

**CENTRO UNIVERSITÁRIO FACVEST
CURSO DE CIÊNCIA DA COMPUTAÇÃO
TRABALHO DE CONCLUSÃO DE CURSO**

SISTEMA DE GERENCIAMENTO DE ESTOQUE WEB

Área: Aplicação Web

Leonardo Filipe Rigon

Lages (SC),
Novembro de 2012.

**CENTRO UNIVERSITÁRIO FACVEST
CURSO DE CIÊNCIA DA COMPUTAÇÃO
TRABALHO DE CONCLUSÃO DE CURSO**

SISTEMA DE GERENCIAMENTO DE ESTOQUE WEB

Área: Aplicação Web

Leonardo Filipe Rigon

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

Lages (SC), novembro de 2012.

EQUIPE TÉCNICA

Acadêmico

Leonardo Filipe Rigon

Professor Orientador

Profº. Márcio José Sembay, Msc.

Coordenador de TCC

Profº. Márcio José Sembay, Msc.

Coordenador do Curso

Profº. Márcio José Sembay, Msc.

SUMÁRIO

EQUIPE TÉCNICA	03
RESUMO.....	06
ABSTRACT	07
AGRADECIMENTOS	08
LISTA DE ABREVIATURAS.....	09
LISTA DE FIGURAS.....	10
I. INTRODUÇÃO	11
1.1 Justificativa	11
1.2 Importância.....	11
1.3 Objetivo do Trabalho	12
1.3.1 Objetivo Geral	12
1.3.2 Objetivos Específicos.....	12
1.4 METODOLOGIA.....	12
1.4.1 Estudo de caso.....	12
1.4.2 Cronograma	13
1.5 Estrutura do Trabalho	13
II. FUNDAMENTAÇÃO TEÓRICA.....	15
2.1 Aplicação Web	15
2.2 Automação Comercial Web.....	16
2.3 Protocolo HTTP.....	16
2.4 Linguagem de marcação de texto HTML	17
2.5 Linguagem PHP.....	17
2.5.1 História	18
2.5.2 Características	19
2.5.3 Sessões.....	19
2.5.3.1 Sessão	20
2.5.4 Segurança	20
2.5.5 Cookies.....	20
2.5.6 Biblioteca FPDF.....	21
2.6 Banco de Dados MySQL	22
2.6.1 Transações.....	22
2.7 phpMyAdmin	23
2.8 Ajax.....	23
2.8.1 Java Script.....	25
2.8.2 XML.....	25
2.9 Servidor Web	25
2.9.1 Servidor Apache	26
2.9.2 Xampp	27
2.9.3 Wamp.....	27
III PROJETO.....	28
3.1 Análise de Requisitos.....	28
3.2 Requisitos não Funcionais	28
3.3 Regras de Negócio.....	28
3.4 Problemas a Serem Sanados.....	29
3.5 Melhorias	29
IV FUNDAMENTAÇÃO TEÓRICA	30

4.1 Dreamweaver	30
4.2 Unified Modeling Language – UML	30
4.2.1 Diagrama Caso de Uso	31
4.2.2 Diagrama de Classe	31
V MODELAGEM DO SISTEMA.....	32
5.1 Diagrama de Caso de Uso da Aplicação	32
5.2 Arquitetura Geral da Aplicação.....	33
5.3 Diagrama de Entidade Relacionamento da Aplicação.....	33
5.4 Diagrama de Classe da Aplicação	34
VI CONCLUSÃO	40
REFERÊNCIAS BIBLIOGRÁFICAS	41

RESUMO

Com o aumento significativo do uso de sistemas que possuem seu banco de dados acessados remotamente, surgiu a possibilidade de diminuir custos e dar maior segurança aos dados de usuário, seja ele doméstico ou empresarial. Com novas tecnologias e avanços na área da internet, não é mais necessário manter um servidor de banco de dados em uma empresa. O projeto proposto apresenta uma aplicação Web, que é um sistema de controle de estoque inteiramente ambientado na internet, desenvolvido na linguagem PHP vinculada ao banco de dados MySQL, sem o uso de *softwares* instalados no computador, apenas utilizando o mesmo software de navegação de internet, que acompanha o sistema operacional, apenas com alguma atualização.

Palavras chaves: banco de dados, PHP, MySQL.

ABSTRACT

With the significant increase in the use of systems that have accessed your database remotely, it has become possible to reduce costs and provide greater security to user data, be it home or business. With new technologies and advances in the internet, it is no longer necessary to maintain a database server in a company. The proposed project presents a web application, which is a system of inventory control fully acclimated on the Internet, developed in PHP linked to MySQL database without the use of software installed on your computer, just using the same software for internet browsing that accompanies the operating system, just with an update.

Keywords: Database, PHP, MySQL.

AGRADECIMENTOS

Agradeço primeiramente meus pais, por terem me dado todo o suporte necessário para realizar meus sonhos e poder estar concluindo o ensino superior;

Agradeço também meus professores pelo conhecimento passado durante esses anos de faculdade;

Agradeço a meus colegas pelos momentos de companheirismo e de mútua ajuda, além de meus amigos e meus companheiros de trabalho, que sempre me apoiaram.

LISTA DE ABREVIATURAS

ASP - *Active Server Pages*

CGI - *Common Gateway Interface*

FI - *Form Interpreters*

HTTP - *Hypertext Transfer Protocol*

HTTPS - *Hypertext Transfer Protocol Secure*

IP - *Internet Protocol*

PHP - *PHP Hypertext Preprocessor*

SQL - *Structured Query Language*

SGDB - *Sistema gerenciador de banco de dados*

TCC - *Trabalho de Conclusão de Curso*

UML - *Unified Modeling Language*

URL - *Uniform Resource Locator*

W3C - *World Wide Web Consortium*

LISTA DE FIGURAS

Figura 01. Etapas do desenvolvimento do TCC	14
Figura 02. Comparativo entre as arquiteturas clássica e AJAX	25
Figura 03. Interface do Wamp	28
Figura 04. Diagrama caso de uso da aplicação.....	32
Figura 05. Diagrama da arquitetura geral da aplicação	33
Figura 06. Diagrama entidade relacionamento da aplicação	34
Figura 07. Diagrama de classe da aplicação.....	34
Figura 08. Interface de login para acesso ao sistema	35
Figura 09. Tela inicial do sistema.....	35
Figura 10. Tela de cadastro de lembrete.....	36
Figura 11. Tela de cadastro de clientes.....	36
Figura 12. Tela de pesquisa de clientes	36
Figura 13. Tela de resultado de pesquisa de cliente	37
Figura 14. Tela de listagem de clientes	37
Figura 15. Tela de cadastro de produtos.....	37
Figura 16. Tela de pesquisa de produtos	38
Figura 17. Tela de resultados da pesquisa de produtos	38
Figura 18. Tela de listagem de produtos.....	38
Figura 19. Tela de alteração de cadastro de produtos.....	38
Figura 20. Tela de listagem de produtos com alteração de estoque	39

INTRODUÇÃO

Sistemas ambientados totalmente na Web (rodando em um navegador) se tornaram comuns em nosso cotidiano. Alguns exemplos são: Editores de Texto, Editores de Imagens, Editores de Vídeo, Jogos, *Softwares* de Mensagens Instantâneas entre outros.

Praticamente para todo tipo de *software*, existe um similar via Web, rodando no navegador, ou que necessite de uma pequena atualização do navegador para funcionar (*plugin*).

Controles de estoque são talvez os mais comuns tipos de *softwares* existentes no mercado, salvo algumas poucas exceções, são exatamente os mesmos com pequenas modificações, alguns com mais recursos outros com menos. Ou mesmo estão embutidos em outros softwares maiores.

Muitos deles são os “*Softwares* de Prateleira”, que são usados normalmente por empresas de pequeno porte. Porém, ainda existem os feitos sob medida que se aplicam em empresas maiores e são de custo maior.

Por serem *softwares* comerciais, sua facilidade e eficiência, além do custo ser baixo, tem que ser extremamente bons, para que o empresário consiga competir no mercado, visto que é um dos mais concorridos e complexos para se trabalhar.

1.1 Justificativa

Aplicações Web são muito comuns nos dias de hoje, com o avanço da tecnologia e um maior acesso a internet por parte de todo o tipo de usuário, novas formas para se trabalhar foram criadas, tudo para fazer mais fácil a vida do usuário seja ele doméstico ou comercial.

1.2 Importância

Com crescente aumento de usuários de computadores, foram necessárias novas formas de uso para *softwares*, tudo para facilitar o trabalho do usuário e sua rapidez e precisão na obtenção de resultados.

Sistemas para gerenciamento e controle de estoque são muito comuns no mercado. O

modelo padrão é composto por bancos de dados integrados que fazem o controle do estoque de modo geral, além de rotinas para entrada e saída de itens.

Vários tipos existem de ferramentas para controle de estoque existem, porém todas com o mesmo fim, só variando seus recursos externos.

1.3 Objetivo do Trabalho

1.3.1 Objetivo Geral

Este projeto tem como objetivo o desenvolvimento de uma Aplicação Web para gerenciamento de estoque, usando uma interface simples, para que otimize o desempenho do sistema e o uso do operador.

1.3.2 Objetivos Específicos

Os objetivos específicos deste trabalho consistem em:

- ✓ Facilidade na operação da Aplicação Web;
- ✓ Rapidez nas funções do sistema;
- ✓ Geração de Relatórios de dados de clientes;
- ✓ Geração de Relatórios de entrada e saída de mercadorias;
- ✓ Controle sobre entrada e saída de mercadorias;
- ✓ Cadastro e visualização de lembretes;
- ✓ Segurança na Aplicação web.

1.4 METODOLOGIA

1.4.1 Estudo de Caso

Para dar-se início ao estudo, foi feito um levantamento de dados com um representante comercial de uma empresa de pequeno porte, para que o mesmo pudesse informar o que seria necessário em um *software* que pudesse lhe auxiliar em suas vendas, tornando seu serviço mais efetivo e satisfatório.

Sendo assim o trabalho foi dividido da seguinte forma:

a) Levantamento dos dados necessários para que o sistema seja útil à uma empresa pequena, ou um profissional autônomo vendedor de manufaturas;

- b) Informações de como é quais as ações que o sistema deve realizadas;
- c) Levantamento do que pode ser feito a fim de facilitar o trabalho na empresa.

1.4.2 Cronograma

A metodologia apresentada anteriormente mostrou os passos para o desenvolvimento do TCC. Em primeiro plano, ocorrendo nos quatro primeiro meses do segundo semestre iniciou-se as pesquisas de materiais para realizar a fundamentação teórica do trabalho.

Nos dois primeiros meses, juntamente com as pesquisas, elaborou-se a entrega da pré-proposta para a coordenação do TCC, já no segundo mês iniciaram-se as atividades para a confecção da revisão bibliográfica, esta foi a principal etapa do trabalho, onde foi dedicada uma reserva maior de tempo para sua conclusão.

As etapas do cronograma estão resumidas na Tabela 1.

ATIVIDADES	JUN	JUL	AGO	SET	OUT	NOV	DEZ
Pesquisa	X	X	X	X	X	X	
Confecção e apresentação da pré-proposta			X	X			
Elaboração da revisão bibliográfica			X	X	X		
Apresentação dos relatórios quinzenais			X	X	X	X	
Pré-modelagem do software					X	X	
Entrega e defesa do TCC à banca avaliadora							X

Tabela 1 – Cronograma do Projeto Final I

1.5 Estrutura do Trabalho

A figura 1 demonstra todas as etapas necessárias para o desenvolvimento do TCC (Trabalho de Conclusão de Curso).

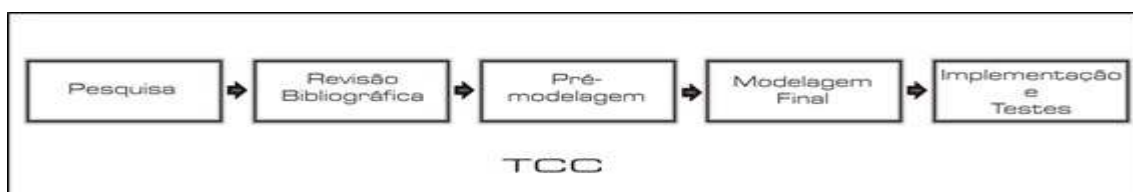


Figura 1. Etapas do desenvolvimento do TCC.

Fonte: o Autor (2012)

Para concluir o objetivo do trabalho, muitos dados foram levantados, para que daí pudesse se iniciar o trabalho, posteriormente para fundamentar a teoria, foi feita uma revisão bibliográfica, a fim de dar embasamento técnico ao trabalho, justificando assim os tópicos abordados para o desenvolvimento do TCC. Os dados foram pesquisados em livros, páginas da Web, artigos, revistas, trabalhos acadêmicos, entre outras modalidades. A última etapa contou com a implementação do sistema, desenvolvida com auxílio de várias ferramentas para programação web, utilizando por fim banco de dados MySQL, testes necessários, anotações recorrentes e conclusões.

Hardware utilizado:

- Um *notebook* para implementação e documentação;
- Processador Intel Core I3 2.27Mhz;
- 3GB memória RAM ;
- HD 400GB.

I FUNDAMENTAÇÃO TEÓRICA

Este capítulo aborda os conceitos que envolvem o contexto deste TCC fazendo uma revisão bibliográfica sobre o tema para que após a aquisição do conhecimento possa ser proposto um sistema aplicando tais fundamentos.

2.1 Aplicação WEB

O enorme crescimento da Internet e da Web como um todo, causou um grande impacto em nossas vidas, seja no meio pessoal ou profissional. A Web tornou-se então o fator central de várias aplicações nas mais diferentes áreas. Nos dias de hoje, diversos setores de negócios, realizam suas operações no ambiente da Internet, muitos sistemas de informação e de bancos de dados estão migrando para os ambientes de Internet. Uma vasta abrangência de novos e complexos sistemas comerciais e corporativos está emergindo no ambiente de Web (LI ET AL, 2009)

Para Chaubey e Suresh (2001), um dos principais fatores que contribui para essa migração, é que a Web permite principalmente uma diminuição de custos e novas possibilidades de criação de produtos, além de que acompanha o surgimento de novas tecnologias e ferramentas para desenvolvimento e emprego das aplicações, principalmente tecnologias de segurança que permitiram aplicações mais confiáveis.

Um aplicativo Web é um *software* projetado para ser executado diretamente do navegador de internet, sendo um conjunto de aplicações executados em um servidor de Internet.

A tendência é que com o avanço da internet, e o aumento significativo de sua capacidade de velocidade e armazenamento, que com o tempo, as aplicações web, substituam lentamente alguns dos *softwares* comuns que são usados hoje em dia. Atualmente já existem versões para Web de editores de texto, que com um pouco menos de recursos, conseguem fazer exatamente o que as versões “tradicionais” fazem, com a vantagem de já estarem integradas a serviços de e-mails ou outros tipos de serviços da web.

Além disso, os antigos sites sofreram uma grande evolução, tornaram se mais úteis e interativos. Páginas da Web que simplesmente exibiam informações para os visitantes tornaram-se sistemas interativos e altamente funcionais que permitem muitos e variados tipos de usuários interagirem com muitos e diferentes tipos de aplicações. (OFFUTT, 2002).

2.2 Automação Comercial para Web

Automação comercial é a forma de trabalhar com as mais diversas áreas de uma empresa, usando de equipamentos eletrônicos, *hardwares* e *softwares*, ao invés de processos manuais. Pode ser utilizada em todas as áreas de uma empresa, independentemente do seguimento, área ou serviço que a empresa atua. Hoje em dia é inviável uma empresa prosperar e competir no mercado sem ter o uso de sistemas de automação comercial, podendo ser desde simples aplicações para controle de estoque até sistemas gerenciais que trabalham e controlam todos os setores de uma empresa.

Para Ean (2005), uma empresa que faz uso de sistemas de automação comercial, terá mais lucros que uma empresa que não faz uso, conquistando novos clientes e tornando seu serviço mais efetivo.

Sistemas de automação comercial para Web são novos, se comparados com modelos de automação comercial clássicos. Essas aplicações Web podem fazer tudo o que uma aplicação para *desktop* faz, inclusive no armazenamento de informações e dados.

As Aplicações Web para Automação Comercial apresentam algumas vantagens sobre os sistemas de Automação Comercial para *desktop*, principalmente no que diz respeito ao manuseio das funções do sistema e dispensar o uso de instaladores, além da praticidade de poder ser acessada de qualquer computador ou *laptop*, desde que o mesmo tenha conexão com a internet, o que pode ser muito útil em determinados tipos de empresa, principalmente no que diz respeito a empresas de representações comerciais, e pequenas empreendimentos de manufaturas.

2.3 Protocolo HTTP

O protocolo HTTP (*Hypertext Transfer Protocol*), desde o início dos 90 é o protocolo mais utilizado na internet. Sua versão mais usada é a 1.0, que permite o uso de mais recursos, que sua versão anterior a 0.9. Seu objetivo é permitir a transferência de arquivos (no formato HTML) que são encontrados devido ao uso da URL entre o cliente e um servidor Web.

Para Gourley e Totty (2002) o *Hypertext Transfer Protocol* (HTTP) é uma metodologia de comunicação usada para troca de dados entre o navegador e um servidor web. Sendo utilizado quando, o usuário digita algo na barra de endereços de seu navegador de internet, podendo ser tanto um site ou uma aplicação, a ligação da página até a plataforma do

cliente e suas ligações com o servidor web são feitas por este protocolo. A comunicação do Protocolo HTTP entre cliente e servidor ocorre por meio de troca de mensagens. O cliente envia mensagens de requisição e o servidor retorna, enviando uma mensagem de resposta para o cliente, sendo essas mensagens compostas de corpo e cabeçalho. O corpo contém dados que serão tratados como texto, imagem ou arquivos pra download pelo HTML e o cabeçalho contém as informações de cliente ou de servidor.

Seus métodos para transferência de informações entre cliente e servidor, são utilizados como padrão por muitos códigos HTML e linguagens para envio de formulários ao servidor.

2.4 HTML

A linguagem HTML é usada para criação de páginas web, sendo lidas e interpretadas pelos navegadores gerando assim páginas estáticas. Foi criada por Tim Berns-Lee no começo dos anos 90 a fim de possibilitar a troca de informações entre seus amigos. Com a disseminação do padrão HTML em toda internet, em 1994 foi criado a *World Wide Web Consortium* (W3C), o consórcio internacional da internet, que tinha como objetivo a padronização e manutenção do HTML, entre outras linguagens criadas posteriormente (LOGMAN, 1998).

Os códigos HTML são construídos a partir de *tags*, onde os comandos ficam entre os símbolos de menor que “<” e maior que “>”. Arquivos com as extensões “.html” e “.htm” ficam armazenados no servidor Web, onde o navegador chama o conteúdo, fazendo assim o download do arquivo HTML, posteriormente mostrando e interpretando o arquivo HTML.

2.5 Linguagem PHP

O PHP (*PHP Hypertext Preprocessor*) é uma linguagem *open source* muito usada, principalmente para aplicações embutidas no HTML, sendo usada para criação de scripts executados no servidor web. O PHP por ser uma linguagem interpretada diretamente de um servidor, faz com que o usuário leia as *tags* em HTML, e não o código do PHP. A funcionalidade e capacidade da linguagem PHP é constantemente testada em sites com um número grande de acessos, podendo ser editado, usado e redistribuído contendo outros tipos de módulos não tendo restrições (MELO; NASCIMENTO, 2007).

As variáveis dos códigos são checadas de forma dinâmica pela linguagem, em outras palavras, pode se armazenar vários tipos de dados enquanto existe a variável, sem ter a necessidade de declarar qual é o tipo de sua criação.

2.5.1 História

O Php foi criado por Rasmus Lerdorf, em 1994, sendo um pacote CGI (*Common Gateway Interface*) para uma página que usava até então um conjunto de scripts Perl. Para Muto (2006), a primeira versão do PHP, que tinha o nome de PHP/FI, já tinha um interpretador chamado FI (*form Interpreter*) e algumas características comuns até os dias atuais, como suporte ao banco de dados MySQL, interpretação automática de variáveis do formulário e sintaxe embutida no HTML.

Depois de uma maior elaboração e da distribuição das primeiras versões da linguagem, o PHP ganhou o reconhecimento e apoio da comunidade livre, segundo Melo e Nascimento (2007 p. 21): “Assim como ocorreu com o sistema operacional *Linux*, diversos entusiastas do *software* livre tomaram conhecimento do pacote PHP/FI, os quais passaram a contribuir e acrescentar suas ideias ao mesmo”. O que tornou o PHP popular foi sua acessibilidade no código fonte, concedida pelo autor.

De acordo com MORAZ (2007), na metade dos anos 90, o PHP era utilizado em aproximadamente quinze mil sites distribuídos pelo mundo.

Segundo o PHP Group (2012), em sua versão três, o PHP já possibilitava a criação de códigos orientados a objetos e herança, suporte a grande parte dos bancos de dados existentes e a adaptação para novos tipos de protocolos de internet, sendo a primeira versão que possibilitava o uso de extensões, atraindo um número maior de programadores que desenvolveram muitos módulos. Em 1998, o até então PHP/FI foi renomeado para PHP, e no fim do mesmo ano já estava presente em cerca de 10% dos servidores web do mundo.

Em sua quarta versão o PHP foi melhorado para suprir várias necessidades e resolver alguns problemas que estavam presentes na versão anterior. A característica mais importante na quarta versão é a rapidez que foi alcançada graças a mudança de seu motor, fazendo com que o código, nessa versão, seja primeiramente compilado, e depois, em um segundo momento seja executado, diferente do que ocorria anteriormente onde o código era compilado e executado ao mesmo tempo. Em 2004 a versão Cinco foi lançada contendo várias funções novas, como um maior e mais eficaz gerenciamento da memória e a implementação de

ponteiros no código, além de uma melhoria no novo suporte ao banco de dados MySQL. Souza (2007) afirma que a versão cinco do PHP, permite a inclusão da interpretação dos comandos SQL (*structured Query Language*) por meio da extensão MySQLi que deu ao PHP total compatibilidade com novas funções da versão 4.1 do banco de dados.

2.5.2 Características

O PHP é uma linguagem para desenvolvimento Web, sua sintaxe é muito semelhante às linguagens C e C++, por ser uma linguagem simples, permite que vários programadores adaptem-se rapidamente a sua sintaxe e suas funções. Seu foco é a criação de sistemas com conteúdos dinâmicos para páginas da internet, dando ao usuário uma interatividade que gera efeitos na aplicação, em outras palavras, faz com que a inserção de informações traga um *feedback* para o cliente.

Para detectar o começo do código PHP, o servidor faz uma busca pelas *tags* “<?php” e “?>”, todo comando presente entre essas *tags* será interpretado pelo servidor e o código não ficará disponível para o cliente.

Segundo Ferreira (2010), o PHP é completamente modularizado, sendo possível alterá-lo com a instalação de outros módulos, que aumentam significativamente suas funções, possuindo inclusive extensões compatíveis a bancos de dados variados, como o Oracle e o MySQL. Por permitir a manipulação de arquivos com a linguagem, torna os sistemas mais dinâmicos e permitem que os usuários possam realizar várias ações na página. Suporta vários tipos de protocolos por poder usar bibliotecas de terceiros.

2.5.3 Sessões

O PHP Group (2010), grupo que mantém o PHP, descreve as sessões como uma forma de proteger os dados, possibilitando a criação de aplicações personalizadas. Os dados que foram criados, são usados somente em um único acesso sendo considerados como parâmetros para dar acessos e modificações.

2.5.3.1 Sessão

A sessão no PHP serve para proteger dados específicos até o usuário encerrar o navegador, ou a expiração da sessão, podendo ser por inatividade ou ação no aplicativo.

Para Muto A. (2006) no código, uma sessão é onde se podem guardar todos os tipos de valores e variáveis. Esta sessão conhecida como objeto sessão, possuindo uma identificação que é automaticamente enviada ao usuário quando a sessão é iniciada.

Um arquivo é criado no servidor para cada sessão criada por cliente, existindo apenas enquanto a sessão não for fechada. A gerência de sessões no servidor, serve para dar uma maior flexibilidade para o aplicativo poder mudar pelas determinações das variáveis, em sistemas com autenticação de usuário, as informações contidas em cada usuário, determinam quais dados ou funções ele poderá acessar.

Sessão é uma função muito limitada do servidor, por estar vinculada à memória não é indicado que se a utilize para armazenar informações que são desnecessárias para usuários específicos.

2.5.4 Segurança

A segurança em um Aplicativo Web é imprescindível, para que seus dados permaneçam íntegros afim de garantir sua confidencialidade. FIGUEIREDO (1999) afirma que com crescimento da web, a segurança é um dos fator de muita importância, principalmente para corporações e pessoas que tem como maior apelo comercial o ambiente da web.

Para que o aplicativo seja confiável, se utiliza métodos de criptografia de senhas verificação de sessão e tratamentos de formulários, tudo isso afim de dar uma maior confiabilidade a Aplicação Web e reduzir empecilhos que possam ocorrer.

2.5.5 Cookies

Cookies são um conjunto de dados enviados e recebidos entre os navegadores de internet e os servidores de páginas de internet, que são guardados em variáveis diferentes. Com o auxílio dos cookies pode se criar uma personalização para os usuários, como carrinhos

de compras em sites e-commerce e outros tipos de interações entre usuário e página de internet. Para Muto (2006) como o uso dos cookies, pode-se deixar informações guardadas no computador do usuário e usa-las futuramente, quando o usuário acessar novamente a página, desde que não exceda o tempo de expiração da sessão.

Poder armazenar certas variáveis no navegador do usuário dá a possibilidade de que a exclusividade de uma mudança na aplicação web não se perca quando for encerrada a página. Futuramente, quando o usuário inicie a página, a aplicação web verificará em seu cookie, quais os parâmetros de configuração nele contido.

O PHP Group (2010) descreve 6 tipos de variáveis contidos em cookies:

- a) *name*, nome do *cookie*;
- b) *value*, valor da variável;
- c) *expiration*, tempo de expiração do acesso aos dados;
- d) *path*, contém os nomes para quais diretórios o *cookie* é válido;
- e) *domain*, guarda o domínio ou o *host*;
- f) *security*, determina se o cookie será transmitido pelo *Hypertext Transfer Protocol secure* (HTTPS).

Caso um ou mais desses valores não seja informado, automaticamente são atribuídos valores padrões, no caso do valor *expiration* por exemplo, o cookie irá expirar assim que o usuário encerrar seu navegador de internet.

2.5.6 Biblioteca FPDF

Segundo Antunes Lamim (2010), o FPDF é uma biblioteca flexível, que possibilita modificar o formato das páginas, as margens, os cabeçalhos, as cores, quebra de linhas e outras opções que permitem a total modificação do conteúdo gerado.

A biblioteca FPDF, é uma classe da linguagem compatível com as versões cinco e quatro da linguagem PHP, que permite gerar arquivos PDF. Sua performance em gerar o documento fica um tanto comprometida devido ao fato de o FPDF não utilizar PDFlib, que é a biblioteca padrão da linguagem já embutida no servidor web. O nível de personalização que o FPDF permite, contribui para seu uso em

aplicativos que necessitam gerar arquivos com informações oriundas de consultas a banco de dados ou preenchimento de formulários, e sua facilidade de documentação no site do criador e em toda comunidade livre, também contribui para isso.

2.6 Banco de Dados MySQL

Desenvolvido pelos programadores Davi Axmark, Michael Widenius e Allan Larsson, o MySQL é um sistema para controle e gerência de bancos de dados, tendo duas versões, uma comercial, e outra aberta. Tendo sido publicado no ano 1996, e depois tendo sido comprado pela Sun Microsystems, e no ano de 2009 adquirido pela Oracle, nos dias de hoje está presente em cerca de 10 milhões de sistemas ao redor do mundo, em sistemas de setores públicos ou privados (STROPARO, 2010).

A principal vantagem do MySQL em comparação aos seus concorrentes e similares é a sua velocidade, além de sua afinidade com várias linguagens de programação, tornando a integração do código com o banco de dados muito simples. É usado em praticamente todo tipo de site, seja ele um portal grande e complexo, até os mais simples sites com pouco fluxo de dados e acesso. Videiro (2008) diz que o desempenho do MySQL é a sua maior propaganda, sendo considerado assim o mais popular banco de dados existente.

O MySQL possui muitos tipos de tabelas que são divididas em dois grupos, tabelas transacionais e não transacionais.

Algumas das principais características desse gerenciador de banco de dados são: sua portabilidade, dando suporte para todas as plataformas atuais, sua estabilidade e desempenho, e ser pouco exigente no que diz respeito ao hardware do servidor onde está hospedado.

2.6.1 Transações

As transações simples em qualquer banco de dados são interpretadas imediatamente, ou seja, após executado a sintaxe os dados são inseridos, editados, consultados ou excluídos. O suporte de transações são delimitadores que indicam o início e o fim de uma sintaxe, onde a execução do comando no MySQL é realizada ou abortada somente quando o delimitador final é interpretado. (MADEIRA, 2004).

O comando que indica o início da diretiva SQL é *start transaction*. O conteúdo da sintaxe só é aplicado no banco de dados se for finalizado com o comando *commit*, caso não seja incluído nenhuma instrução de delimitador no final, a operação ficará pendente. A sintaxe pode ser cancelada com algum mau comportamento do *software*, que utilizará meios de tratamento de erros incluindo o comando *rollback* como delimitador final, dessa forma todo o SQL da transação não será interpretado pelo banco de dados.

2.7 phpMyAdmin

Desenvolvido em código PHP, o phpMyAdmin possui uma interface Web para navegadores de internet, e possui as funcionalidades de criação, edição e exclusão de banco de dados, campos e tabelas, resumindo com o phpMyAdmin o usuário tem controle total da configuração do banco de dados MySQL. O software consiste em um conjunto de arquivos PHP, tornando-o multi plataforma, sendo necessário apenas um servidor Web para funcionar plenamente. Suas atualizações e desenvolvimento são criados por toda a comunidade de *software* livre mundial, garantindo que esteja disponível em setenta e duas linguagens (PHPMYADMIN DEVEL TEAM, 2003)

Nos dias de hoje o phpMyAdmin é muito usado em todo mundo, pois permite que os administradores de bancos de dados realizem todo tipo de operação remotamente em qualquer plataforma que tenha uma conexão com a internet.

2.8 Ajax

Ajax é a junção do XML (*eXtensible Markup Language*) e do JavaScript. Para Moroni (2007) seu objetivo é fazer com que as solicitações enviadas pelo cliente até o servidor não necessitem do carregamento de toda a página.

O método que o Ajax utiliza, proporciona a criação de páginas Web muito mais rápidas e interativas, devido ao fato de que seu tráfego de dados é muito menor, pois não precisa solicitar novamente as informações de uma página e carrega-la no navegador de internet.

Segundo Majer (2010, p. 2):

Um dos grandes problemas do HTML (e seus derivados PHP, ASP, ASP.NET, etc.) é o fato de que ao precisar acessar alguma informação muitas vezes em função de uma interação com o usuário), numa determinada página, esta informação só pode ser recuperada através de uma chamada à outra página. Isto faz com que o processo seja lento e tedioso.

A proposta da criação de um objeto Ajax, é solucionar muitos problemas das linguagens Web, dando acesso à informações de outras páginas, sem ter a necessidade de sair da página na qual se encontra. Garantindo assim uma melhor utilização por parte do usuário e um melhor desempenho na transição das informações .

A requisições ao servidor ocorrem através de um objeto criado em JavaScript com o Ajax, esse objeto é quem faz a ligação entre a interface do HTML e o servidor. O desenho abaixo compara os dois tipos de arquitetura, uma utilizando Ajax e a outra no “modo clássico”.

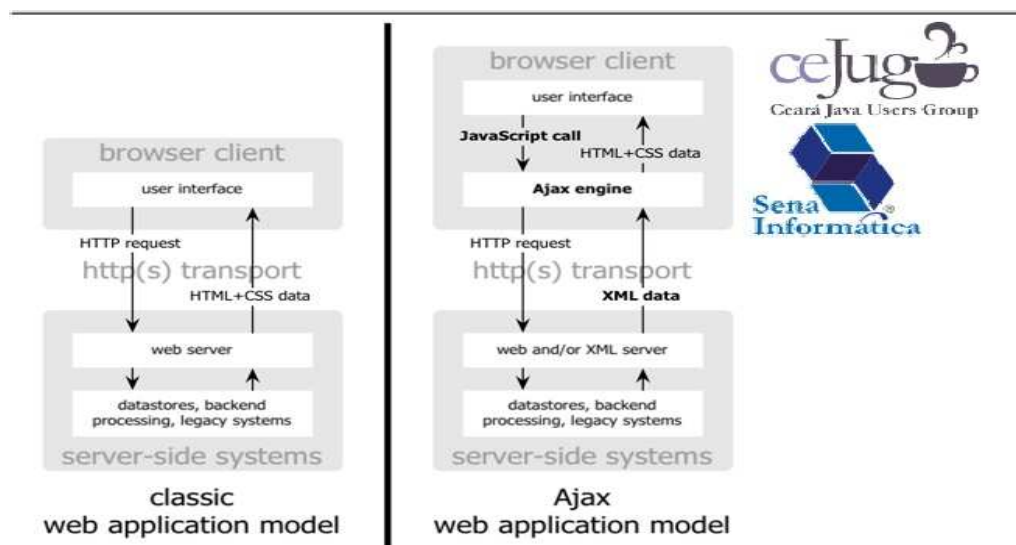


Figura 02: Comparativo entre as arquiteturas clássica e AJAX

Fonte: Ceará Java Users (2010)

O método utilizado pelo Ajax, permite que as requisições do HTTP sejam assíncronas, onde não se necessita da de imediato da resposta do servidor para continuar uma ação específica, e síncronas onde a aplicação só fará outra ação no momento que tiver a resposta anterior, colocando as requisições e respostas em sequencia. (PAIVA, 2005)

2.8.1 JavaScript

Criado em 1995, o Java Script é uma linguagem de programação por um funcionário da Netscape, sendo chamada de LiveScript, devido a um acordo entre Netscape e a Sun, foi renomeada definitivamente para JavaScript (AURELIO, 2004).

O Java Script é usado principalmente para validações de formulários, sendo possível criar funções dentro do Código HTML, por ser carregado ao mesmo tempo que o HTML, tem um desempenho de resposta muito bom, isso porque não depende de respostas do servidor para receber e enviar dados.

Quando foi criada a Linguagem JavaScript tinha muitos problemas no que diz respeito a compatibilidade com os navegadores existentes. Devido ao fato de que a NetScape, detentora da licença da linguagem, não liberou a licença de uso do JavaScript para o *browser* Internet Explorer, a Microsoft criou uma outra linguagem compatível chamada ActiveX. Com isso, os programadores e desenvolvedores de páginas Web foram obrigados a desenvolver scripts diferentes para cada navegador específico, fazendo com que o código necessite detectar o qual é o *browser* quando for executar uma página que utilize tais scripts TRIACCA (2007).

2.8.2 XML

O XML (*Extensible Markup Language*) é uma linguagem de marcação de dados que gera um formato específico para mostrar dados estruturados. Fazendo com que as declarações mais precisas do conteúdo e maiores resultados em buscas em outras plataformas. Novas formas de manipulação e visualização de dados via internet estão surgindo com o auxílio do XML (REIS 2005).

O XML é facilmente interpretado pelos computadores, podendo ser validado apenas usando arquivos de definição de tipo de documento. O XML armazena apenas um conteúdo, formatações e outros tipos de estilos são usados em arquivos diferentes.

2.9 Servidor Web

Servidor Web é um *software* instalado e executado em um computador conectado à

Internet, normalmente possui um *hardware* melhor que a maioria dos computadores pessoais. Este *software* é quem aceita os pedidos via protocolo HTTP, gerenciando e armazenando o conteúdo de páginas Web. Servidores Web armazenam o conteúdo das páginas, sendo acessados pelos domínios que estão ligados ao endereço de IP (*Internet Protocol*) do servidor (STEVENSON; FENNER; RUDOFF, 2004).

Um servidor Web deve ficar ligado o tempo todo, pois é ele quem garante que a página fique disponível na Internet, para que usuário a todo e qualquer momento possa acessar o conteúdo da página e assim cumprir com seu propósito. Os grandes portais e grandes páginas da web, usam uma configuração diferente para que possam garantir que a página web fique estável enquanto recebe milhares de acessos simultâneo.

2.9.1 Servidor Apache

Criado em 1995 e presente em cerca de 50% dos servidores existentes no mundo, o servidor web Apache, é um *software* livre e de muita utilidade para uso em servidores web. O nome Apache é inspirado em uma tribo de índios americanos que eram conhecidos por serem grandes combatentes principalmente no que se diz respeito a estratégias de guerra.

O Apache é compatível com o protocolo HTTP, sendo suas funções e recursos, compostos por vários tipos de módulos, é compatível com várias linguagens de programação. Para Coar e Bowen (2008, XI):

O pacote básico distribuído pela Apache Software Foundation é razoavelmente abrangente e bastante poderoso, e foi feito um grande esforço por parte dos programadores para evitar que o software sofresse do problema de ter muitos recursos que ninguém usa. Um aspecto do pacote o torna especialmente interessante: ele inclui extensibilidade por padrão.

Ser um *software* livre, torna o apache um sistema que constantemente sofre melhorias por diversos programadores ao redor do mundo, fazendo-o assim sempre atualizado e que acompanhe as novas tecnologias que são lançadas. Por ser compatível com o banco de dados MySQL permite utilizar o PHP e o MySQL juntos, estando presente em muitas páginas web e aplicações.

2.9.2 Xampp

Xampp é um pacote de softwares usados em servidores web, que contém softwares como o Apache e o PHP e outros softwares livres. Seu nome é um acrônimo que significa os principais *softwares* embutidos, a letra “x” significa o sistema operacional, é multiplataforma, a letra “a” de Apache, “m” de MySQL, e duas vezes a letra “p”, que significam PHP e PERL (APACHE FRIENDS, 2009).

O Xampp é muito usado para testes de páginas web, por ser de fácil utilização e instalação, além de que ele simula um servidor web, tornando desnecessário o upload de informações para um servidor web na internet

2.9.3 Wamp

Wamp é um pacote de softwares que instala simultaneamente o PHP, o MySQL e o Apache. Seu nome nasceu da junção das letras iniciais dos programas que ele instala em seu pacote (Apache MySQL e PHP), além da letra “W” do nome do sistema operacional para qual ele foi desenvolvido o Windows, atualmente existem versões menos populares para outros sistemas operacionais.

Sua principal vantagem é a rapidez com que cumpre suas funções, e a integração quase que perfeita entre o phpMyadmin e o servidor apache, que são muito utilizados em todo mundo.

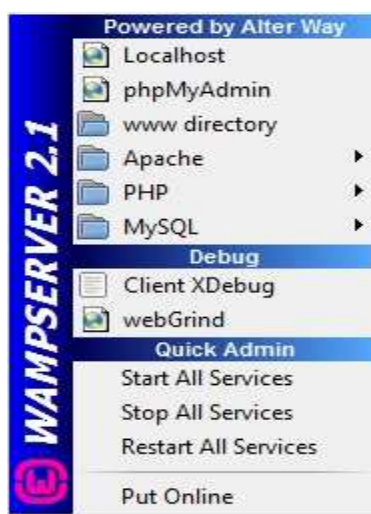


Figura 03: Interface do Wamp.

Fonte: Próprio autor.

III. PROJETO

O projeto do aplicativo utiliza-se da fundamentação teórica para definição de seu funcionamento e requisitos técnicos. O código fonte da aplicação é dividido em arquivos PHP, onde cada um é responsável por um módulo.

3.1 Análise de Requisitos

Os requisitos funcionais englobam de forma completa tudo o que deve conter na aplicação. De acordo com a análise feita, alguns itens como requisitos devem ser atribuídos:

RF-001	Logar no Sistema;
RF-002	Cadastrar Clientes;
RF-003	Pesquisar Cadastros de Clientes;
RF-004	Listar Clientes;
RF-005	Cadastrar Produtos;
RF-006	Modificar Cadastros de Produtos;
RF-007	Pesquisar Cadastros de Produtos;
RF-008	Gerar Relatório de Estoque;
RF-009	Cadastrar Lembrete;
RF-010	Encerrar Sessão;

3.2 Requisitos Não Funcionais

RNF-01	Conexão com Internet;
RNF-02	Browser de Internet;

3.3 Regras de Negócio

RN-01 Somente uma pessoa (representante ou profissional autônomo) pode ter acesso ao sistema;
RN-02 Para acessar qualquer função do sistema é necessário autenticação através de Login e Senha;

3.4 Problemas a Serem Sanados

- Controle de estoque de produtos;
- Controle de cadastro de clientes;
- Manter o estoque atualizado;
- Manter cadastro de produtos;
- Manter cadastro de clientes;
- Listar cadastro de produtos;
- Listar cadastro de clientes;
- Alterar cadastro de produtos.

3.5 Melhorias

- Maior controle de estoque;
- Maior facilidade para localizar produtos e clientes.
- Portabilidade.
- Compatibilidade com praticamente todo tipo hardware que utilize um sistema operacional

IV FUNDAMENTAÇÃO TEÓRICA

A seguir são apresentados os conceitos e ferramentas utilizadas no desenvolvimento do sistema e documentação.

4.1 DREAMWEAVER

O Dreamwaver é um *software* para criação, desenvolvimento e edição de páginas e aplicativos Web, dando suporte para várias tecnologias e linguagens de programação como o PHP e possuindo um poderoso editor HTML integrado (REMOALDO 2008).

Possuí uma interface amigável e intuitiva que facilita na criação e desenvolvimento de códigos para páginas, podendo inclusive em seu modo de *design* criar botões de forma rápida prática sem precisar necessariamente escrever um código para cada botão. A função *preview* permite ao usuário testar o código elaborado chamando o browser de internet para que a mesmo leia o arquivo salvo e execute o site, para o usuário ter uma base de como está ficando a página e verificar sua compatibilidade.

O Dreamweaver possui também uma função para criação e manipulação de formulários, que facilita o desenvolvimento da parte do *layout* de uma página Web, afim de que o programador mantenha o foco no código da linguagem do servidor.

4.2 Unified Modeling Language – UML

Para CONALEN (2000) a UML é uma linguagem que serve para modelar sistemas de *software* intensivos. No caso das páginas e aplicações Web, algumas extensões foram agregadas, usando modelos que correspondam ao modelo empregado.

O padrão UML não se ajustava perfeitamente às aplicações Web, sendo assim tomaram a atitude de tentar sanar esses problemas e definiram um modo formal para estender tais funcionalidades através de uma semântica diferente aplicada a modelagem.

Ainda segundo CONALEN (2000) alguns padrões foram definidos, de acordo com a arquitetura utilizada. Porém apenas três padrões foram destinados para as arquiteturas web:

- *Thin Web Client*: é utilizado em aplicações padronizadas onde os objetos estarão sendo

executados no servidor web;

- *Thick Web Client*: possibilita que alguns objetos sejam executados diretamente no cliente;
- *Web Delivery*: é usado para sistemas de objetos distribuídos.

Modelos nos ajudam a compreender um sistema, tornando os seus detalhes mais simples para se entender. A opção do que modelar tem um grande efeito no entendimento do problema e na solução do mesmo, aplicações web são representadas com um conjunto de modelos.

4.2.1 Diagrama de caso de uso

Diagramas de caso de uso tem como finalidade descrever qual funcionalidade o sistema em questão deve prover. Geralmente um sistema provê vários serviços para determinados atores. Usualmente a funcionalidade de um sistema é muito extensa para ser analisada de forma individual, assim é necessário ter um mecanismo para dividir essa funcionalidade em partes pequenas para serem gerenciadas. O conceito do caso de uso é muito útil para esse fim.

4.2.2 Diagrama de Classe

Fowler (2000) diz “Um diagrama de classes descreve os tipos de objetos no sistema e os vários tipos de relacionamentos estáticos que existem entre eles”. Uma classe representa a propriedade e o comportamento dos objetos em um sistema, e assim é importante representar seus relacionamentos pois essas classes não existem sozinhas.

V MODELAGEM DO SISTEMA

A seguir são apresentados os conceitos, modelagens e arquiteturas utilizadas no desenvolvimento do sistema, além das figuras que contém as telas demonstrando algumas ações do sistema.

5.1 Diagrama de Caso de Uso da Aplicação

Com o uso de um diagrama de caso de uso, pode-se visualizar todas as funções que o ator pode exercer. O ator é representado por um usuário, que pode exercer determinadas funções no sistema. O esquema serve para demonstrar as operações padrões que podem ser realizadas, ajudando a desenvolver a aplicação, com o intuito de cumprir com seus objetivos.

O Diagrama de caso de uso, auxilia no entendimento das possibilidades que o usuário possui com relação a aplicação, podendo entender de uma forma mais abrangente as ações, comportamento e interações que o *software* permite.

A figura a seguir, representa o diagrama de caso de uso utilizado para o desenvolvimento e entendimento do projeto:

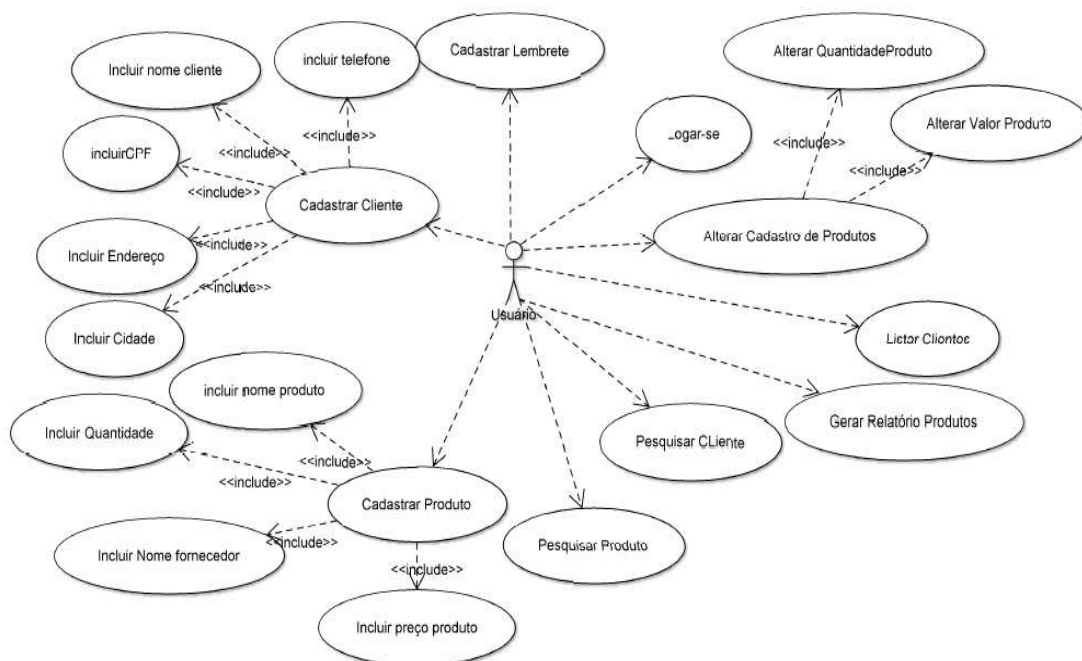


Figura 04: Diagrama caso de uso da aplicação

Fonte: Próprio Autor (2012)

5.2 Arquitetura Geral da Aplicação

A aplicação web é executada entre o cliente e o servidor web ao mesmo tempo. O browser é quem solicita e processa o código HTML, ao mesmo tempo o navegador envia valores para o servidor, e os recebe novamente. Com isso o código PHP é interpretado, e executa suas funções no banco de dados. A figura abaixo demonstra como funciona a arquitetura:

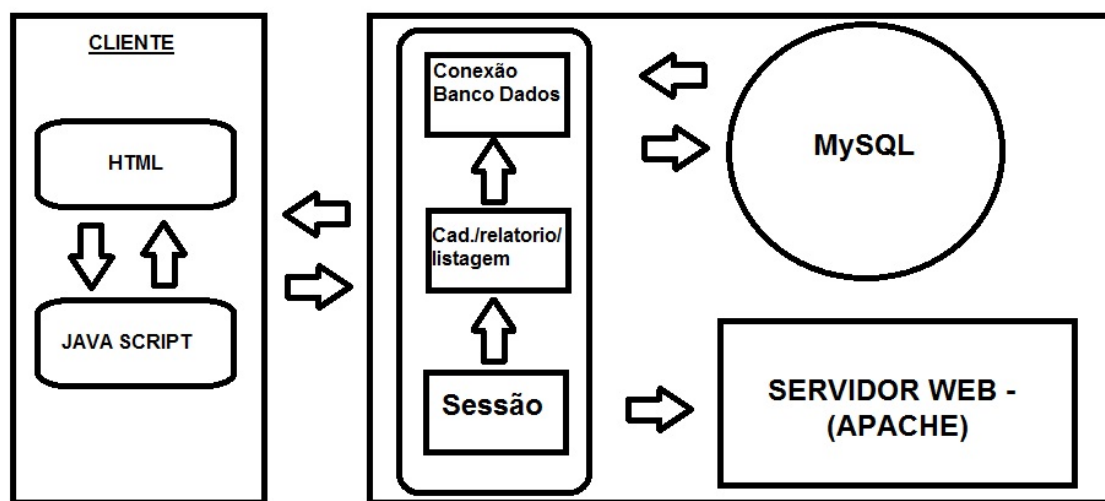


Figura 05: Diagrama da arquitetura geral da aplicação.

Fonte: Próprio Autor (2012)

5.3 Diagrama de Entidade Relacionamento

O Diagrama de Entidade Relacionamento é um modelo de dados conceitual de alto-nível, ou seja, seus conceitos foram projetados para serem compreensíveis a usuários, descartando detalhes de como os dados são armazenados.

Atualmente, o Diagrama de Entidade Relacionamento é usado principalmente durante o processo de projeto da base de dados.

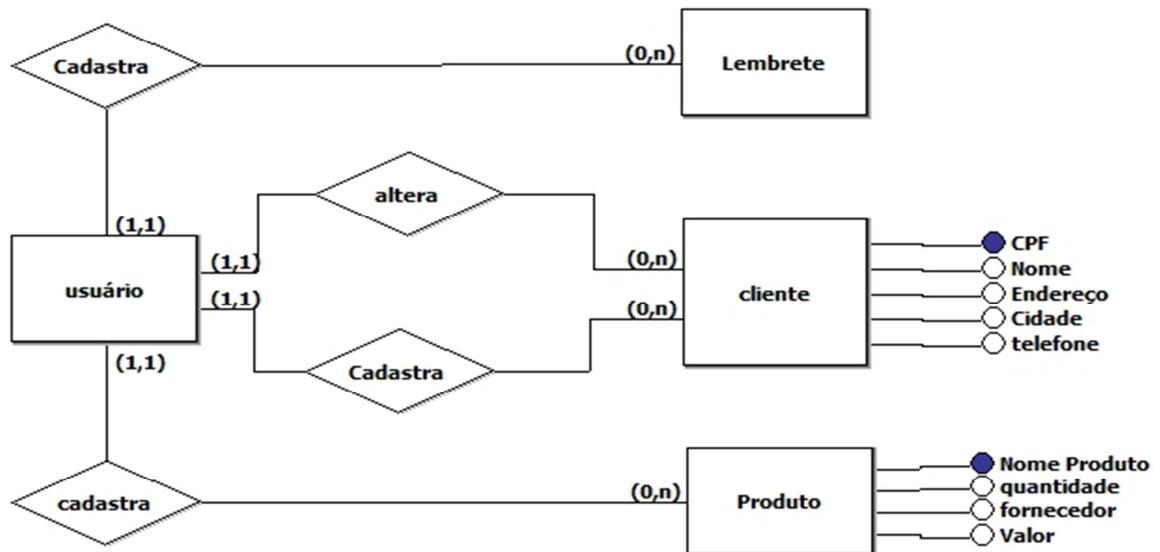


Figura: 06: Diagrama de entidade relacionamento da aplicação.
Fonte: Próprio Autor (2012).

5.2 Diagrama de Classe

Diagramas de Classes representam as estruturas de um sistema usando o conceito de classe e suas relações. Seu modelo de classes é resultado de um processo complexo de abstração, no qual se identifica os objetos que tem algum valor real para o sistema em estudo.

A figura abaixo representar um diagrama de classe de uma aplicação Web:

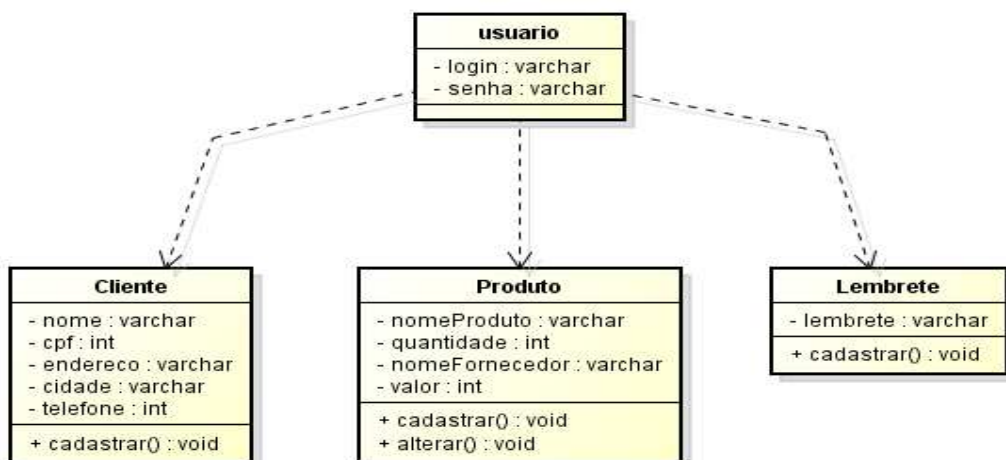


Figura 07: Diagrama de classe da aplicação.
Fonte: Próprio Autor (2012)

5.3 Interfaces do sistema

Neste capítulo serão mostradas e apresentadas as interfaces presentes no sistema.



Figura: 08 – Interface de *login* para acesso ao sistema.
Fonte: Próprio Autor (2012).

A figura 08 mostra que para acessar qualquer tela do sistema o usuário precisa fornecer um usuário e uma senha, que são fornecidas pelos desenvolvedores da aplicação. Essa função conta com o uso do Ajax, que permite que as informações sejam trocadas com servidor, sem precisar carregar toda a página novamente.



Figura 09: Tela inicial do sistema.
Fonte: Próprio Autor (2012).

A Figura 09 exibe a tela de entrada do sistema quando o usuário acessa o sistema, nela é exibido o lembrete que foi cadastrado previamente na figura 10.

Início Cadastrar Lembrete Cadastrar Cliente Cadastrar Produto Pesquisar Clientes Listar Clientes Pesquisar Produtos Listar Produtos Atualizar Produtos

Encerrar Sessão

Bem vindo ao CAD-1.0

Cadastrar Recados

Bom dia!

Cadastrar Recado Apagar Formulário

Figura 10: Tela de cadastro de lembrete.
Fonte: Próprio Autor (2012).

Início Cadastrar Lembrete Cadastrar Cliente Cadastrar Produto Pesquisar Clientes Listar Clientes Pesquisar Produtos Listar Produtos Atualizar Produtos

Encerrar Sessão

Bem vindo ao CAD-1.0

Cadastrar Cliente

Nome: João da Silva Melo

CPF/CNPJ: 06708343827

Endereço: Barão do Rio Branco, 29

Cidade: Cuntibanos

Telefone: 4932412833

Cadastrar Limpar Campos

Figura 11: Tela de cadastro de Cliente.
Fonte: Próprio Autor (2012).

A figura 11 apresenta os campos necessários para o cadastro de clientes, contendo campos a serem preenchidos pelo usuário, para que possa ser pesquisado, como mostra a figura 12.

Início Cadastrar Lembrete Cadastrar Cliente Cadastrar Produto Pesquisar Clientes Listar Clientes Pesquisar Produtos Listar Produtos Atualizar Produtos

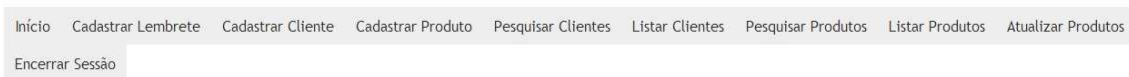
Encerrar Sessão

Bem vindo ao CAD-1.0

Pesquisa Cliente

CPF: (11 digitos) 06508043927 Pesquisar

Figura 12: Tela de pesquisa de Cliente.
Fonte: Próprio Autor (2012).



Bem vindo ao CAD-1.0

Clientes

Nome do Cliente	CPF	Endereço	Cidade	Telefone
Leonardo Felipe Rigon	06508043927	Salomão	Curitiba	32452222

Figura 13: Tela de resultado de pesquisa de cliente.
Fonte: Próprio Autor (2012).

A figura 13 representa o resultado da pesquisa feita na figura 12, nela apareceram as informações previamente cadastradas, o sistema faz uma busca em seu banco de dados e imprime os dados na tela do usuário.



Bem vindo ao CAD-1.0

Clientes

Nome do Cliente	CPF	Endereço	Cidade	Telefone
Leonardo Felipe Rigon	06508043927	Salomão	Curitiba	32452222
Luis Carlos da Silva	06408143927	Capitão Antonio de Melo, 49	Ponte Alta do Norte	4933233451

Figura 14: Tela de listagem do cadastro de clientes.
Fonte: Próprio Autor (2012).

A figura 14, demonstra como são exibidos os clientes no sistema, todos ficam listados de acordo com a ordem de cadastro, com seus dados exibidos de forma clara e objetiva.



Bem vindo ao CAD-1.0

Cadastrar Produto

Nome:

Quantidade:

Nome do Fornecedor:

Preço:

Figura 15: Tela de cadastro de Produto.
Fonte: Próprio Autor (2012).

A figura 15 representa a tela de cadastro de produtos, que posteriormente serão pesquisados na figura 16.

Início Cadastrar Lembrete Cadastrar Cliente Cadastrar Produto Pesquisar Clientes Listar Clientes Pesquisar Produtos Listar Produtos Atualizar Produtos
Encerrar Sessão

Bem vindo ao CAD-1.0

Pesquisa Produto

Nome do Produto:

Figura 16: Tela de pesquisa de produtos.
Fonte: Próprio Autor (2012).

Início Cadastrar Lembrete Cadastrar Cliente Cadastrar Produto Pesquisar Clientes Listar Clientes Pesquisar Produtos Relatório de Produtos Manipular Estoque
Encerrar Sessão

Bem vindo ao CAD-1.0

Produtos

Nome do Produto	Quantidade em Estoque	Nome do Fornecedor	Valor de Venda
Balaio Sintético	12	João da Silva	13

Figura 17: Tela de resultado da pesquisa de produtos.
Fonte: Próprio Autor (2012).

Início Cadastrar Lembrete Cadastrar Cliente Cadastrar Produto Pesquisar Clientes Listar Clientes Pesquisar Produtos Listar Produtos Atualizar Produtos
Encerrar Sessão

Bem vindo ao CAD-1.0

Relatório de Produtos

Nome do Produto	Quantidade	Nome do Fornecedor	Valor R\$
Balaio Sintético	12	João da Silva	13
Cesta de Palha Pequena	12	Antonio da Silva	12

Figura 18: Listagem de produtos.
Fonte: Próprio Autor (2012).

Início Cadastrar Lembrete Cadastrar Cliente Cadastrar Produto Pesquisar Clientes Listar Clientes Pesquisar Produtos Listar Produtos Alterar Estoque Produtos
Encerrar Sessão

Bem vindo ao SISTEMA DE GERENCIAMENTO DE ESTOQUE

Movimentar Estoque, altere o número de produtos que de deseja!

Nome do Produto	Quantidade Atual	Fornecedor	Valor
Balaio Sintético	255	João da Silva	13

Figura 19 – Tela de alteração de cadastro de produtos.
Fonte: Próprio Autor (2012).

A figura 19 demonstra a tela de alteração de cadastro dos produtos, podendo apenas serem mudadas as informações referentes a estoque e valor do produto, pode-se fazer um controle superficial de entrada e saída de mercadorias, apenas alterando a quantidade que foi vendida pela quantidade atual de estoque de um determinado produto.

Início	Cadastrar Lembrete	Cadastrar Cliente	Cadastrar Produto	Pesquisar Clientes	Listar Clientes	Pesquisar Produtos	Listar Produtos	Alterar Estoque Produtos
Encerrar Sessão								
Bem vindo ao SISTEMA DE GERENCIAMENTO DE ESTOQUE								
Relatório de Produtos								
Nome do Produto	Quantidade	Nome do Fornecedor	Valor R\$					
Balão Sintético	300	João da Silva	13					

Figura 20 – Tela listagem de produtos com alteração de estoque.

Fonte: Próprio Autor (2012).

Por fim a figura 20 exhibe os resultados que ocorreram devido a alterações feitas na tela de alteração de cadastro de produtos.

VI CONCLUSÃO

Com o surgimento de novas tecnologias, e o uso constante da internet em todo o mundo, as aplicações web tem se tornado muito populares entre os usuários da internet. Aos poucos os sistemas de informação antigos estão se tornando ultrapassados, dando espaço para a entrada de sistemas que fazem a mesma coisa só que de forma mais barata e que fazem as mesmas coisas, com a vantagem de poderem ser acessados de qualquer lugar que se tenha conexão com a internet, e sem a instalação de softwares e intervenções de terceiros para que o sistema funcione.

Com o desenvolvimento desta aplicação web, nota-se que apesar de muito recente, existem muitas ferramentas para auxiliar a criação e desenvolvimento de páginas e aplicações Web.

Atualmente é inviável uma empresa, que queira ser competitiva no mercado, não utilizar sistemas para aperfeiçoar seus serviços, independente do tamanho e do ramo dela.

Aplicações Web tendem a ter um custo muito inferior aos antigos sistemas, principalmente no que diz respeito a sistemas que utilizem bancos de dados para clientes e produtos, além de que podem ser acessadas em qualquer lugar do mundo.

Conclui-se então que em futuro próximo os sistemas comuns com servidores deem espaço a aplicações web com servidores remotos, baixando o custo de operação das empresas e criando um novo tipo de modelo de gerencia de dados.

VII REFERÊNCIAS BIBLIOGRÁFICAS

ANTUNES LAMIM, Jonatham. **Gerando PDFs com PHP e a classe FPDF**. 2010.

Disponível em:

<http://www.oficinadanet.com.br/artigo/php/gerando_pdfs_com_php_e_a_classe_fpdf_instalacao_e_primeiras_linhas_de_codigo> . Acesso em: 23 out. 2012.

APACHE FRIENDS. **XAMPP**.2009. Disponível em:

<http://www.apachefriends.org/pt_br/xampp.html>. Acesso em: 18 out. 2012.

AURELIO, Dorival Eloy. **JavaScript**. 2004. Disponível em:

<<http://orbita.starmedia.com/~edaurelio/javascript.htm>>. Acesso em: 12 jun. 2012.

COAR, Ken; BOWEN, Rich. **Apache Guia Prático**. Rio de Janeiro: Alta Books, 2008. 272 p.

CONALLEN, J. **Building Web applications with UML**. USA: Addison Wesley, 2000.

CHAUBEY, Rahul; SURESH, J.K. **Integration vs Development: an Engineering Approach to Building Web Applications**. Proceedings 2001. Austrália: 2001. 171 p.

EAN Brasil. **Associação Brasileira de Automação – AUTOMAÇÃO COMERCIAL** Disponível em < <http://www.gs1br.org>> Acesso em 13/12/2012.

FIGUEIREDO, Antonio. **Administração de sistemas e segurança**. 1999. Disponível em: <<http://www.ccuec.unicamp.br/revista/infotec/admsis/admsis6-1.html>>. Acesso em: 25 out. 2012.

FLANAGAN, David. **JavaScript: O Guia Definitivo**. 4 ed. São Paulo: Artmed, 2002. 816 p.

FOWLER, Martin. **UML essencial: um breve guia para a linguagem padrão de modelagem de objetos**. 3. ed. Porto Alegre: Bookman, 2005. 165 p.

GOURLEY, David; TOTTY, Brian. **HTTP: TheDefinitiveGuide**. Sebastopol, Ucrânia: O'Reilly Media, 2002. 635 p.

LI, Hongshng; SHI, Tielin; YANG, Shuzi. **An approach of sampling computing for wavelet analysis and its Web-Based implementation**. Proceedings, 2000. WCCC-ICSP 2000. 395-398 p.

LOGMAN, Wesley Addison. **A history of HTML**.1998. Disponível em:

<<http://www.w3.org/People/Raggett/book4/ch02.html>>. Acesso em: 23 out. 2012.

MADEIRA, Frederico; **Suporte a transações**. 2004. Disponível em:

<http://imasters.com.br/artigo/2787/mysql/suporte_a_transacoes/>. Acesso em: 18 out.2012.

MAJER, Carlos. **Desenvolvimento Web: Introdução ao AJAX**. [São Paulo], 07 maio. 2010. Disponível em: <<http://www.ebah.com.br/desenvolvimento-webintroducao-ao-ajax-pdf-a48787.html>>. Acesso em: 19 out. 2012.

MCLEAN, Epharaim; TURBAN, Efraim; WETHERBE, James. **Tecnologia da informação para gestão**. 3. ed. São Paulo: Artmed, 2002. 645 p.

MELO, Alexandro Altair; G.F. NASCIMENTO, Mauricio. **PHP Profissional**. São Paulo: Novatec, 2007. 464 p.

MORONI, Herbert. **Criação de sites em Ajax**. São Paulo: Digeratti Books, 2007. 128 p.

MUTO, Claudio Adonai. **PHP e MySQL – Guia Introdotório**. 3. ed. Rio de Janeiro: Brasport, 2006. 412 p.

OFFUTT, J. And Hayes, J. Huffaman. **Applying a Semantic Fault Model to the Empirical Study of Corrective Maintenance**. 8th IEEE Workshop on Empirical Studies of Software Maintenance, Montreal, Canada, Outubro de 2002.

PAIVA, Raphael. **Desmitificando o Ajax**. Esquema. 2005. Disponível em: <http://www.jack.eti.br/www/arquivos/apostilas/web/apostila_ajax.pdf>. Acesso em: 23 out. 2012.

PHP GROUP. **PHP**. 2010. Disponível em: <<http://php.net/>>. Acesso em: 20 out. 2012.

PHPMYADMIN DEVEL TEAM. **phpMyAdmin: About**.2003. Disponível em: <http://www.phpmyadmin.net/home_page/index.php>. Acesso em: 19 out. 2010.

PLATHEY, Oliver. **FPDFLibrary**. 2008. Disponível em: <<http://www.fpdf.org>>. Acesso em: 20 out. 2012.

REMOALDO, Pedro. **Guia Pratico do Dreamweaver CS3 com PHP, JavaScript e Ajax**. Lisboa, Portugal:Centro Atlântico, 2008. 676 p.

RODRIGUES DE FREITAS; Andrey. **Pericia forense aplicada a informática: ambiente Microsoft**. Rio de Janeiro: Brasport, 2006. 215 p.

SANTOS, Alfredo. **Quem mexeu no meu sistema?: Segurança em sistemas de informação**. Rio de Janeiro: Brasport, 2008. 212 p.

SOMERA, Guilherme. **Treinamento prático em CSS**. São Paulo: Digeratti Books, 2006. 160 p.

SOUZA, Sandro J. **A extensão MySQLi**. 2007. Disponível em: <<http://www.htmlstaff.org/ver.php?id=10071>>. Acesso em: 22 out. 2012.

STALLINGS, William. **Redes e Sistemas de Comunicação de Dados**. Rio de Janeiro: Elsevier, 2005. 460 p.

STEVENS, W. Richard; FENNER, Bill; RUDOFF, M. Andrew. **Programação de Rede Unix**. São Paulo: Artmed, 2004. 906 p.

STROPARO, Elder. **Historia do MySQL**. 2010. Disponível em: <<http://elderstroparo.blogspot.com/2010/01/historia-do-mysql.html>>. Acesso em: 22 out. 2012.

TRACCA, Anderson. **História do Javascript**. Disponível em: <[http://www.andersontracca.com/index.php?Hist%F3ria-do Javascript&id_postagem=2](http://www.andersontracca.com/index.php?Hist%F3ria-do+Javascript&id_postagem=2)>. Acesso em: 24 out. 2012.

VIDEIRO, Rafael. **Criação de bases de dados em linguagem SQL**. Disponível em: <<http://mysql.softonic.com.br/>>. Acesso em: 26 out. 2012.

WELLING, Luke; THOMSON, Laura. **PHP e MySQL Desenvolvimento Web**. Rio de Janeiro: Elsevier, 2003. 675 p.

WORLD WIDE WEB CONSORTIUM. **Extensible Markup Language (XML)**. Disponível em: <<http://www.w3.org/XML/>>. Acesso em: 26 out. 2012

OFFUTT, J. And Hayes, J. Huffaman. **Applying a Semantic Fault Model to the Empirical Study of Corrective Maintenance**. 8th IEEE Workshop on Empirical Studies of Software Maintenance, Montreal, Canada, Outubro de 2002.