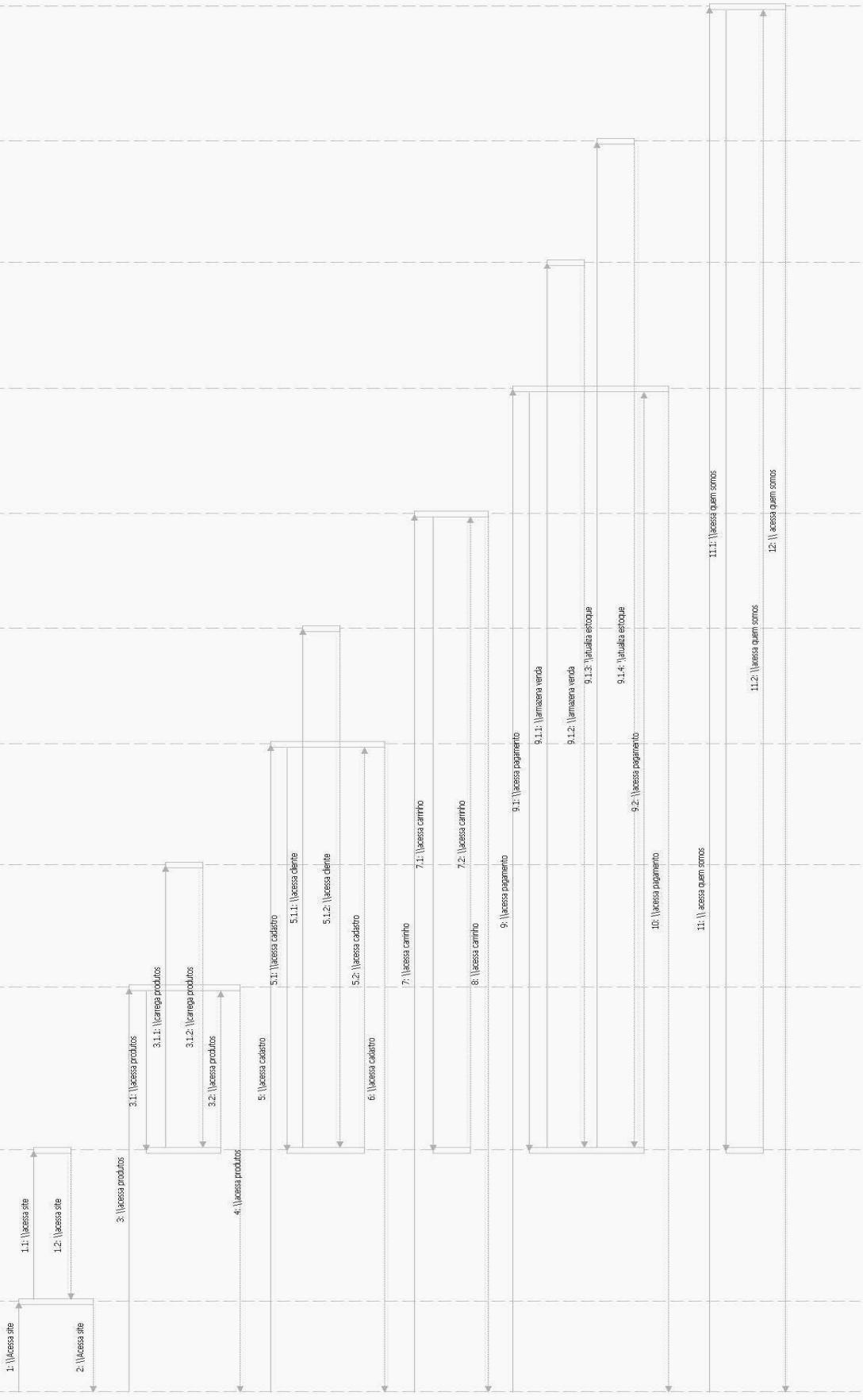


Figura 6: Diagrama de Sequência do Sistema Proposto

1.4. Diagrama de Componentes

O diagrama de componentes é utilizado na modelagem dos aspectos físicos do sistema demonstrando a configuração do sistema, ou seja, como as partes do sistema se relacionam.

Segundo Booch, Rumbaugh e Jacobson (2000), o diagrama de componentes é essencialmente diagrama de classes que focaliza os componentes de um sistema.

Cada parte do sistema é identificada por um componente, e a organização, os relacionamentos e a dependência entre esses componentes formam o sistema.

Um diagrama de componentes modela os itens físicos do sistema, proporcionando a equipe responsável pelo desenvolvimento do sistema identificar e relacionar quais serão os arquivos executáveis que serão gerados, quais são as bibliotecas de sistema que já existem e quais serão criadas e quais os arquivos de configuração do sistema serão gerados.

Angel e Re Acessorios Diagrama de Componentes

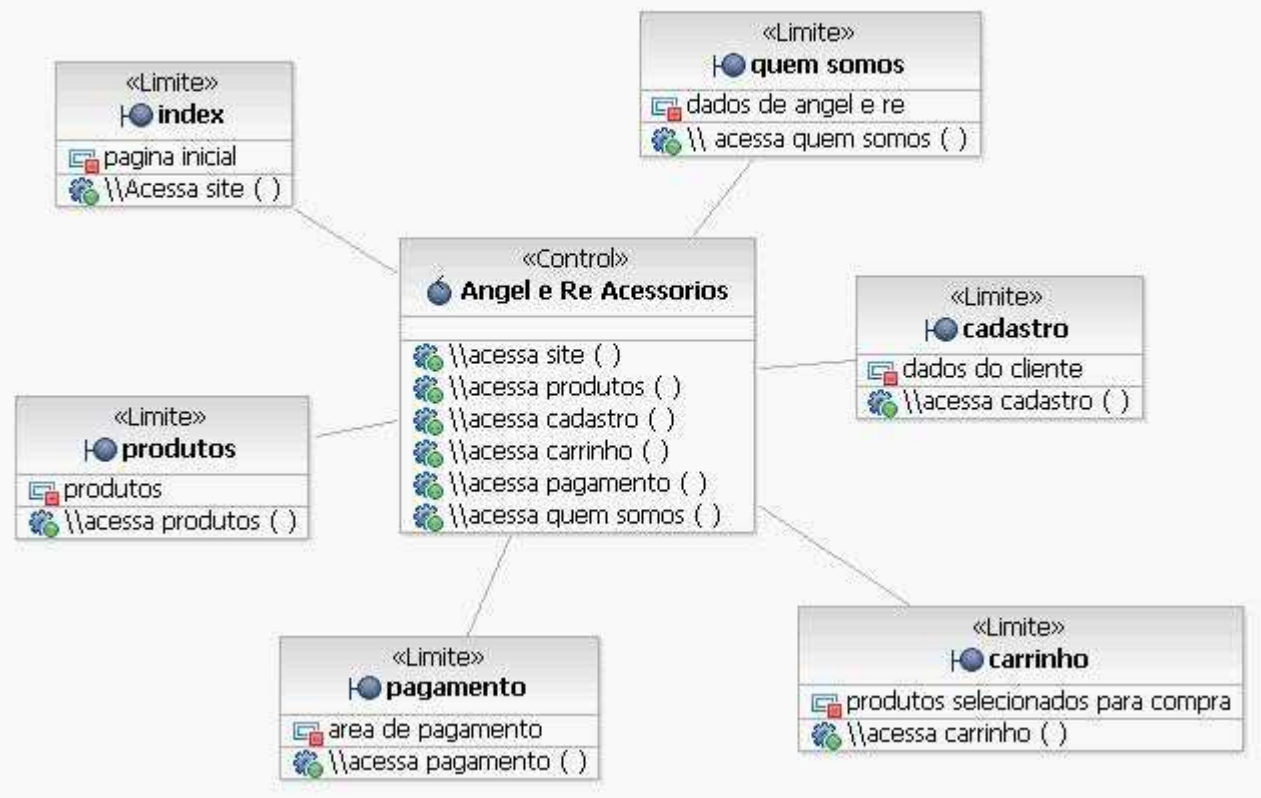


Figura 7: Diagrama de Componentes do Sistema Proposto.

1.5. Diagrama de Atividades

O diagrama de atividades é um diagrama UML utilizado para modelar o aspecto comportamental de processos. É um dos diagramas que mais sofreu mudanças em seu meta-modelo, desde seu surgimento no UML 1.0. Neste diagrama, uma atividade é modelada como uma sequência estruturada de ações, controladas potencialmente por nós de decisão e sincronismo. Em seu aspecto mais simples, um diagrama de atividades pode ser confundido com um fluxograma. Entretanto, ao contrário de fluxogramas, os diagramas de atividades UML suportam diversos outros recursos, tais como as partições e os nós do tipo fork e merge, além da definição de regiões de interrupção, que permitem uma modelagem bem mais rica do que simplesmente um fluxograma.



Figura 8: Diagrama de Atividades do Sistema Proposto.

V - CONSIDERAÇÕES FINAIS

Inúmeros têm sido os estudos que buscam entender o comércio eletrônico e seus impactos no ambiente empresarial. Durante o desenvolvimento deste trabalho foi possível estudar e praticar os conceitos de desenvolvimento de software, iniciados pela análise de requisitos, passando pelas fases intermediárias até a finalização a frente da sua implantação.

A qualidade aliada a competitividade nas vendas pela internet pode ser um grande diferencial. As empresas que atuam na área de comércio eletrônico devem adotar estratégias gerenciais que busquem aumentar a percepção do cliente em relação ao controle da compra, resultando assim em melhores índices de satisfação.

Com o objetivo de compreender o comércio eletrônico e conhecer os *déficits* na qualidade de compras pela internet verifica-se que grande parte dos problemas que impedem o crescimento do *e-commerce* encontra-se na falta de estratégias e planejamento das empresas que se lançam na rede sem entender o que realmente há de importante por trás deste conceito.

E necessário entender que os fatores que irão satisfazer o cliente estão interligados em grau de confiabilidade na rede, o respeito, a privacidade, a segurança da informação e a qualidade no atendimento, portanto deve igualar o tratamento a esses aspectos.

Os resultados obtidos foram muito satisfatórios. Superando dificuldades, aprendendo novas tecnologias, agregado os conhecimentos, embaçados pelos professores durante o decorrer do curso de graduação.

VI - REFERÊNCIAS

BATISTA, Emerson de O. **Sistemas de Informação: o uso consciente da tecnologia para o gerenciamento**. São Paulo: Saraiva, 2004.

BOOCH, Grady; RUMBAUGH, James; JACOBSON, Ivar. **UML, Guia do Usuário**. Rio de Janeiro. Editora Campus, 2000.

CASTRO, Maria Alice Soares de. São Carlos – SP Disponível em:
<<http://www.icmc.usp.br/ensino/material/html/intro.html>>. Acesso em: 18/05/2005.

COELHO, Idemir Dias. **Guia de Consulta Rápida JavaServer Pages**. São Paulo. Novatec Editora Ltda., 2000.

CONALLEN, Jim. **Desenvolvimento de aplicativos WEB COM UML**. Rio de Janeiro: Campus, 2003.

CHURCHILL, Gilbert A. **Marketing: criando valor para os clientes**. 2ª Edição. São Paulo : Saraiva, 2003.

DATE, C. J. **Introdução a sistemas de banco de dados**. Rio de Janeiro: Editora Campus, 1984.

FOWLER, Martin; Scott, Kendall. **UML Essencial: Um breve Guia para a Linguagem-Padrão de Modelagem de Objetos**. Porto Alegre: Boohman, 2000.

FURLAN, J. D. **Modelagem de Objetos Através da UML**. São Paulo: Editora Makron Books, 1998.

GOMES, Luiz Flávio Autran Monteiro et al. **Tomada de Decisão Gerencial: enfoque multicritério**. São Paulo: Atlas S.A., 2002.

HEUSER, Carlos Alberto. **Projeto de Banco de Dados**. 3 ed. São Paulo: Editora Sagra Luzzatto, 2000.

JORGE, Marcos. **Java – (Série passo a passo lite)**. São Paulo. Pearson Education do Brasil, 2004.

KALAKOTA, Ravi; WHINSTON. Andrew B. **Frontiers of Electronic Commerce**. New York: Addison-Wesley, 1996.

KENN, Peter G. W. **Guia Gerencial para a Tecnologia da Informação: conceitos essenciais e terminologia para empresas e gerentes**. Rio de Janeiro: Campus, 1996.

KORTH, Henry F., Abraham Silberschatz. Sistema de Banco de Dados. Makro Books; 1995;

KOTLER, PHILIP (2000). Administração de Marketing. 10a Edição. São Paulo, Prentice Hall.

LAKATOS, E M.; MARCONI, M. A.. Fundamentos da metodologia científica. 6. ed. São Paulo: Atlas, 2005.

LEI DO SOFTWARE Lei nº 9.609, de 19 de fevereiro de 1998. Brasília. Disponível em: <<http://www.edutec.net/Leis/Gerais/software.htm>>. Acesso em: 01/05/2005.

MASLOW, A. H.. Motivation and Personality. New York: Harper, 1954. O`REILLY, Tim . What is Web 2.0: Design Patterns and Business Models for the Next Generation of Software. Disponível através do link: (<http://www.oreillynet.com/pub/a/oreilly/tim/news/2005/09/30/what-is-web>). Acessado em 27/10/2012.

PEARSON, Addison Wesley. **Sistema de Banco de Dados.** 4ª Edição, São Paulo: Pearson Education, 2005. 806 p.

STAIR, Ralph M.; REYNOLDS, George W. **Princípios de Sistemas de Informações.** 4 ed. Rio de Janeiro: LTC, 2002. 496 p.

HORTINHA, Joaquim. **X-Marketing.** Edições Sílabo. 1a edição. Lisboa, 2002.

