

UNIVERSIDADE REGIONAL DE BLUMENAU
CENTRO DE CIÊNCIAS EXATAS E NATURAIS
CURSO DE CIÊNCIAS DA COMPUTAÇÃO
(Bacharelado)

**DESENVOLVIMENTO DE UM SITE DE COMÉRCIO
ELETRÔNICO UTILIZANDO PHP E MYSQL**

TRABALHO DE CONCLUSÃO DE CURSO SUBMETIDO À UNIVERSIDADE
REGIONAL DE BLUMENAU PARA A OBTENÇÃO DOS CRÉDITOS NA
DISCIPLINA COM NOME EQUIVALENTE NO CURSO DE CIÊNCIAS DA
COMPUTAÇÃO — BACHARELADO

CARLOS ALBERTO ZAGO

BLUMENAU, DEZEMBRO/2001

2001/2-07

DESENVOLVIMENTO DE UM SITE DE COMÉRCIO ELETRÔNICO UTILIZANDO PHP E MYSQL

CARLOS ALBERTO ZAGO

ESTE TRABALHO DE CONCLUSÃO DE CURSO, FOI JULGADO ADEQUADO
PARA OBTENÇÃO DOS CRÉDITOS NA DISCIPLINA DE TRABALHO DE
CONCLUSÃO DE CURSO OBRIGATÓRIA PARA OBTENÇÃO DO TÍTULO DE:

BACHAREL EM CIÊNCIAS DA COMPUTAÇÃO

Prof. Wilson Pedro Carli — Orientador na FURB

Prof. José Roque Voltolini da Silva — Coordenador do TCC

BANCA EXAMINADORA

Prof. Wilson Pedro Carli

Prof. Alexander R. Valdameri

Prof. Everaldo Artur Grahl

SUMÁRIO

LISTA DE FIGURAS	V
LISTA DE QUADROS	VII
LISTA DE SIGLAS	VIII
DEDICATÓRIA.....	IX
AGRADECIMENTOS	X
RESUMO	XI
ABSTRACT	XII
1 INTRODUÇÃO	1
1.1 OBJETIVOS DO TRABALHO	3
1.2 ESTRUTURA DO TRABALHO	3
2 COMÉRCIO ELETRÔNICO	5
2.1 BUSINESS-TO-BUSINESS (B2B)	6
2.2 BUSINESS-TO-CONSUMER (B2C)	6
3 FERRAMENTAS	7
3.1 PERSONAL WEB SERVER (PWS)	7
3.1.1 PERMISSÕES DE ACESSO AOS DIRETÓRIOS	7
3.1.2 BUSCA NO DIRETÓRIO	8
3.1.3 LOG DE ATIVIDADE DO SITE DA WEB	8
3.1.4 DESEMPENHO DO SITE.....	9
3.2 A LINGUAGEM PHP.....	9
3.2.1 ESTRUTURA DE UM PROGRAMA PHP	10
3.2.2 RESULTADO NA WEB	12
3.2.3 COMENTÁRIOS NO PHP.....	13

3.2.4 VARIÁVEIS NO PHP	14
3.2.5 OPERADORES DO PHP	17
3.2.6 FORMULÁRIOS HTML.....	18
3.2.7 FUNÇÕES DO PHP	19
3.3 MYSQL.....	21
3.3.1 CONECTANDO UM BANCO DE DADOS.....	22
4 DESENVOLVIMENTO DO TRABALHO	25
4.1 LEVANTAMENTO DE INFORMAÇÕES.....	25
4.2 REQUISITOS PRINCIPAIS DO PROBLEMA A SER TRABALHADO.....	26
4.3 ESPECIFICAÇÃO	27
4.3.1 DIAGRAMAS DE CASO DE USO	27
4.3.2 MER LÓGICO E FÍSICO	29
4.3.3 DICIONÁRIO DE DADOS	30
4.3.4 ESQUEMA NAVEGACIONAL DO SITE	32
4.3.5 MIGRANDO O BANCO DE DADOS DE PRODUTOS	35
4.4 IMPLEMENTAÇÃO	35
4.4.1 TÉCNICAS E FERRAMENTAS UTILIZADAS.....	35
4.4.2 OPERACIONALIDADE DA IMPLEMENTAÇÃO.....	36
5 CONCLUSÕES	46
5.1 EXTENSÕES	47
ANEXO 1 – LOGIN DO CLIENTE	48
ANEXO 2 – CADASTRO DE CLIENTES	51
ANEXO 3 – CONSULTA POR NOME OU TRECHO DO NOME DE PRODUTOS	59
REFERÊNCIAS BIBLIOGRÁFICAS	62

LISTA DE FIGURAS

Figura 1 – Diagrama de caso de uso primário do sistema.....	28
Figura 2 – MER lógico.....	29
Figura 3 – MER físico.....	29
Figura 4 – Dicionário de dados (parte 1).....	30
Figura 5 – Dicionário de dados (parte 2).....	31
Figura 6 – Esquema navegacional do <i>site</i> do cliente.....	32
Figura 7 – Esquema navegacional do <i>site</i> do operador.....	33
Figura 8 – Esquema navegacional do site do administrador.....	34
Figura 9 – Página principal.....	37
Figura 10 – <i>Login</i> do cliente.....	37
Figura 11 – Página de boas-vindas.....	38
Figura 12 – Login do cliente ok.....	39
Figura 13 – Cadastro de clientes.....	39
Figura 14 – Ofertas da semana.....	40
Figura 15 – Ofertas para cliente com login ok.....	41
Figura 16 – Consulta de produtos pelo nome ou trecho do nome.....	41
Figura 17 – Compra de produtos.....	42
Figura 18 – Carrinho de compras.....	43
Figura 19 – Escolha da forma de pagamento e local de entrega.....	43

Figura 20 – Enquete satisfação do cliente.....	44
Figura 21 – Área administrativa.....	45

LISTA DE QUADROS

Quadro 1 – Exemplo de estrutura do programa PHP.....	11
Quadro 2 – Exemplo 1 de resultado do programa PHP.....	12
Quadro 3 – Exemplo 2 de resultado do programa PHP.....	13
Quadro 4 – Exemplo 3 de resultado do programa PHP.....	13
Quadro 5 – Exemplo 1 de comentários no PHP.....	14
Quadro 6 – Exemplo 2 de comentários no PHP.....	14
Quadro 7 – Variáveis no PHP – Exemplo 1.....	15
Quadro 8 – Variáveis no PHP – Exemplo 2.....	15
Quadro 9 – Variáveis no PHP – Exemplo 3.....	16
Quadro 10 – Operadores de atribuição.....	18
Quadro 11 – Operador de string.....	18
Quadro 12 – Formulários HTML.....	19
Quadro 13 – Funções no PHP – Exemplo 1.....	20
Quadro 14 – Funções no PHP – Exemplo 2.....	20
Quadro 15 – Passagem de argumentos.....	21

LISTA DE SIGLAS

ASP	<i>Active Server Pages</i>
B2B	<i>Bussines-to-Bussines</i>
B2C	<i>Bussines-to-Consumer</i>
FTP	<i>File Tranfers Protocol</i>
HTML	<i>Hyper Text Markup Language</i>
HTTP	<i>Hyper Text Transfer Protocol</i>
IIS	<i>Internet Information Server</i>
IMAP	<i>Internet Message Access Protocol</i>
NCSA	<i>National Center For Supercomputing Applications</i>
NMTP	<i>Network Management Tranfer Protocol</i>
PHP	<i>Personal Home Page Tools</i>
PHP/FI	<i>Personal Home Page Tools / Form Interpreter</i>
POP3	<i>Post Office Protocol Version 3</i>
PWS	<i>Personal Web Server</i>
SNMP	<i>Simple Network Management Protocol</i>
SQL	<i>Structured Query Language</i>
URL	<i>Uniform Resourse Locator</i>
WWW	<i>World Wide Web</i>

DEDICATÓRIA

Eu gostaria de dedicar meu esforço para o término deste trabalho a minha família que apostou muito na minha capacidade, além de minha namorada e amigos que tiveram um papel muito importante para a conclusão deste trabalho.

AGRADECIMENTOS

Agradeço as pessoas que sabiam que eu era capaz de desenvolver este trabalho por si próprio. Além de pessoas que me ajudaram momentos muito importantes como o prof. Wilson Carli. Agradeço também a força que a galera de Lages, meus amigos, tanto nas horas boas como nas ruins, mas sempre me apoiando, sabendo que eu iria terminar por mais demorado que fosse. Muito obrigado a todos, principalmente minha família que ao passar destes anos da faculdade aumentou muito, pois são muitos novos “irmãos”, colegas que se tornaram irmãos.

RESUMO

O trabalho consiste na especificação e implementação de um *site* de comércio eletrônico utilizando PHP e MySQL, para uma loja de ferramentas, materiais de construção e madeiras, situada na região de Lages(SC). O aplicativo é desenvolvido em PHP que acessa o banco de dados MySQL e permite que somente usuários cadastrados realizem compras, além de enviar e-mails promocionais e publicitários aos clientes. O aplicativo tem como finalidade ajudar a empresa na tomada de decisão em seus negócios.

ABSTRACT

The work consists of the specification and implementation of a site of electronic trade using PHP and MySQL, for a store of tools, construction materials and wood, placed in the area of Lages (SC). The application is developed in PHP that access the database MySQL and it allows that only registered users accomplishes purchases, besides sending e-mails of promotions and advertising to the customers. The application has as purpose helping the company in the taking of decision in its business.

1 INTRODUÇÃO

De acordo com Ean Brasil (1999), automação comercial consiste na informatização de todas as operações internas da empresa, bem como na integração desses processos internos com o mundo externo e até mesmo com os consumidores, além de contribuir decisivamente para que as empresas comerciais conquistem e mantenham a clientela e obtenham lucros adicionais. A automação comercial visa a racionalização dos processos, eliminação de tarefas que não agregam valor, minimização de erros, melhoria do atendimento, aproveitamento da informação. Como consequência a automação comercial confere maior produtividade e confiabilidade aos processos da empresa, propiciando-lhe claras vantagens competitivas. Em termos bem práticos, através da automação comercial o lojista pode obter lucros adicionais e cativar a clientela.

Desta forma, para o desenvolvimento de iniciativas de comércio eletrônico, necessita-se uma preocupação mínima com os aspectos rotineiros de um negócio varejista qualquer, como atenção a logística e atendimento ao cliente. Com o enorme crescimento do *e-commerce* (Comércio Eletrônico), o *Bussines-to-Bussines* (B2B) e o *Bussines-to-Consumer* (B2C) vem sendo adotados por várias empresas nos variados ramos de negócio. O B2B, é uma transação entre empresas, com objetivo de aumentar a eficiência da cadeia produtiva, com isso ocasionando uma troca de informações estratégicas e não podendo ser disponibilizadas livremente. Já o B2C, possui um público com diferentes intenções, além de ser um serviço adicional, cria-se e estreita-se um relacionamento com bons clientes, aumentando a conveniência do processo de compras. Ter uma estratégia bem definida de comércio eletrônico e criar condições para que a loja virtual realmente proporcione uma experiência de compra prazerosa e bem sucedida aos seus clientes no momento em que desejarem é um dos objetivos principais do B2C.

Como pode-se fazer quase tudo no mundo virtual (comprar, vender, trocar informações, visualizar relatórios), a possibilidade de gerenciar informações correlatas a estes contratos a distância através de acesso a banco de dados via *Web*, está se tornando uma realidade. Uma das tecnologias atualmente disponíveis são as de programação via Internet, que fazem com que o *Hiper Text Markup Language* (HTML), antes inflexível, transforme-se em algo mutante e totalmente dinâmico. O *Personal Home Page Tools* (PHP), que é uma tecnologia de *scripts* que rodam no servidor, é *Server-side*, ou seja, sua interpretação é feita

integralmente pelo servidor, sendo apenas a resposta (*output*) enviada ao cliente. Assim a empresa difere das outras, estando disponível para o cliente durante 24 horas por dia, sete dias por semana, pois com o *site*, o cliente pode se comunicar a qualquer hora, fazer compras, pesquisar preços e ofertas.

Na Internet, os dados são checados no banco de dados imediatamente e retornados ao usuário. Este por sua vez, quer encontrar algo que só a Internet possa oferecer, como por exemplo, a comodidade de fazer suas compras na hora em que desejar. Os recursos devem ser convenientes ao usuário, facilitar a sua vida, ou seja, devem prestar-lhe serviço e não dar-lhe um serviço. Conforme Strange (2000), segurança, agilidade e facilidade são alguns dos benefícios que são oferecidos através do acesso a banco de dados via *Web*.

Conforme Hartman (2000), um *site* freqüentado é uma calçada movimentada e com baixo custo. A empresa deve movimentar seu espaço na rede com o máximo possível de informações a respeito de sua atividade no mercado. Outros dados a serem considerados é que a maioria das análises registra que, tanto em vendas pela rede quanto em serviços e publicidade digital, os números brasileiros deverão continuar dobrando.

A Internet pode ser importante também durante as várias fases do processo de marketing. De forma similar à propaganda tradicional, a Internet pode alcançar objetivos de marketing como o aumento do conhecimento da empresa e da marca de um produto. Ao analisar o processo de venda, torna-se claro que a Internet pode visar muitas coisas além da simples venda. Milhares de clientes potenciais podem ser contatados e conduzidos até o fechamento de negócios, pelas diversas fases do processo de venda, do estímulo ao interesse até o desejo e a ação (Gonçalves, 2001).

Face a estas necessidades, uma empresa de materiais de construção, para não perder seu espaço, e sim aumentar as possibilidades de oferecer mais a seus clientes, resolveu investir neste mercado. Mesmo sendo difícil de avaliar com precisão o alcance desta alternativa, que é a possibilidade de o cliente entrar no site da empresa e realizar uma compra que o satisfaça em vários sentidos, a gerência vislumbra no acesso das informações um grande ganho de cunho estratégico para o negócio.

Em entrevista com um dos sócios da empresa Zago e Cia. Ltda., no município de Lages (SC), concretizou-se a idéia de desenvolver um aplicativo para controle e vendas da loja através da Internet. Os sócios acharam uma grande idéia saltar na frente da concorrência disponibilizando este serviço aos clientes, sendo que esta prestação de serviço via Internet, visa também o relacionamento com o cliente, e quando fala-se nessa estratégia, a tecnologia dever ter a função viabilizadora do projeto. Esta evolução requer uma gestão empresarial que ajuste os processos internos da empresa, integrando-os ainda mais através do computador, disponibilizando informações de forma transparente para tomadas de decisões rápidas e seguras. Com isto, há uma diminuição do ciclo operacional e a fidelização dos clientes, que sentem mais eficiência e credibilidade no processo.

1.1 OBJETIVOS DO TRABALHO

O objetivo principal deste trabalho é a implementação de um *site* de comércio eletrônico utilizando PHP e MySQL para controle de vendas com acesso ao banco de dados via *Web*, para uma loja de materiais de construção.

Os objetivos secundários do trabalho são:

- a) utilização da linguagem PHP, através da construção de um aplicativo utilizando o banco de dados MySQL, com a conversão da base de dados em *Cobol*;
- b) disponibilizar compras de qualquer produto da empresa durante 24 horas;
- c) manter um relacionamento mais próximo com o cliente através do site e correio eletrônico;
- d) aproveitar não somente para controle interno, mas também para B2C, a infraestrutura que a empresa oferece;

1.2 ESTRUTURA DO TRABALHO

No primeiro capítulo deste trabalho tem-se a introdução, apresentando os objetivos e a organização do mesmo.

No segundo capítulo, estão descritas informações sobre comércio eletrônico, seu crescimento e perspectivas.

No terceiro capítulo, são apresentadas as características da ferramenta PWS, utilizada no protótipo para integração com *Windows 98*.

O quarto capítulo deste trabalho é sobre a linguagem PHP, sua estrutura, suas variáveis, operações, funções e resultado na *Web*.

No quinto capítulo são descritas as ferramentas utilizadas no desenvolvimento do aplicativo, como conexão a um banco de dados MySQL.

No sexto capítulo é descrito o desenvolvimento do aplicativo.

No capítulo 7 descrevem-se sugestões e principais conclusões do trabalho.

2 COMÉRCIO ELETRÔNICO

Conforme Hartman (2000), a Internet está conduzindo uma economia eletronicamente processada que cria oportunidades para países, empresas e indivíduos em todo o mundo, como jamais se registrou na história. Observando-se a crescente utilização da Internet em todo o mundo, o grande interesse das organizações em fazerem um uso produtivo e adequado de seus recursos como ferramenta de marketing e comunicação, verifica-se a importância do tema para a sociedade.

O crescimento do uso da Internet no Brasil e no mundo tem despertado interesse de organizações e grupos de pesquisa, em torno das oportunidades e facilidades que se sucedem. Pode-se observar hoje no comércio eletrônico, exemplos de sucesso milionário (Amazon Books, Cisco e Yahoo), bem como ações de marketing e vendas que trazem pouco resultado, gerando assim dentro do mesmo campo de atuação um contraste de grande envergadura (Gonçalves, 2001). Apesar destas tendências impressionantes, a revolução digital está só começando. Este crescimento pode influenciar outros setores da economia, a medida que o número de pessoas que utilizam a Internet aumenta juntamente com o seu uso comercial. Empresas parceiras começaram a utilizar a Internet para trocarem informações a alguns anos. Estes usuários reportaram significantes ganhos de produtividade pelo uso da rede para criar, vender distribuir, comprar e prestar serviços via Internet. Pelo ano de 2002, a Internet deve movimentar 300 bilhões em negócios entre empresas (Gonçalves, 2001).

A Internet pode ser importante, também, durante as várias fases do processo de marketing. De forma similar à propaganda tradicional, a Internet pode alcançar objetivos de marketing como aumento do conhecimento da empresa e da marca de um produto. Ao analisar o processo de venda, torna-se claro que a Internet pode visar muitas coisas além da simples venda. Milhares de clientes potenciais podem ser contatados e conduzidos até o fechamento de negócios, pelas diversas fases do processo de venda - do estímulo ao interesse até o desejo e a ação. É a imagem do negócio que fará a diferença. Na hora de se fazer conhecido, a Internet dá recursos para que o pequeno criativo consiga se promover, fazendo muito barulho com pouco recurso. Cabe ao empreendedor aprender a utilizar esses recursos. A estrutura básica do sistema de comércio eletrônico inclui: desenvolvimento do sistema do *Web store*, estrutura de navegação para catálogo de produtos, funções de *check out* para

cadastramento e garantia de transação segura no pagamento, ferramentas de apoio como pesquisas, históricos de pedidos e ofertas, além de serviço de atendimento a clientes.

Entre as várias formas de transações através do *e-commerce*, tem-se o B2B e o B2C.

2.1 BUSINESS-TO-BUSINESS (B2B)

O B2B, é uma transação entre empresas, com objetivo de aumentar a eficiência da cadeia produtiva, com isso ocasionando uma troca de informações estratégicas e não podendo ser disponibilizadas livremente. O relacionamento entre as empresas em geral, dá-se na forma de que uma é a empresa fornecedora de matéria prima e a outra a consumidora, e que ambas através da Internet se comunicam com suas áreas de suprimentos e vendas dinamicamente.

2.2 BUSINESS-TO-CONSUMER (B2C)

Embora não esteja claro que a *Web* seja realmente um bom canal de vendas, e portanto ainda não é encarada prioritariamente pelas organizações, é indiscutível que o comércio eletrônico estreita o canal de relacionamento com os clientes. Além disso, o público que compra pela rede é basicamente classes A e B, com gastos médios (*tickets* médios) bem acima da média das lojas físicas.

O B2C pode não ser atraente hoje, mas com certeza terá um papel importante como serviço adicional, criando e estreitando relacionamentos com bons clientes e aumentando a conveniência do processo de compras. E entrar neste negócio hoje apresenta menos riscos, uma vez que as lições dos pioneiros podem ser apreendidas. Uma das lições principais é ter uma estratégia bem definida de comércio eletrônico e criar condições para que a loja virtual realmente proporcione uma experiência de compra prazerosa e bem sucedida aos seus clientes. Conhecer o cliente e o por que estão indo fazer compras na loja virtual é condição essencial para uma operação bem sucedida de B2C.

Na prática o cliente entra em uma determinada página porque quer fazer uma compra. Ele quer apenas duas alternativas: clicar no produto para comprar ou clicar no botão de saída ou retorno, porque não se interessou pelo produto.

3 FERRAMENTAS

3.1 PERSONAL WEB SERVER (PWS)

O *Microsoft Personal Web Server* (PWS) é um servidor *Web* de área de trabalho que permite publicar *home-pages* e compartilhar documentos na rede corporativa diretamente do próprio computador. O PWS é como uma plataforma de desenvolvimento em etapas antes de carregar o *site* num provedor Internet. O servidor *Web* de um *site* da Internet, coloca os documentos à disposição dos visitantes da Internet, através do navegador *Web* da sua Intranet corporativa (Microsoft Corporation, 1998).

Segundo Microsoft (1998), o PWS resolve problemas de compartilhamento convertendo documentos em HTML e publicando-os num *site* da *Web*. Ele possui uma grande plataforma para testar o *site* antes de colocá-lo no servidor da empresa ou num provedor de serviços de Internet. E uma das vantagens do PWS é que o *site* assim que estiver pronto, pode ser migrado para outros servidores sem problemas de incompatibilidade como por exemplo, o *Internet Information Server* (IIS) do *Windows NT* e o *Apache* para *Linux* ou *Windows*.

O PWS é totalmente gratuito e está incluído no cd do *Windows 98*. Os usuários de *Windows 95/98/98SE/ME/NT* podem compartilhar conteúdo de *Web* agora tão facilmente quanto eles compartilham pastas e arquivos em uma rede. Com isto se torna fácil compartilhar arquivos HTML e FTP nas Intranets e na Internet tanto como é compartilhar arquivos e impressoras em uma rede.

Para adicionar diretórios virtuais, o PWS cria um diretório onde pode ser colocado qualquer arquivo a ser publicado em um *Web site*. Também podem ser criados diretórios adicionais para *scripts*, por exemplo, para publicar seu conteúdo.

3.1.1 PERMISSÕES DE ACESSO AOS DIRETÓRIOS

As permissões de acesso de diretório são: “Lido”, “Executar” e “Scripts”.

No “Lido”, se um cliente envia um pedido para um arquivo que está em um diretório sem permissão de leitura, o servidor devolve uma mensagem de erro. O “Lido” permite que clientes da *Web* tenham acesso a ler e baixar arquivos armazenados no diretório virtual.

”Executar” permite que qualquer aplicação ou script seja executado neste diretório. Por questões de segurança, não é aconselhável ativar esta permissão para um diretório.

“Scripts” permite que *scripts* sejam executados sem a necessidade de que a permissão “Executar” esteja setada. É utilizada para diretórios que contenham *scripts* PHP, ASP, IDB, entre outros. Esta permissão é mais segura que a “Executar” porque pode limitar as aplicações que podem ser executadas neste diretório.

3.1.2 BUSCA NO DIRETÓRIO

Permitir busca no diretório é um modo alternativo de navegação para o *site*. Ao invés da *home-page* ter *links* para arquivos no diretório *home*, o documento padrão é desabilitado e então é apresentada uma lista dos arquivos disponíveis no diretório. Esta função é útil para quem quer compartilhar um grande número de documentos no seu formato nativo, o ASP.

Se a opção de uso automático de documentos padrão for desabilitada, e a opção de busca no diretório estiver habilitada, os visitantes podem ver a lista de todos arquivos e diretórios especificados na sua requisição.

3.1.3 LOG DE ATIVIDADE DO SITE DA WEB

Pode ser criado um arquivo de *log* que lista todos visitantes que acessaram o *site*, além dos arquivos que foram acessados enquanto eles estavam nele. Os arquivos de *log* são escritos no formato NCSA, e podem ser vistos em qualquer editor de texto. Novos arquivos são criados mensalmente e ficam armazenados no diretório “Logfiles/W3SPC1”, diretório do *Windows System*. O nome dos arquivos começa com Nc e são seguidos pelo ano, mês e dia em que foi criado o arquivo. Por exemplo, um arquivo criado em setembro de 2001, pode-se chamar “Nc092001.log”.

3.1.4 DESEMPENHO DO SITE

O acompanhamento do desempenho do *site* pode revelar a performance além de descobrir possíveis problemas de gargalo. O PWS oferece uma variedade de modos para se ver o desempenho do *site*. O gráfico de monitoramento de performance e também as estatísticas podem ser vistos no próprio painel do PWS.

Os dados monitorados são: conexões ativas, data de início do serviço, visitantes, requisições, bytes servidos e conexões concorrentes. As estatísticas disponíveis são: requisições por dia, requisições por hora, visitantes por dia e visitantes por hora.

As conexões ativas especificam quantas conexões estão transferindo dados no momento. A data de início é o tempo que o servidor está conectado desde a última vez que o serviço foi iniciado. Visitantes, é o número de endereços diferentes que se conectaram ao servidor desde que o serviço foi iniciado pela última vez. Requisições, é o número total de requisições recebidas pelo servidor desde a última vez que foi iniciado. Bytes servidos é a quantia total de dados enviados pelo servidor desde a sua última inicialização. E conexões concorrentes é o número de conexões iniciadas simultaneamente.

3.2 A LINGUAGEM PHP

Personal Home Page Tools – PHP é uma linguagem que permite criar *sites Web* dinâmicos, possibilitando uma interação com o usuário através de formulários, parâmetros, parâmetros da URL e *links* (Vivas, 2001). É uma linguagem em forma de script que interage no lado do servidor para a criação de páginas dinâmicas na *Web* (Anselmo, 2000). Uma linguagem *script*, pode ser entendida como uma página que é interpretada pelo servidor, devolvendo um formato HTML de acordo com a requisição de um *browser* cliente. Este *script* funciona como um direcionador de saída que irá exibir ao cliente uma página HTML.

Uma das vantagens do PHP, é que o código é executado no servidor, sendo enviado ao cliente apenas HTML puro. Assim então interagir com o banco de dados e aplicações existentes no servidor, não expando o código fonte para o cliente. Outra vantagem é que também é possível utilizar outras linguagens de *script*, como o *Javascript*, dentro do código PHP. O PHP também tem como uma das características mais importantes o suporte a um grande número de bancos de dados. E com o PHP, torna-se uma tarefa simples a construção

de uma página baseada em um banco de dados. Além disso, o PHP tem suporte a outros serviços através de protocolos como IMAP, SNMP, NMTP, HTTP e POP3. Ainda é possível abrir *sockets* e interagir com outros protocolos.

A linguagem PHP foi concebida durante o outono de 1994 por Rasmus Lerdorf. As primeiras versões foram utilizadas em sua *home-page* apenas para que ele pudesse ter informações sobre as visitas que estavam sendo feitas. A primeira versão utilizada por outras pessoas foi disponibilizada em 1995, e ficou conhecida como “*Personal Home Page Tools*” (ferramentas para página pessoal). Era composta por um sistema bastante simples que interpretava algumas macros e alguns utilitários que rodavam “por trás” das *home-pages*: um livro de visitas, um contador e algumas outras coisas.

Em meados de 1995 o interpretador foi reescrito, e ganhou o nome de PHP/FI, o “FI” veio de um outro pacote escrito por Rasmus que interpretava dados de formulários HTML (*Form Interpreter*). Ele combinou os scripts do pacote *Personal Home Page Tools* com o FI e adicionou suporte ao banco de dados *mSQL*, nascendo assim , o PHP/FI.

Por volta de 1997 houve uma mudança no desenvolvimento do PHP. Ele deixou de ser um projeto de Rasmus com contribuições de outras pessoas para ter uma equipe de desenvolvimento mais organizada. Conforme Valda (2000), o interpretador foi escrito por Zeev Suraski e Andi Gutmans. Este interpretador acabou sendo a base para a versão 3.

3.2.1 ESTRUTURA DE UM PROGRAMA PHP

Um script PHP possui basicamente esta estrutura:

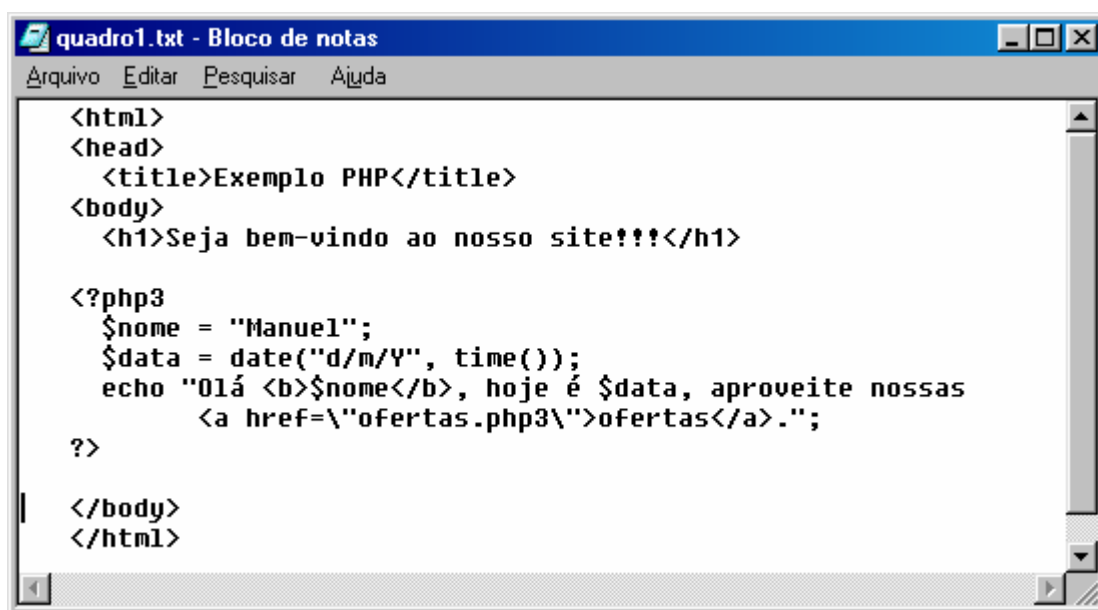
Comandos HTML – são enviados ao *browser* do cliente sem qualquer formatação pelo PHP. Nesta parte pode-se colocar quaisquer comandos HTML e outras informações úteis para a melhor apresentação da página;

Código PHP – sempre começa com `<?php`, ou simplesmente `<?`, podendo colocar vários scripts PHP em uma página e concatena-los com *tags* HTML. Para fechar o script utiliza-se `?>`, podendo dentro destes atribuir valores a variáveis, chamar funções, acessar o banco de dados, formatar resultados entre outros recursos do PHP. Ao fim de cada linha de

comando utiliza-se o “;” (ponto-e-vírgula). Caso não coloca-lo, o PHP retorna um erro em tempo de execução.

Assim pode-se misturar HTML e PHP, pois desta forma não é preciso se preocupar com a formatação dos dados de saída. Basta coloca-los no formato HTML e inserir o código PHP para gerar os dados dinâmicos. (Quadro 1).

Quadro 1 – Exemplo de estrutura do programa PHP



```
<html>
<head>
  <title>Exemplo PHP</title>
</head>
<body>
  <h1>Seja bem-vindo ao nosso site!!!</h1>

  <?php3
    $nome = "Manuel";
    $data = date("d/m/Y", time());
    echo "Olá <b>$nome</b>, hoje é $data, aproveite nossas
      <a href=\"ofertas.php3\">ofertas</a>.";
  ?>

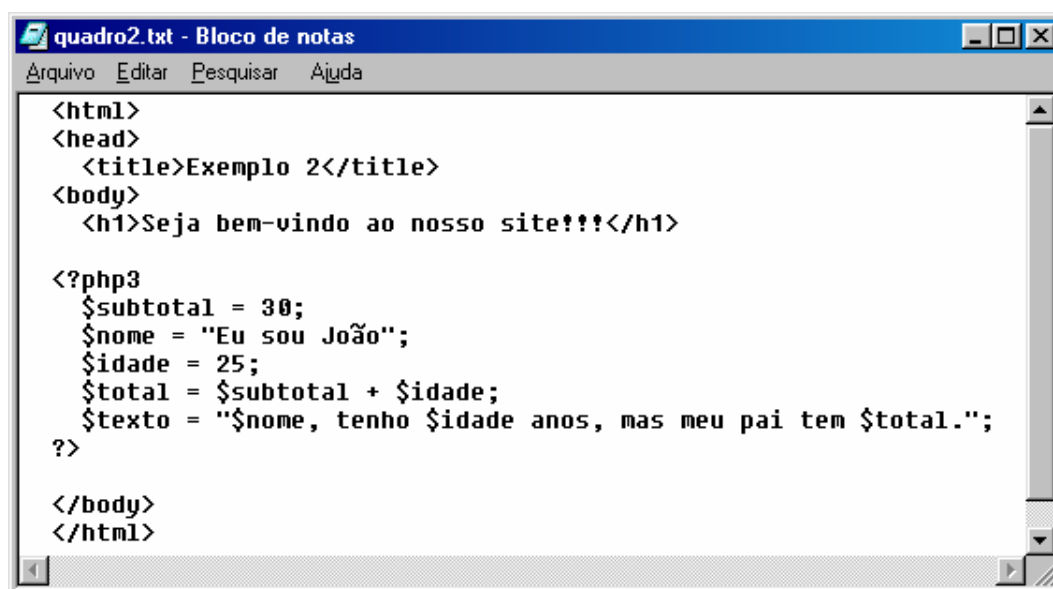
</body>
</html>
```

Como mostra o Quadro 1, existem comandos PHP e tags HTML combinados. É este um dos motivos que faz com que o PHP seja uma ferramenta de autoria em *Web*, pois o valor atribuído a uma variável continua válido até o término do processamento desta página. Desta forma pode-se, ler o banco de dados e armazenar os valores encontrados em variáveis do PHP, liberar o *buffer* e continuar usando os valores por toda a página do PHP.

3.2.2 RESULTADO NA WEB

Assim que o *script* PHP estiver escrito, ele já pode ser exibido na *Web*. Para este apresentar algo no *browser* é necessário que haja comandos “*echo*” e “*printf*”, caso contrário não exibirão nenhuma informação ao usuário (Quadro 2).

Quadro 2 – Exemplo 1 de resultado do programa PHP



```
quadro2.txt - Bloco de notas
Arquivo Editar Pesquisar Ajuda
<html>
<head>
  <title>Exemplo 2</title>
<body>
  <h1>Seja bem-vindo ao nosso site!!!</h1>

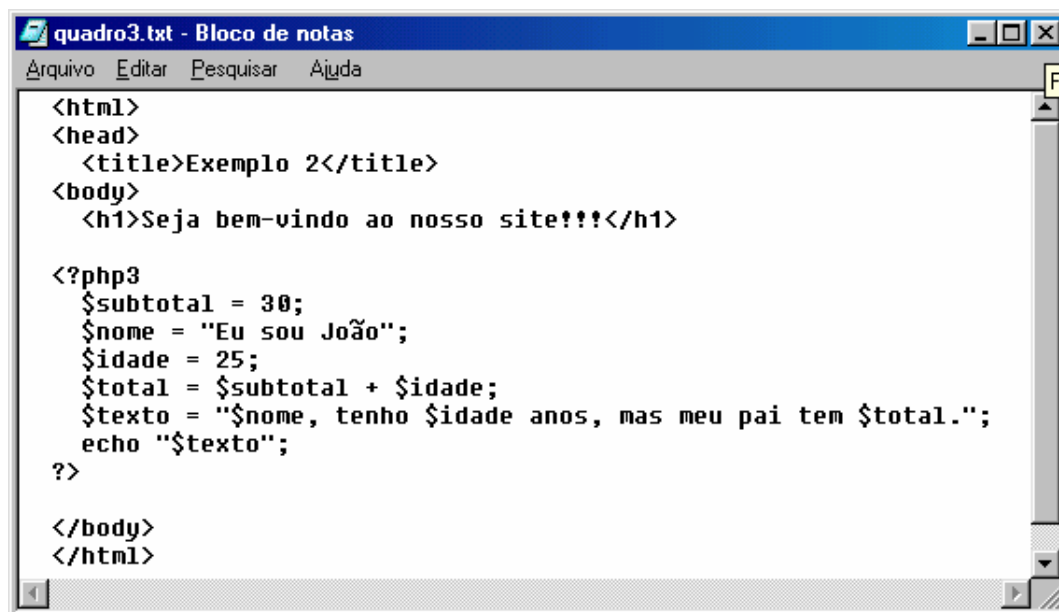
  <?php3
    $subtotal = 30;
    $nome = "Eu sou João";
    $idade = 25;
    $total = $subtotal + $idade;
    $texto = "$nome, tenho $idade anos, mas meu pai tem $total.";
  ?>

</body>
</html>
```

O programa no Quadro 2 vai mostrar no *browser*, simplesmente uma tela em branco, pois não existe qualquer linha de comando de saída. O *script* foi processado e o PHP armazenou em memória os valores das variáveis declaradas, mas esses valores foram perdidos quando o resultado foi enviado ao *browser* do usuário.

Alterando um pouco o programa anterior, inserindo um comando de exibição após a definição das variáveis (Quadro 3), obtém-se uma página com as informações apresentadas no Quadro 4.

Quadro 3 – Exemplo 2 de resultado do programa PHP

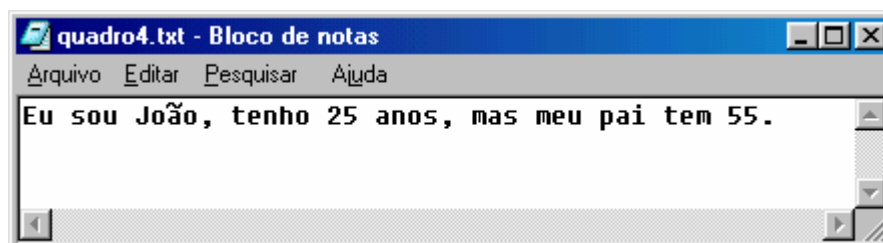


```
<html>
<head>
  <title>Exemplo 2</title>
<body>
  <h1>Seja bem-vindo ao nosso site!!!</h1>

  <?php3
    $subtotal = 30;
    $nome = "Eu sou João";
    $idade = 25;
    $total = $subtotal + $idade;
    $texto = "$nome, tenho $idade anos, mas meu pai tem $total.";
    echo "$texto";
  ?>

</body>
</html>
```

Quadro 4 – Exemplo 3 de resultado do programa PHP



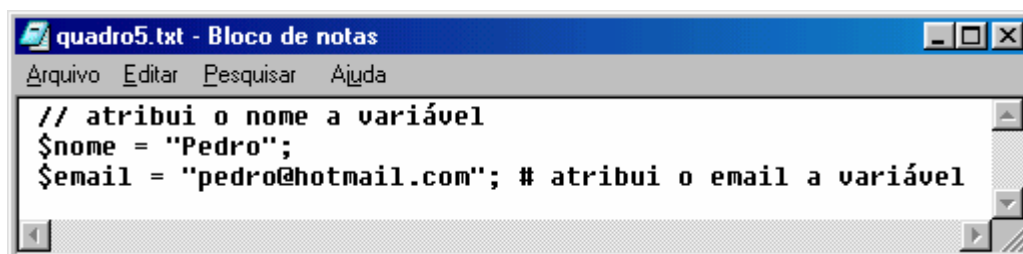
```
Eu sou João, tenho 25 anos, mas meu pai tem 55.
```

A menos que não se precise mostrar nada ao usuário, é imprescindível que se utilize os comandos de saída do PHP.

3.2.3 COMENTÁRIOS NO PHP

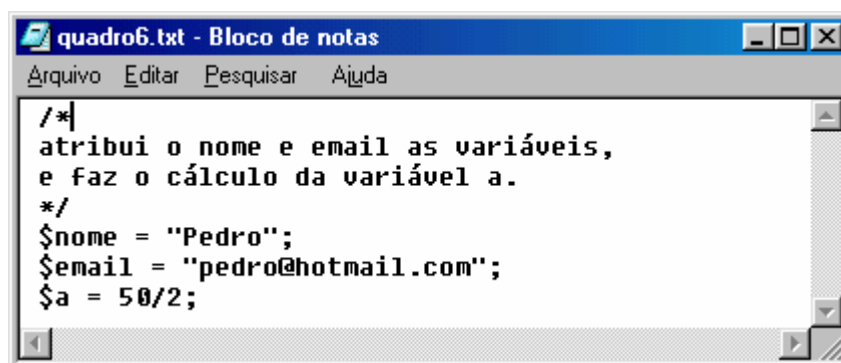
Todo programa deve possuir comentários, visando o entendimento do código em consultas posteriores. No PHP, existem três tipos de marcadores de comentário, que são: // # e /* */ para comentário de uma linha, conforme os Quadro 5 e Quadro 6.

Quadro 5 – Exemplo 1 de comentários no PHP



```
// atribui o nome a variável
$nome = "Pedro";
$email = "pedro@hotmail.com"; # atribui o email a variável
```

Quadro 6 – Exemplo 2 de comentários no PHP

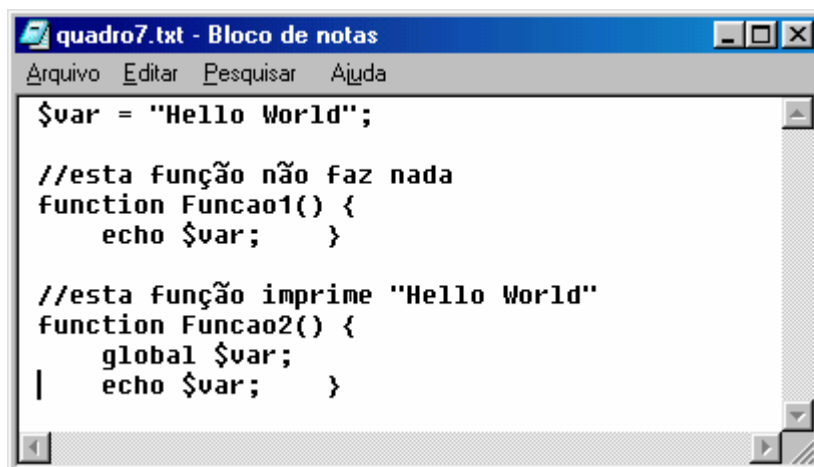


```
/*
atribui o nome e email as variáveis,
e faz o cálculo da variável a.
*/
$nome = "Pedro";
$email = "pedro@hotmail.com";
$a = 50/2;
```

3.2.4 VARIÁVEIS NO PHP

No PHP as variáveis podem ser globais, locais e estáticas. As variáveis globais são declaradas em qualquer parte do código PHP e podem ser usadas em qualquer parte desse código a partir de sua declaração. Para se usar variáveis globais dentro de uma função, elas devem ser declaradas, dentro da mesma função, utilizando-se a palavra global conforme o Quadro 7.

Quadro 7 – Variáveis no PHP – Exemplo 1



```
quadro7.txt - Bloco de notas
Arquivo  Editar  Pesquisar  Ajuda

$var = "Hello World";

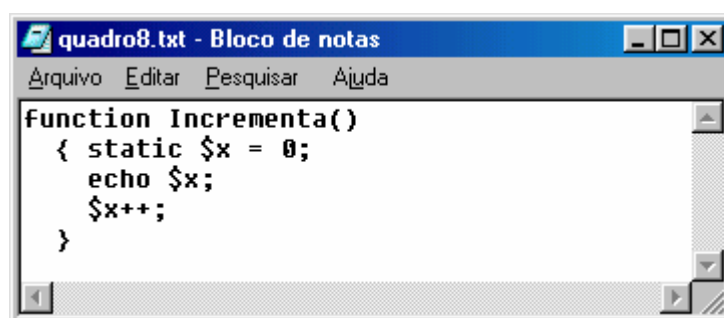
//esta função não faz nada
function Funcao1() {
    echo $var;    }

//esta função imprime "Hello World"
function Funcao2() {
    global $var;
    |   echo $var;    }
```

Fonte: Fischer (2000)

As variáveis locais são aquelas declaradas dentro de uma função e que fazem parte somente da mesma. As variáveis estáticas são uma adaptação das variáveis locais para que o valor atribuído a elas dentro da função não seja perdido quando a mesma função terminar conforme o Quadro 8.

Quadro 8 – Variáveis no PHP – Exemplo 2



```
quadro8.txt - Bloco de notas
Arquivo  Editar  Pesquisar  Ajuda

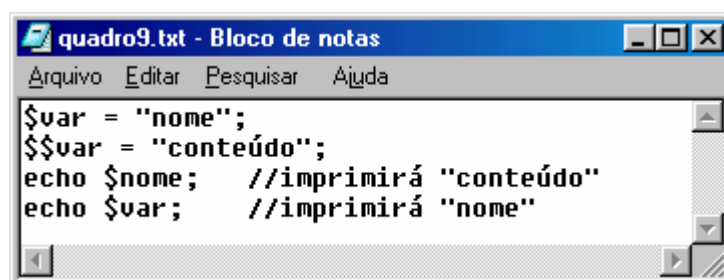
function Incrementa()
{ static $x = 0;
  echo $x;
  $x++;
}
```

Fonte: Fischer (2000)

No exemplo do Quadro 8, cada vez que a função `Incrementa()` for executada, a variável estática `$x` será impressa e depois incrementada, sem que, ao término da função, seu valor seja perdido. Conforme Fischer (2000), as variáveis estáticas são essenciais quando usadas em funções recursivas.

O PHP também dispõe de um recurso muito interessante que é a possibilidade de se mudar o nome de um variável em qualquer parte do código. Para isto, utiliza-se o duplo cifrão (`$$`), conforme mostra o Quadro 9.

Quadro 9 – Variáveis no PHP – Exemplo 3



```
quadro9.txt - Bloco de notas
Arquivo Editar Pesquisar Ajuda
$var = "nome";
$$var = "conteúdo";
echo $nome; //imprimirá "conteúdo"
echo $var; //imprimirá "nome"
```

Fonte: Fischer (2000)

O PHP possui diversas variáveis de ambiente, como a `$PHP_SELF`, por exemplo, que contém o nome e o *path* do próprio arquivo. Algumas outras contém informações sobre o navegador do usuário, o servidor HTTP, a versão do PHP entre outras. Para se obter uma listagem de todas as variáveis e constantes de ambiente, e seus respectivos conteúdos, deve-se utilizar a função `phpinfo()`.

A função que verifica o tipo de uma variável é usada em virtude da tipagem dinâmica utilizada pelo PHP. Nem sempre é possível saber qual o tipo de uma variável em determinado instante se não contar com a ajuda de algumas funções que ajudam a verificar isso. Esta verificação pode ser feita de duas maneiras:

Função que retorna o tipo da variável: Esta função é a `gettype`, onde sua assinatura é a seguinte: `string gettype (mixed var)`. A palavra “*mixed*” indica que a variável *var* pode ser de

diversos tipos. Ela retorna os seguintes *strings*: “*integer*”, “*double*”, “*string*”, “*array*”, “*object*”, e “*unknow type*”;

Funções que testam o tipo da variável: são as funções *is_int*, *is_integer*, *is_real*, *is_long*, *is_float*, *is_string*, *is_array* e *is_object*. Todas tem o mesmo formato: *int is_integer (mixed var)*. Todas essas funções retornam *true* se a variável for daquele tipo, e *false* em caso contrário.

A destruição de uma variável torna-se possível desalocando uma variável se ela não for usada posteriormente através da função *unset*, que tem a seguinte assinatura: *int unset (mixed var)*.

A função destrói a variável, ou seja, libera a memória ocupada por ela, fazendo com que ela deixe de existir. Se posteriormente for feita uma chamada a variável, será criada uma nova variável de mesmo nome e de conteúdo vazio, a não ser que a chamada seja pela função *isset*. Se a operação for bem sucedida, retorna *true*.

Para se verificar se uma variável possui um valor, existem dois tipos de teste que podem ser feitos para verificar se uma variável está setada: com a função *isset* e com a função *empty*.

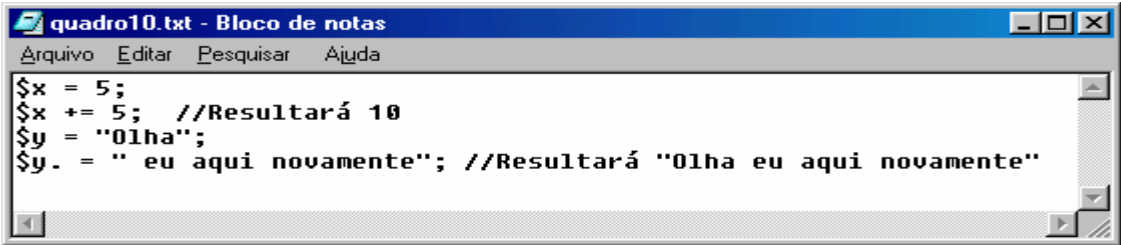
A função *isset*: *int isset (mixed var)*, retorna *true* se a variável estiver setada (ainda que com uma *string* vazia ou o valor zero), e *false* em caso contrário.

A função *empty*: *int empty (mixed var)*, retorna *true* se a variável não contiver um valor (não estiver setada) ou possuir valor 0 (zero) ou uma *string* vazia, caso contrário, retorna *false*.

3.2.5 OPERADORES DO PHP

O PHP possui vários tipos de operadores, são eles: operadores de atribuição, de *string*, aritméticos, de comparação e lógicos. O operador de atribuição padrão do PHP é o =, e com este pode-se combinar para criar outros atribuidores, conforme o Quadro 10. Já o operador de *string* é utilizado para concatenar vários *strings*, conforme o Quadro 11. E os outros operadores, aritméticos, de comparação e lógicos seguem o padrão das outras linguagens.

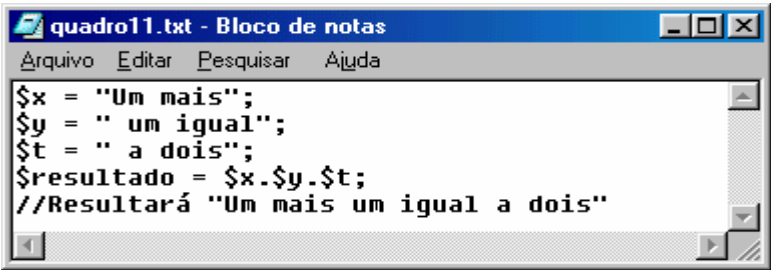
Quadro 10 – Operadores de atribuição



```
quadro10.txt - Bloco de notas
Arquivo  Editar  Pesquisar  Ajuda
$х = 5;
$х += 5; //Resultará 10
$у = "Olha";
$у. = " eu aqui novamente"; //Resultará "Olha eu aqui novamente"
```

Fonte: Stoco (2000)

Quadro 11 – Operador de string



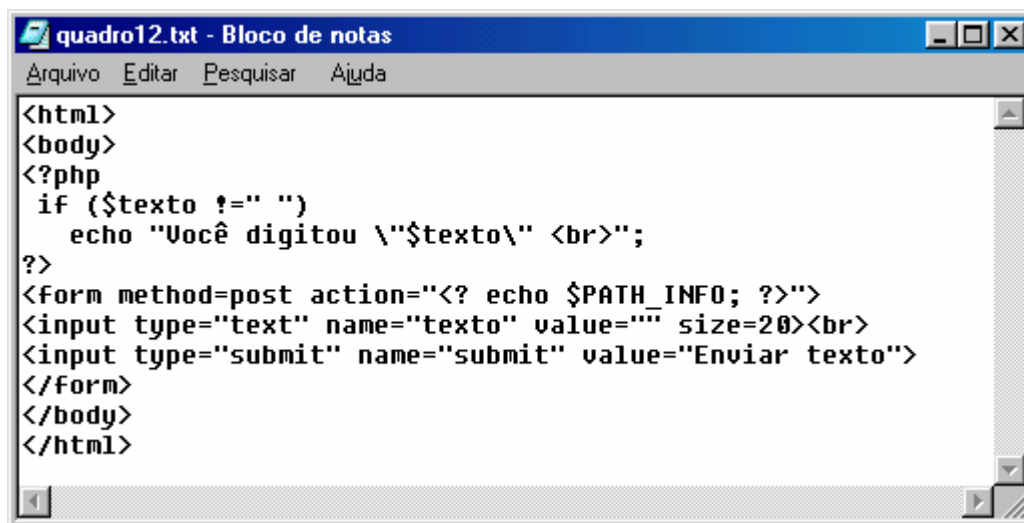
```
quadro11.txt - Bloco de notas
Arquivo  Editar  Pesquisar  Ajuda
$х = "Um mais";
$у = " um igual";
$t = " a dois";
$resultado = $х.$у.$t;
//Resultará "Um mais um igual a dois"
```

Fonte: Stoco (2000)

3.2.6 FORMULÁRIOS HTML

O *Hypertext Markup Language* (HTML) é uma linguagem de formatação usada para construir páginas na Internet. Ao clicar num botão “*Submit*” em um formulário HTML as informações dos campos serão enviados ao servidor especificado para que possa ser produzida uma resposta. O PHP trata esses valores como variáveis, cujo nome é o nome do campo definido no formulário, conforme o Quadro 12.

Quadro 12 – Formulários HTML



```
<html>
<body>
<?php
  if ($texto !=" ")
    echo "Você digitou \"\$texto\" <br>";
?>
<form method=post action="<? echo $PATH_INFO; ?>"
<input type="text" name="texto" value="" size=20><br>
<input type="submit" name="submit" value="Enviar texto">
</form>
</body>
</html>
```

Fonte: Barreto (2000)

Ao salvar o arquivo acima e carregá-lo no *browser*, o usuário verá apenas um formulário que contém um espaço para digitar o texto. Ao digitar um texto qualquer e submeter o formulário, a resposta, que é o mesmo arquivo PHP (indicado pela constante `$PATH_INFO`, que retorna o nome do arquivo).

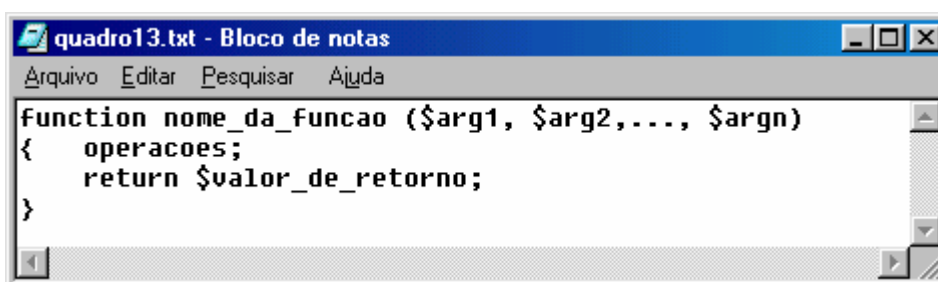
Isso ocorre porque o código PHP testa o conteúdo da variável `$texto`. Inicialmente ele é uma *string* vazia, e por isso nada é impresso na primeira parte. Quando algum texto é digitado no formulário e submetido, o PHP passa a tratá-lo como uma variável. Como no formulário o campo possui o nome “texto”, a variável com seu conteúdo será `$texto`. Assim no próximo teste o valor da variável será diferente de uma *string* vazia, e o PHP imprime um texto antes do formulário.

3.2.7 FUNÇÕES DO PHP

As funções devem ser declaradas antes de se fazer referência a elas. Sua sintaxe pode ser vista no Quadro 13. No PHP, as funções podem retornar inteiros, *strings*, *doubles*, *arrays* e objetos. O PHP pode passar argumentos tanto por valor (padrão) como por referência. O

PHP não suporta listas de argumentos de tamanho variável, mas um efeito similar pode ser obtido utilizando-se *arrays*, conforme mostra o Quadro 14.

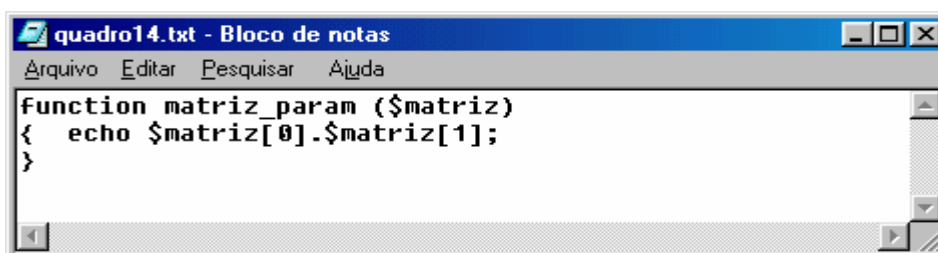
Quadro 13 – Funções no PHP – Exemplo 1



```
function nome_da_funcao ($arg1, $arg2,..., $argn)
{
    operacoes;
    return $valor_de_retorno;
}
```

Fonte: Fischer (2000)

Quadro 14 – Funções no PHP – Exemplo 2

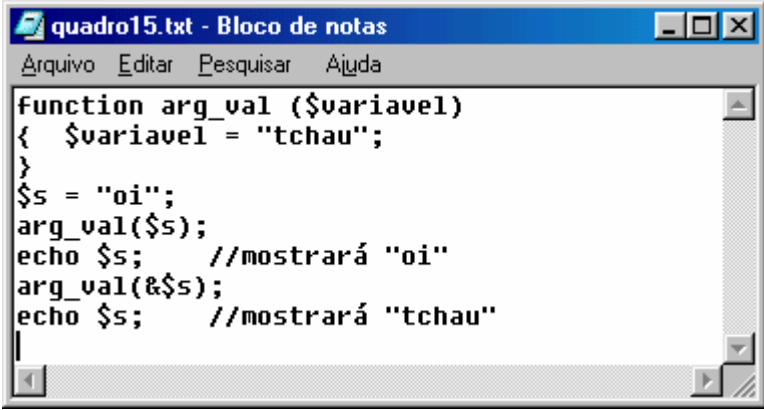


```
function matriz_param ($matriz)
{
    echo $matriz[0].$matriz[1];
}
```

Fonte: Fischer (2000)

Como padrão, o PHP passa parâmetros por valor, o que não possibilita que as alterações feitas pela função na variável permaneçam após o término da função. Para que as funções alterem o valor das variáveis definitivamente, usa-se a passagem de argumentos por referência, que consistem em colocar o sinal **&** antes do **\$**. Caso seja necessário passar um argumento por referência para uma função que utiliza argumentos por valor, utiliza-se o **&** também na passagem de parâmetros, conforme o Quadro 15.

Quadro 15 – Passagem de argumentos



```
function arg_val ($variavel)
{ $variavel = "tchau";
}
$s = "oi";
arg_val($s);
echo $s; //mostrará "oi"
arg_val(&$s);
echo $s; //mostrará "tchau"
```

Fonte: Fischer (2000)

Também é possível definir um valor padrão para os argumentos, caso esses não tenham sido especificados. O valor deve ser uma constante, não uma variável ou um membro de objeto, e todos os argumentos que terão um valor padrão deverão estar à direita dos outros argumentos sem valor padrão.

3.3 MYSQL

O *Structured Query Language* (SQL) é uma linguagem estruturada para manipulação de dados. É padronizada para os bancos de dados relacionais, mas cada gerenciador pode possuir uma extensão própria dessa linguagem (SQL Server, 2000). Quem vai executar propriamente os comandos como inserir, alterar, excluir, listar entre muitos outros, é o gerenciador.

O MySQL é um banco de dados compatível com a linguagem SQL *Ansi*. Ele oferece uma plataforma prática, rápida, eficiente, de fácil manuseio e extremamente confiável. Sua instalação pode ser obtida gratuitamente na Internet. Existem várias versões para as plataformas diferentes como: *Windows95/98*, *Windows NT*, *Linux*. O MySQL nasceu da insatisfação de algumas pessoas com o gerenciador de banco de dados *mSQL*. Essas pessoas desenvolveram novas interfaces com o banco de dados e chamaram-no de MySQL. Para gerenciar o MySQL, existe um utilitário chamado *mysql* que é instalado junto com o

gerenciador de banco de dados. Para executá-lo, basta digitar na linha de comando do *command.com*, o *mysql*, o qual abrirá o gerenciador do banco de dados. Mas há também ferramentas gráficas para *Windows*, que facilitam a interação com o banco de dados.

3.3.1 CONECTANDO UM BANCO DE DADOS

Para acessar a banco de dados é utilizado o gerenciador de banco de dados *mysql*. Para interagir com uma base de dados SQL existem três comandos básicos que devem ser utilizados: um que faz a conexão com o servidor de banco de dados, um que seleciona a base de dados a ser utilizada e um terceiro que executa uma cláusula SQL. A conexão com o servidor de banco de dados MySQL em PHP é feita através do comando *mysql_connect*, que tem a seguinte sintaxe:

```
int mysql_connect (string [hostname], string [usuário], string [password]);
```

Retorno: um link conectado com o MySQL em caso de sucesso, ou *false* em caso de erro.

Os parâmetros usados são o endereço do servidor, o nome do usuário e a senha para a conexão. Estabelece uma conexão a um servidor MySQL. Todos argumentos são opcionais, e se eles estiverem vazios, são assumidos os valores *default*. No caso de uma segunda ligação feita com os mesmos argumentos, não será estabelecida nenhuma ligação nova, mas ao invés disso será devolvido o *link_identifier* da ligação já aberta.

A ligação para o servidor será fechada assim que a execução do script termine, a menos que seja fechado mais cedo chamando a função *mysql_close*. Uma vez conectado, é preciso selecionar o banco de dados existente no servidor com o qual se deseja trabalhar. Isso é feito através da função *int mysql_select_db*, que possui a seguinte sintaxe:

```
int mysql_select_db (string [nome_db], int [link_identifier]);
```

Retorno: *true* em sucesso, *false* em erro.

Isto torna determinado banco de dados ativo no servidor atual associado com o *link_identifier* especificado. Se nenhum *link_identifier* for especificado, a última ligação aberta é assumida. Se nenhuma ligação estiver aberta, a função tentará estabelecer uma ligação com a função *mysql_connect*. Toda chamada subsequente para *mysql_query* será feita no banco de dados ativo.

A execução de cláusulas SQL, após estabelecida a conexão e selecionada a base de dados a ser utilizada, pode ser feita através de consultas escritas em SQL (*Structured Query Language*), com o comando *mysql_query*, que utiliza a seguinte sintaxe:

```
int mysql_query (string consulta, int link_identifier);
```

Retorno: *true* (não-zero) ou *false* indica se a *query* teve sucesso ou não.

Não indica nada sobre o número de linhas que são afetadas (por comando *insert*, *update* ou *delete*) nem devolvidas (por comando *select*). É perfeitamente possível uma *query* ter sucesso, mas não afetar ou devolver nenhuma linha.

O comando envia uma consulta para o banco de dados atualmente ativo no servidor associado com o *link_identifier* da ligação específica. Se *link_identifier* não for especificado, a última ligação aberta é assumida. Se nenhuma ligação estiver aberta, a função tenta estabelecer uma ligação com *mysql_connect*, que será chamado sem argumentos.

O tratamento de resultados de *query select* se dá quando ao executar uma *query SQL select* através do comando *mysql_query*, o identificador do resultado é armazenado numa variável que pode ser tratada de diversas formas. Uma das maneiras de fazê-lo é usar o comando *mysql_result*. O comando *mysql_result* tem a seguinte sintaxe:

```
int mysql_result (int result, int row, mixed [campo]);
```

Este comando retorna os conteúdos de uma única célula de um resultado MySQL. O argumento de campo pode ser o campo *offset*, o nome do campo, ou o nome do campo com o nome da tabela (*nomecampo.nometabela*). Se o nome do campo foi associado a um pseudônimo (*'select codcliente as seqüencial from...'*), utiliza-se este pseudônimo em vez do nome do campo. Ao se trabalhar com resultados muito grandes, você deve considerar a utilização de uma das funções que buscam uma linha inteira. Algumas destas funções podem devolver os conteúdos de células múltiplas em uma chamada, eles são mais rápidos que a função *mysql_result*. As chamadas a função *mysql_result* não devem ser misturadas com chamadas a outras funções que se tratam o resultado. Algumas das alternativas de alto-desempenho indicadas são: *mysql_fetch_row*, *mysql_fetch_array* e *mysql_fetch_object* (Anselmo, 2000).

A consulta aos dados contidos em uma tabela é feita usando o comando *select*. Este comando diz ao banco para separar, de todos os dados contidos em uma tabela, apenas aqueles que se precisa. Para saber quais pessoas, dentre as que preencheram o formulário, tem o primeiro nome Pedro, basta executar o seguinte comando:

```
Select * from cliente where nome like "Pedro";
```

No padrão SQL, pode-se informar como se quer a comparação do "parecido", especificando se o que se quer começa, termina ou contém o texto a ser comparado. Como se quer saber todo mundo cujo primeiro nome é Pedro, fica assim no SQL:

```
Select * from cliente where nome like "Pedro%";
```

4 DESENVOLVIMENTO DO TRABALHO

Com base em pesquisas e entrevistas realizadas com o sócio gerente da empresa Zago & Cia. Ltda situada na região de Lages (SC), analisou-se a possibilidade e a necessidade de desenvolver algo que chame mais clientes de uma maneira diferente e principalmente com baixo custo. Esta empresa atua no ramo de ferramentas, materiais de construção e madeiras. Com a possibilidade de abranger os negócios e a clientela da empresa, viu-se que a infraestrutura da empresa poderia não só servir para controle interno, mas também para vender pela Internet. Com isso, além de invadir uma área desconhecida para empresas do ramo e também pelas outras empresas da cidade, o aplicativo deveria oferecer os seus produtos durante 24 horas, boletins digitais enviados aos seus clientes com as ofertas da semana, visando facilitar a vida dos mesmos, além de manter um contato mais próximo entre empresa e cliente.

4.1 LEVANTAMENTO DE INFORMAÇÕES

Através dos contatos efetuados, foi visto que a empresa possui um sistema de propriedade da SISSUL que é licenciado para a empresa Zago & Cia. Ltda. Este sistema tem o controle da maioria dos processos existentes na empresa como: cadastro, alteração e exclusão de clientes, fornecedores, categorias e produtos. O sistema é do tipo *batch* e roda em ambiente *Windows*. O alvo principal deste aplicativo é a loja matriz da empresa, por ter uma grande variedade de produtos diferentes entre as três lojas, sendo necessário focar somente a loja principal neste primeiro momento. A empresa possui uma estrutura de cabeamento estruturado instalada na loja inteira, sendo que há um plug de rede em cada pilar da empresa, onde brevemente estarão ligados cerca de 25 micros, todos conectados a um switch, o qual se conecta ao servidor central IBM. No escritório existem cerca de 10 micros utilizados para os mais variados processos existentes na empresa como cadastro de novos produtos e clientes.

A empresa possui uma grande quantidade de itens cadastrados. Para este fato, analisou-se a possibilidade e a necessidade de migrar o banco de dados Cobol existente, com isto, evitando perda de tempo e mão-de-obra para o cadastramento manual de itens, clientes e sub-grupos (fornecedores).

Perante este fato, viu-se possível o desenvolvimento deste aplicativo que oferece ao cliente uma compra prazerosa a hora que ele quiser sem ter que esperar em filas ou ser mal atendido, mantendo também um relacionamento mais próximo entre cliente e empresa.

4.2 REQUISITOS PRINCIPAIS DO PROBLEMA A SER TRABALHADO

O aplicativo desenvolvido propõe o gerenciamento e controle estratégico dos dados de produtos, além de efetivar vendas através da *Internet*. No desenvolvimento do aplicativo foram utilizados, a linguagem de programação PHP, o banco de dados MySQL, a ferramenta CASE *Rational Rose* e o *Personal Web Server*. Tanto que para o PHP interagir com o servidor de dados PWS é necessário baixar o pacote do PHP e instalá-lo corretamente para que funcione sem problemas no servidor assim que se faz uma requisição.

O aplicativo será acessado via *browser*, através da *Internet*. As páginas PHP acessam o banco de dados MySQL e retornam ao cliente uma página HTML com o devido resultado. A tabela de produtos do banco de dados foi migrado do banco de dados *Cobol* existente na empresa. Para isto foram utilizadas as ferramentas *Microsoft Excel*, *Microsoft Access* e o software *Access-to-MySQL Demo* da *Intelligent Converters*.

Para a utilização do sistema na empresa, foi necessário utilizar um computador rodando um *browser*, e estar habilitado a ter acesso ao sistema com identificação e senha cadastrada. Somente o administrador do sistema poderá alterar documentos e cadastros em geral. Inclusive algumas das estatísticas são apresentadas somente na página do administrador. Já os funcionários possuem uma página própria para consultar produtos, clientes e efetuar vendas.

Os clientes que desejarem comprar necessitam ter um cadastro na empresa, ou então podem cadastrar-se assim que quiserem através do *site*. Já os clientes cadastrados e “*logados*”, podem ter acesso a todas outras opções disponíveis ao cliente que são: consultar preços e efetuar compras.

4.3 ESPECIFICAÇÃO

As características esperadas por este protótipo são:

- a) rapidez no atendimento do cliente;
- b) disponibilizar ao cliente acesso a consultas de preços e ofertas da semana através de correio eletrônico e do *site*;
- c) maior interação entre cliente e empresa;
- d) disponibilidade ao cliente de 24 horas diárias.

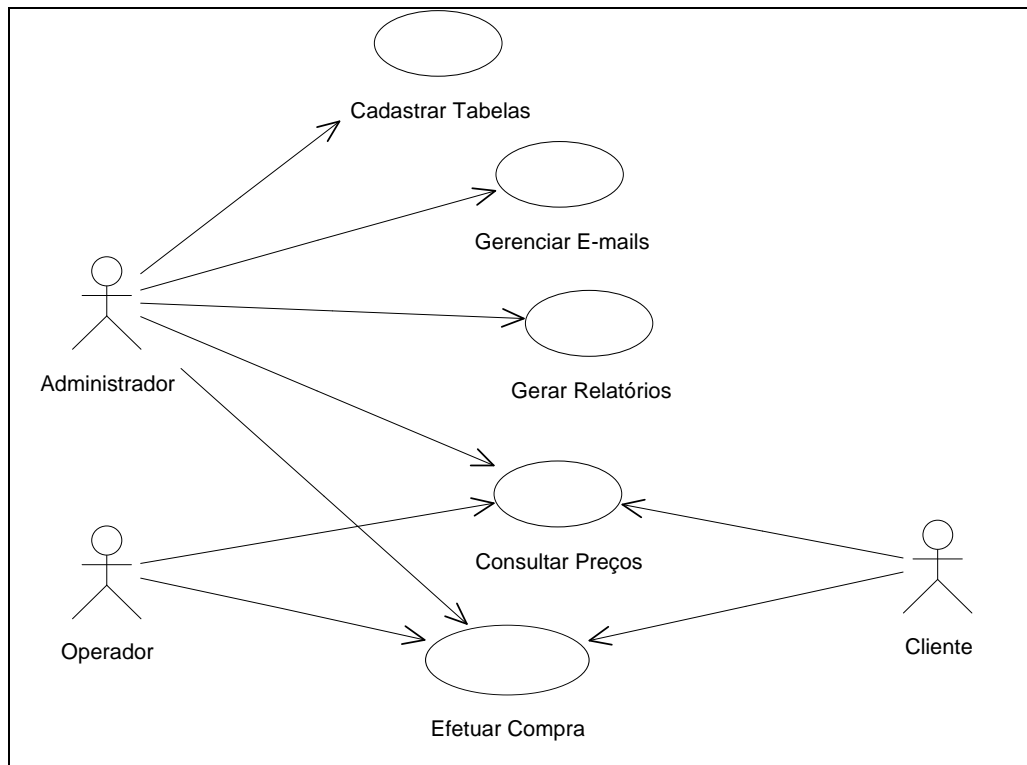
4.3.1 DIAGRAMAS DE CASO DE USO

Na Figura 1 apresenta-se o caso de uso primário do sistema. Os três atores são o cliente, o administrador e o operador. O cliente possui os processos “Consultar Preços” e “Efetuar Compra”.

Os processos do administrador são: “Cadastrar Tabelas” onde são efetuadas todas inserções, alterações e exclusões no banco de dados, e “Gerenciar E-mails” que é a parte e envio de e-mails promocionais, além do recebimento de e-mails dos clientes.

Já o operador possui os mesmos processos que o cliente: “Consultar Preços” onde ele possui um catálogo de produtos, além do processo “Efetuar Compra”, sendo que a única diferença é que para o operador estes processos são realizados na Intranet da empresa enquanto o cliente pode utilizar tanto a Intranet como a Internet.

Figura 1 – Diagrama de caso de uso primário do sistema



4.3.2 MER LÓGICO E FÍSICO

A seguir serão demonstrados os modelos de entidade e relacionamento lógico e físico conforme as Figura 2 e Figura 3.

Figura 2 – MER lógico

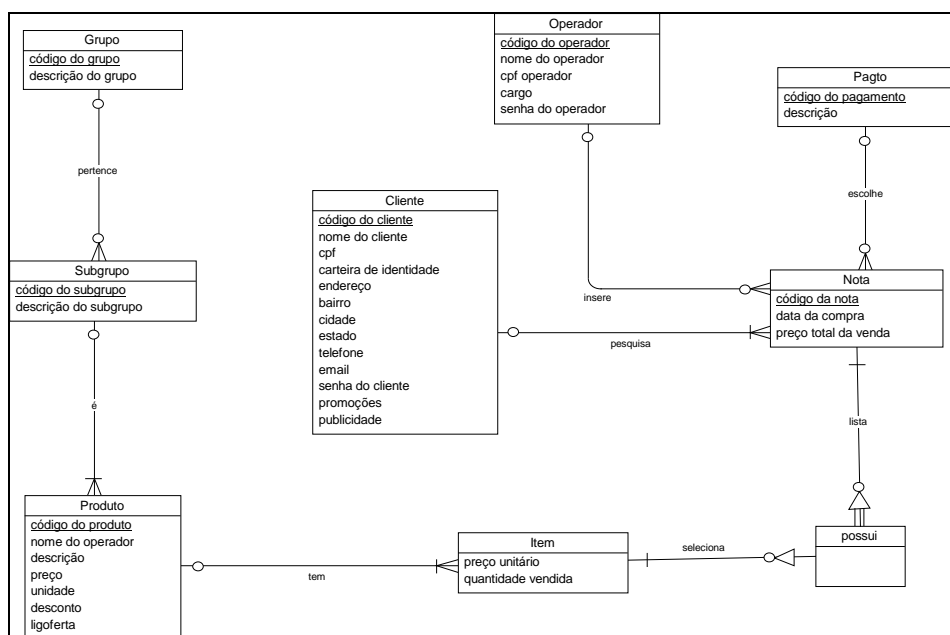
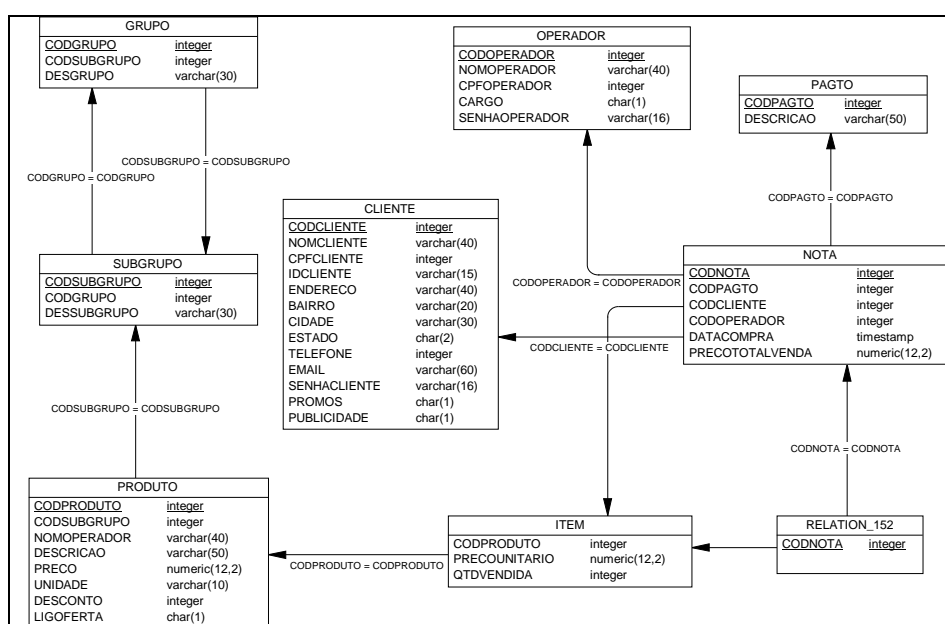


Figura 3 – MER físico



4.3.3 DICIONÁRIO DE DADOS

O dicionário de dados demonstra as tabelas com seus respectivos campos e o tipo de cada campo, conforme as Figura 4 e Figura 5.

Figura 4 – Dicionário de dados (parte 1)

Cliente				
Attribute List				
Name	Code	Type	P	M
código do cliente	CODCLIENTE	integer	Yes	Yes
nome do cliente	NOMCLIENTE	varchar(40)	No	No
cpf	CPFCLIENTE	integer	No	No
carteira de identidade	IDCLIENTE	varchar(15)	No	No
endereço	ENDERECO	varchar(40)	No	No
bairro	BAIRRO	varchar(20)	No	No
cidade	CIDADE	varchar(30)	No	No
estado	ESTADO	char(2)	No	No
telefone	TELEFONE	integer	No	No
email	EMAIL	varchar(60)	No	No
senha do cliente	SENHACLIENTE	varchar(16)	No	No
promoções	PROMOS	char(1)	No	No
publicidade	PUBLICIDADE	char(1)	No	No
Grupo				
Attribute List				
Name	Code	Type	P	M
código do grupo	CODGRUPO	integer	Yes	Yes
código do subgrupo	CODSUBGRUPO	integer	No	No
descrição do grupo	DESGRUPO	varchar(30)	No	No
Item				
Attribute List				
Name	Code	Type	P	M
código do produto	CODPRODUTO	integer	No	No
preço unitário	PRECOUNITARIO	numeric(12,2)	No	No
quantidade vendida	QTDVENDIDA	integer	No	No
Nota				
Attribute List				
Name	Code	Type	P	M
código da nota	CODNOTA	integer	Yes	Yes
código do pagamento	CODPAGTO	integer	No	No
código do cliente	CODCLIENTE	integer	No	No
código do operador	CODOPERADOR	integer	No	No
data da compra	DATACOMPRA	timestamp	No	No
preço total da venda	PRECOTOTALVENDA	numeric(12,2)	No	No

Figura 5 – Dicionário de dados (parte 2)

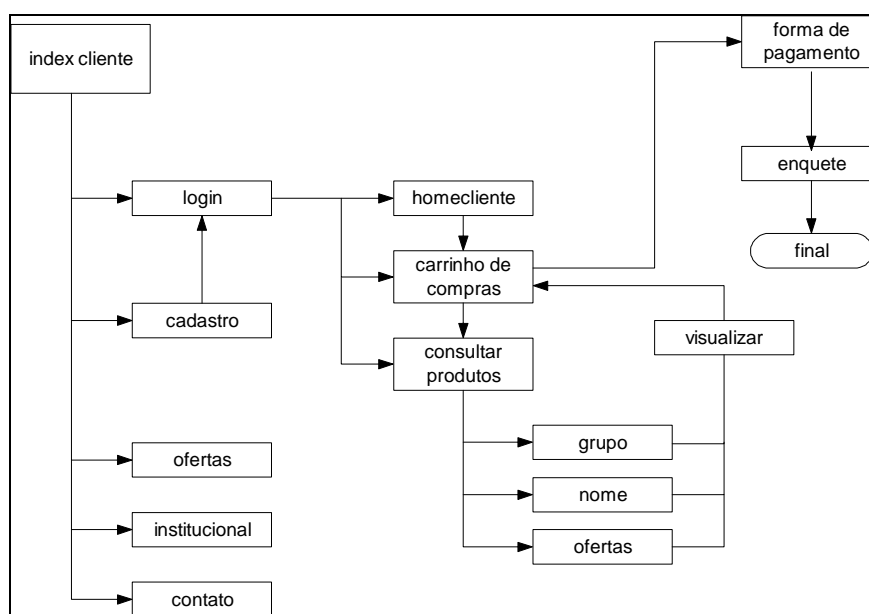
Operador				
Attribute List				
Name	Code	Type	P	M
código do operador	CODOPERADOR	integer	Yes	Yes
nome do operador	NOMOPERADOR	varchar(40)	No	No
cpf operador	CPFOPERADOR	integer	No	No
cargo	CARGO	char(1)	No	No
senha do operador	SENHAOPERADOR	varchar(16)	No	No
Pagto				
Attribute List				
Name	Code	Type	P	M
código do pagamento	CODPAGTO	integer	Yes	Yes
descrição	DESCRICA0	varchar(50)	No	No
possui				
Attribute List				
Name	Code	Type	P	M
código da nota	CODNOTA	integer	Yes	Yes
Produto				
Attribute List				
Name	Code	Type	P	M
código do produto	CODPRODUTO	integer	Yes	Yes
código do subgrupo	CODSUBGRUPO	integer	No	No
nome do operador	NOMOPERADOR	varchar(40)	No	No
descrição	DESCRICA0	varchar(50)	No	No
preço	PRECO	numeric(12,2)	No	No
unidade	UNIDADE	varchar(10)	No	No
desconto	DESCONTO	integer	No	No
ligoferta	LIGOFERTA	char(1)	No	No
Subgrupo				
Attribute List				
Name	Code	Type	P	M
código do subgrupo	CODSUBGRUPO	integer	Yes	Yes
código do grupo	CODGRUPO	integer	No	No
descrição do subgrupo	DESSUBGRUPO	varchar(30)	No	No

4.3.4 ESQUEMA NAVEGACIONAL DO SITE

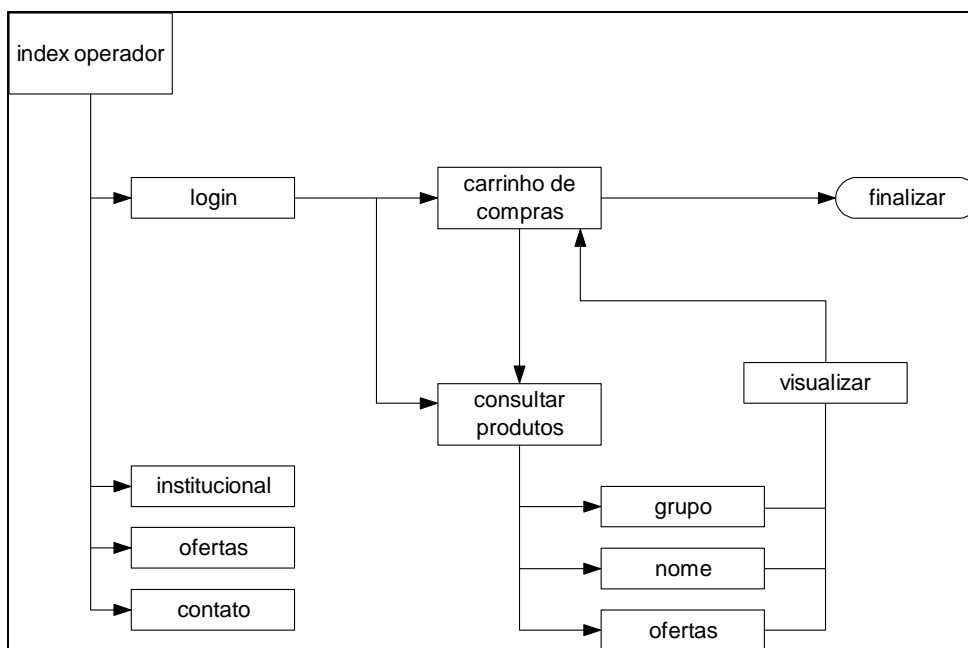
O esquema navegacional do *site* do cliente está descrito na Figura 6. É necessário que o cliente esteja cadastrado para poder utilizar todas as funções disponíveis ao cliente. Caso contrário ele não pode efetuar consultas muito menos compras.

Após ter efetuado o login, o cliente pode acessar as páginas “*homecliente*”, além de poder consultar produtos e então adiciona-lo ao carrinho de compras. Assim que terminar a sua compra, o cliente escolhe a forma de pagamento, responde a uma enquete e então finaliza o processo.

Figura 6 – Esquema navegacional do *site* do cliente



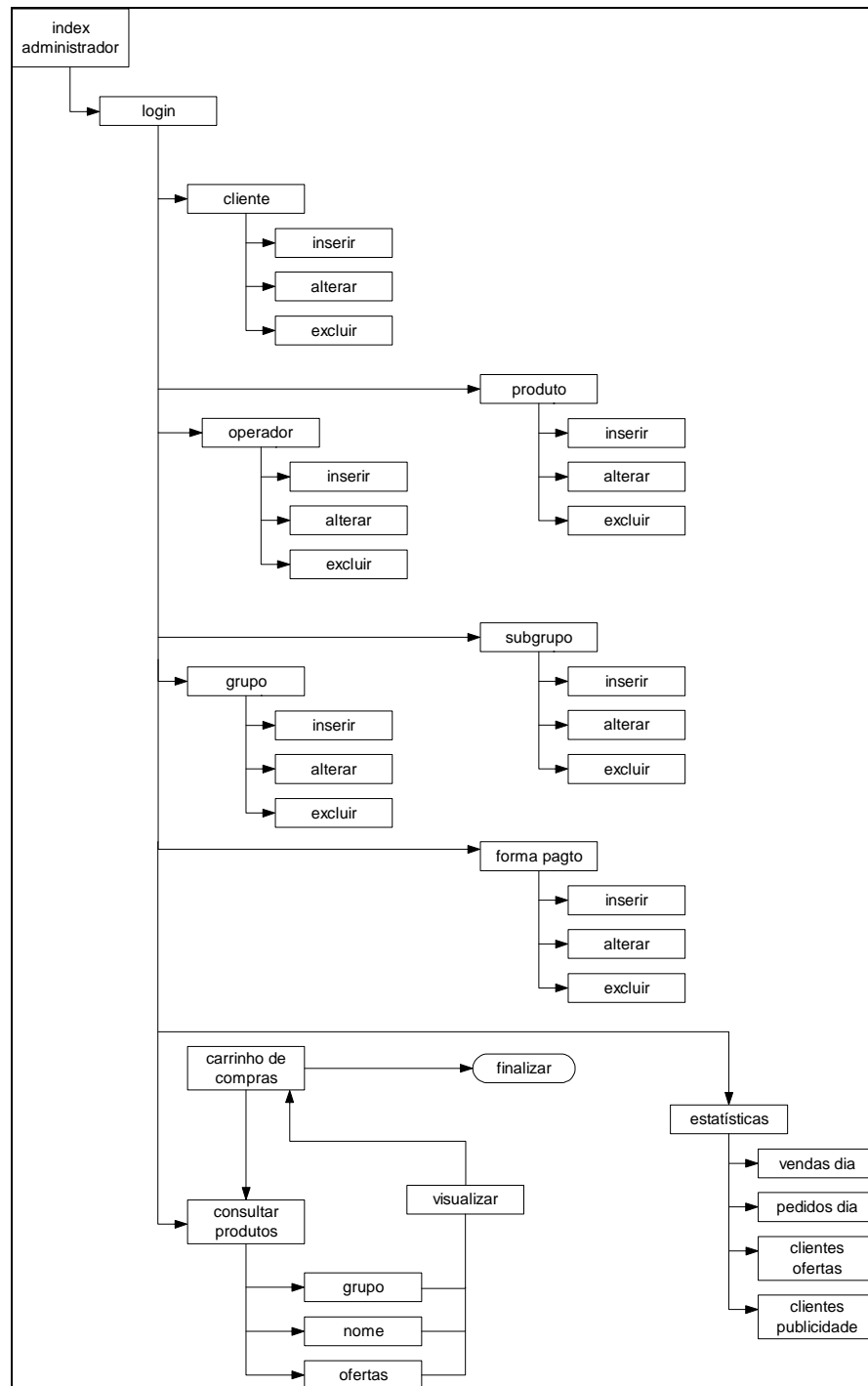
O esquema navegacional do *site* na Intranet da empresa possui um esquema semelhante ao do cliente (Figura 7). O operador deve efetuar o *login* para então poder consultar produtos e preços além de efetuar compras. Assim que terminar o operador encaminha o cliente para o caixa enquanto envia os produtos vendidos ao pacote onde o cliente retirará sua mercadoria.

Figura 7 – Esquema navegacional do *site* do operador

E o esquema navegacional do administrador é o que possui mais processos disponíveis, pois é ele quem tem o controle de todos os processos existentes para a manutenção do *site* (Figura 8).

Para entrar no sistema também é necessário efetuar o *login* e ser o administrador do sistema, pois não é qualquer operador que terá acesso a estes processos. Com isto ele pode fazer qualquer inserção, alteração ou exclusão desde clientes até grupos e produtos. O administrador também pode efetuar consultas e fazer compras como o cliente e o operador. As outras opções disponíveis são alguns dados estatísticos gerados através da quantidade de vendas diárias, valor das vendas no dia, além de listar clientes que gostariam ou não de receber e-mails promocionais e publicitários do *site*.

Figura 8 – Esquema navegacional do site do administrador



4.3.5 MIGRANDO O BANCO DE DADOS DE PRODUTOS

A partir de um banco de dados *Cobol* existente, foi gerado um arquivo texto com os dados do produto que são: código do produto, código do grupo, código do subgrupo, nome, descrição, preço, unidade de medida, valor do desconto e um campo para a oferta (sim ou não). Com o texto gerado, importou-se os dados para o *Microsoft Excel*, separados em colunas. A seguir com a planilha salva, importou-se estes dados para o banco de dados *Microsoft Access*, o qual gerou um banco de dados novo com a tabela produto e seus respectivos registros.

Para o próximo e último passo foi utilizado um software de propriedade da *Intelligent Converters* chamado Access to MySQL. A única versão disponível na Internet é de demonstração, e que importa somente cinco registros dos seis mil existentes no arquivo a ser migrado.

4.4 IMPLEMENTAÇÃO

4.4.1 TÉCNICAS E FERRAMENTAS UTILIZADAS

Durante o desenvolvimento do aplicativo notou-se que a utilização da linguagem PHP supriu as necessidades para se implementar um site de comércio eletrônico. A utilização de algumas funções do PHP como a *isset* para verificação do *login*, além das funções para conexão com o MySQL. A pacote do PHP utilizada foi a “*php-3.0.17-win32*” e o pacote MySQL é o “*mysql-3.23.41-win*”. Para uma melhor interação com o banco de dados utilizou-se uma ferramenta de interface gráfica para *Windows* chamada *mysqlfront*, o que facilitou a visualização das tabelas do banco de dados e seu conteúdo.

O PWS teve muita importância para testar o *site*, pois a cada *script* implementado o teste era executado devido possuir uma boa plataforma para testes. O compartilhamento do *site* através da Intranet não obteve problemas, funcionando perfeitamente de acordo com a requisição do *browser* do cliente.

Já para a migração do banco de dados existente foram feitas várias tentativas de vários modos. A maneira em que foi realizado este processo possibilitou a migração dos dados, não

por completo devido a versão do software *Access-to-MySQL*, mas foi possível comprovar sua funcionalidade que era o necessário.

4.4.2 OPERACIONALIDADE DA IMPLEMENTAÇÃO

O sistema irá consultar os dados diretamente na base da empresa. Através de um *script* rodando sob o servidor *Web PWS*, é aberta uma conexão ao servidor de banco de dados MySQL, que após validação das devidas permissões do usuário, liberará o acesso. Os dados serão consultados através de instruções SQL (*Structured Query Language*), e através de funções e operadores PHP, devidamente tratados.

Nesta fase apresenta-se o protótipo da aplicação conceituada. Conforme a Figura 9, a página inicial visa identificar o cliente, de forma que o cliente não identificado ou não cadastrado, não possa efetuar uma compra. Mas o cliente visitante tem a opção de olhar as ofertas sem ser cadastrado e por consequência ter efetuado o *login*. Na página principal estão disponíveis as categorias de produtos para a melhor visualização das linhas de produto existentes na loja virtual.

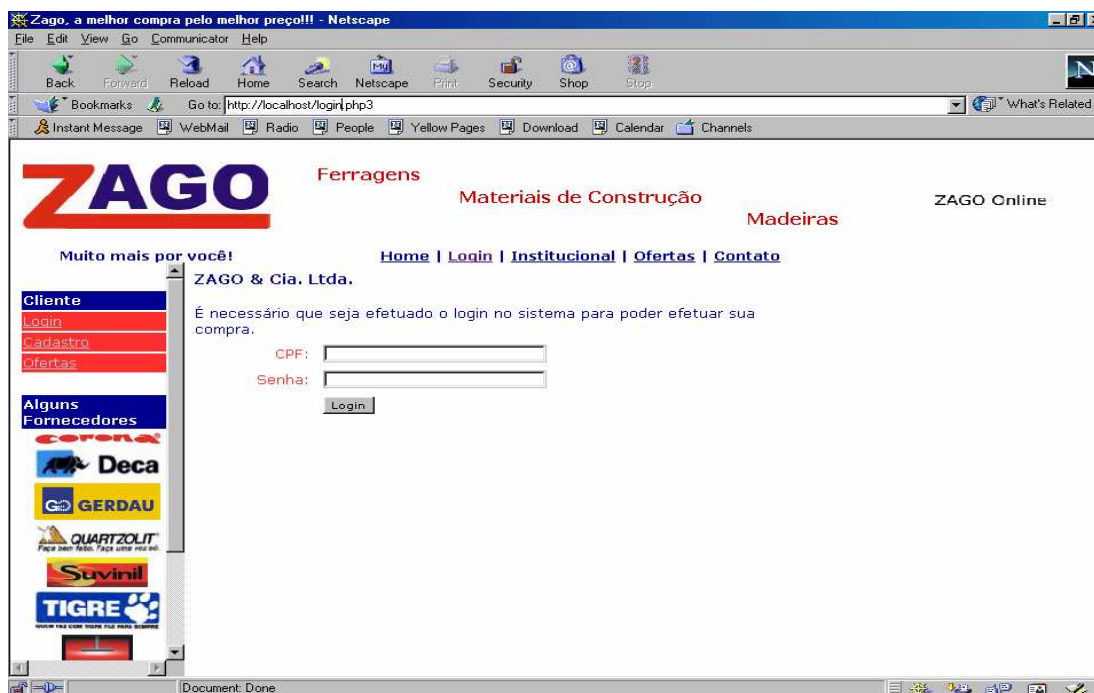
As opções disponíveis para quem entrar no *site* são: “*Login*”, “*Cadastro*”, “*Ofertas*”, “*Intitucional*” e “*Contato*”. Para o cliente ir adiante é necessário cadastrar-se ou efetuar seu *login*.

A tela seguinte da opção “*Login*” (Figura 10) exige do usuário a digitação da chave de localização, no caso o CPF (Código Pessoal Físico) e sua devida senha. Uma vez validado o CPF e senha, o sistema irá apresentar uma página de boas vindas (Figura 11). Assim que o cliente for identificado, e clicar no botão “*Clique aqui p/ continuar*”, o sistema irá disponibilizar outras opções no seu menu, conforme mostra a Figura 12. O fonte do processo de *login* do cliente está descrito Anexo 1.

Figura 9 – Página principal

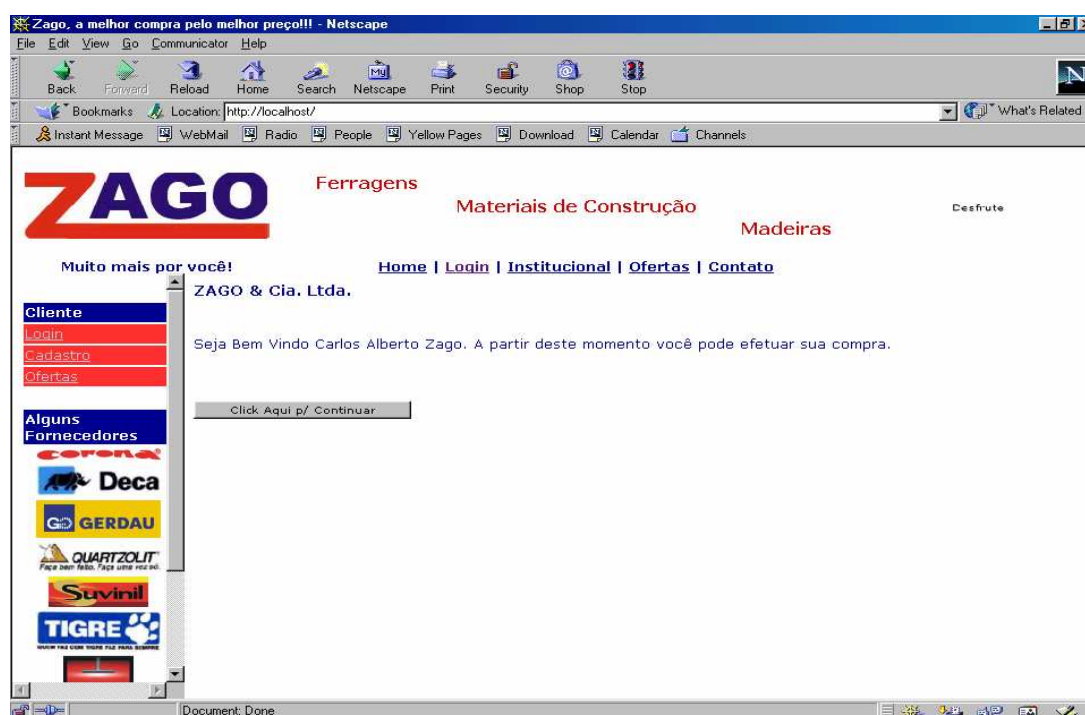


Figura 10 – Login do cliente



As opções a mais disponíveis após o cliente ter efetuado o *login* são: “Home Cliente”, “Carrinho de Compras”, além das consultas de produtos, que são pesquisados por “Nome do produto” e “Código do Grupo”. Através destas opções o cliente já pode iniciar o processo de compra pesquisando os produtos que desejar através das opções de consulta disponíveis. Na tela “Home Cliente” há uma tabela com a foto de alguns produtos que estão em promoção, com seu respectivo preço.

Figura 11 – Página de boas-vindas



Já para os clientes não cadastrados, há a opção “Cadastro”. Nesta tela o cliente preenche um formulário com seus dados (Figura 13). No formulário há duas opções que o cliente pode optar que são: “desejo receber e-mails com as ofertas da semana” e “desejo receber e-mails publicitários do *site*”. O fonte deste processo está descrito no Anexo 2. No caso de alteração dos dados cadastrais, é necessário que o cliente, clique na opção “Contato” e envie um e-mail solicitando as mudanças que desejar, para que somente o administrador do sistema possa efetuar esta modificação.

Figura 12 – Login do cliente ok

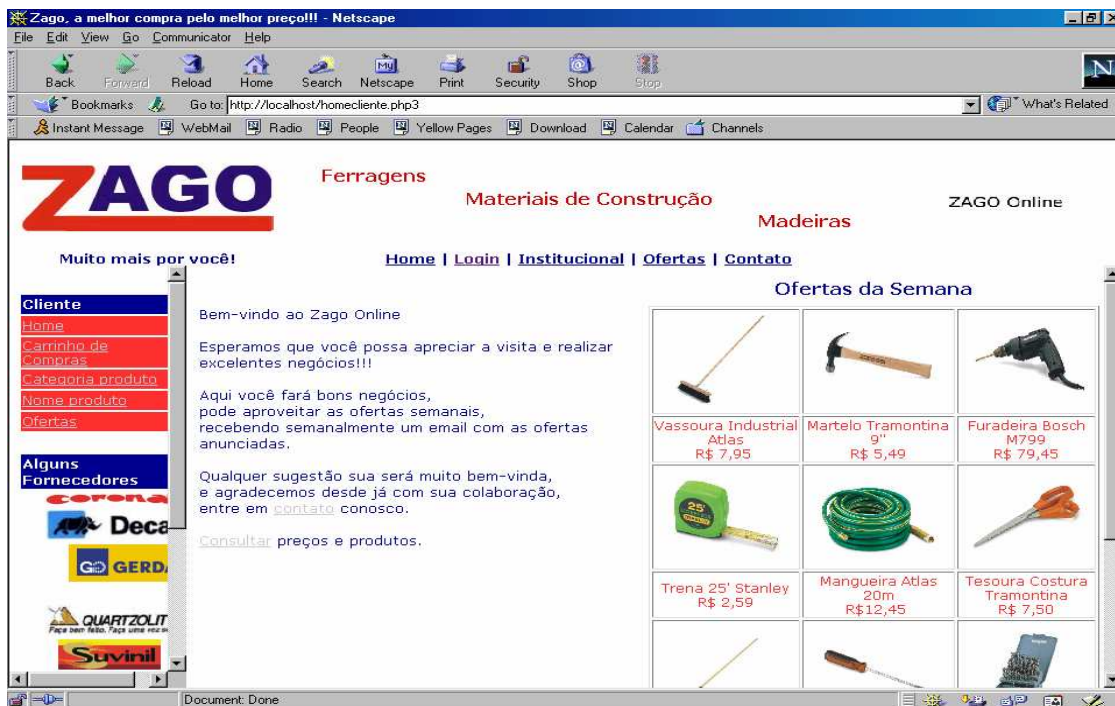
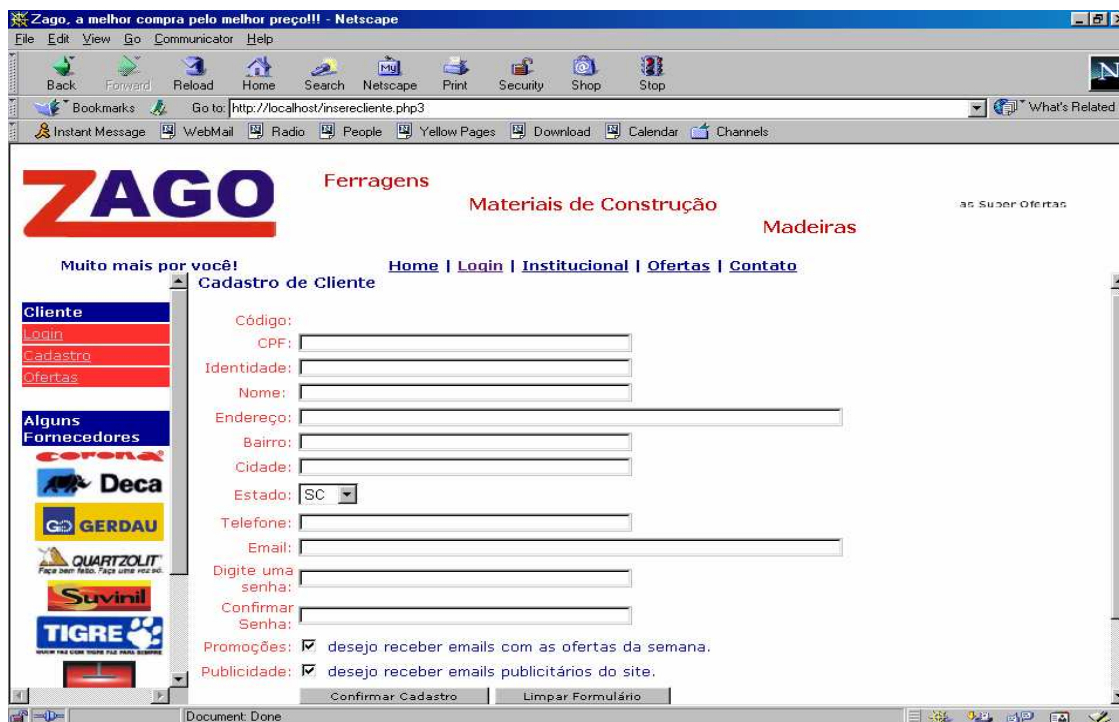


Figura 13 – Cadastro de clientes



A terceira opção disponível na *home-page* principal é a “Ofertas”. Esta opção lista todos os produtos em oferta no momento, que terão a duração exata de uma semana. O aplicativo disponibilizará o nome, unidade, preço normal, desconto (%) e o preço com o desconto (Figura 14).

Figura 14 – Ofertas da semana

The screenshot shows a Netscape browser window with the URL `http://localhost/listaofertas/home.php3`. The page title is "ZAGO, a melhor compra pelo melhor preço!!! - Netscape". The main content area displays the "ZAGO" logo and navigation links: "Home | Login | Institucional | Ofertas | Contato". Below this, the "Ofertas da Semana" section states: "Nesta semana há (s) 10 produtos em oferta, confira!." A table lists the following products:

Nome	Unidade	Preço	Desconto(%)	Preço Oferta
Argamassa Votoran AC-1 int.	kg	0.22	10	0.198
Caixa D'água Fibra Vidro Bakof 200001	pç	2324.00	15	1975.4
Caixa D'água Fibra Vidro Bakof 5001	pç	109.00	15	92.65
Capa de muro	pç	0.38	7	0.3534
Coleira p/ Cão nº2	pç	1.25	10	1.125
Porta Toalha Bometal Argola 6004	pç	8.65	15	7.3525
Serra Multi-máquina Bethil s/ Torno	pç	331.75	30	232.225
Talha Berg Compacta 0.5 x 3m	pç	231.40	10	208.26
Tijolo 06 furos 10x15x19 cm	mil	121.30	15	103.105
Torneira Bóia Bometal Balão Metal 1/2	pç	13.80	10	12.42

The sidebar on the left contains a "Cliente" menu with "Login" and "Cadastro" options, and a "Fornecedores" section with logos for Deca, Gerdaul, Savinil, and Tigre.

Se o cliente já estiver efetuado seu *login*, a suas opções na tela aumentam. A opção ofertas, o cliente já pode clicar no código do produto e visualizar a foto do produto (se disponível no banco de dados), o nome, unidade e preço, conforme mostra a Figura 15. O cliente pode consultar os produtos que quiser tanto pelo seu nome, ou trecho do nome (Figura 16). O fonte deste processo está no Anexo 3. Outras consultas disponíveis à nível de produto são pelo código da categoria, o qual lista os produtos de uma determinada categoria como por exemplo “Acessórios para banheiro”.

Figura 15 – Ofertas para cliente com login ok

ZAGO Ferragens Materiais de Construção Madeiras

Muito mais por você! [Home](#) | [Login](#) | [Institucional](#) | [Ofertas](#) | [Contato](#)

Ofertas da Semana

Nesta semana há (s) 10 produtos em oferta, confira!

Código	Nome	Unidade	Preço	Desconto(%)	Preço Oferta
14741	Argamassa Votoran AC-1 int.	kg	0.22	10	0.198
24406	Caixa D'água Fibra Vidro Bakof 200001	pç	2324.00	15	1975.4
18864	Caixa D'água Fibra Vidro Bakof 5001	pç	109.00	15	92.65
12003	Capa de muro	pç	0.38	7	0.3534
17978	Coleira p/ Cão nº2	pç	1.25	10	1.125
22328	Porta Toalha Bometal Argola 6004	pç	8.65	15	7.3525
23009	Serra Multi-máquina Bethil s/ Torno	pç	331.75	30	232.225
23485	Talha Berg Compacta 0.5 x 3m	pç	231.40	10	208.26
13315	Tijolo 06 furos 10x15x19 cm	mil	121.30	15	103.105
13216	Tomeira Bóia Bometal Balão Metal 1/2	pç	13.80	10	12.42

Figura 16 – Consulta de produtos pelo nome ou trecho do nome

ZAGO Ferragens Materiais de Construção Madeiras ZAGO Online

Muito mais por você! [Home](#) | [Login](#) | [Institucional](#) | [Ofertas](#) | [Contato](#)

Consulta de Produtos

Entre com o trecho do produto para pesquisar.

Trecho:

Encontrado(s) 10 produto(s) no trecho "agu".

Código	Nome	Unidade	Preço
18863	Caixa D'água Fibra Vidro Bakof 2501	pç	68.90
18864	Caixa D'água Fibra Vidro Bakof 5001	pç	109.00
18862	Caixa D'água Fibra Vidro Bakof 10001	pç	158.70
16013	Caixa D'água Fibra Vidro Bakof 20001	pç	355.00
16992	Caixa D'água Fibra Vidro Bakof 30001	pç	520.50
16993	Caixa D'água Fibra Vidro Bakof 50001	pç	750.00
16994	Caixa D'água Fibra Vidro Bakof 70001	pç	983.60
16719	Caixa D'água Fibra Vidro Bakof 100001	pç	1266.00

Logo que o cliente achar o que estiver procurando, para comprar o produto, basta clicar no código e aparecerá a tela da Figura 17, a qual mostra o nome do produto em destaque, além do seu preço e unidade. Então digita-se a quantidade desejada e clicar em comprar. Aparecerá uma outra tela, a do “Carrinho de Compras”, que contém os itens que o cliente deseja comprar (Figura 18). Assim que quiser o cliente pode finalizar suas compras, escolher a forma de pagamento e confirmar o local de entrega dos produtos. Inicialmente a empresa só entregará mercadorias no estado de Santa Catarina devido a infra-estrutura da empresa na parte de fretes.

Após o cliente ter terminado suas compras, deve clicar em “Finalizar”, que chamará outra página, a de confirmação da compra, local de entrega e forma de pagamento (Figura 19). Com isto o cliente efetivou sua compra com sucesso. Agora além de responder a enquête no final da sua compra (Figura 20), é só aguardar que a encomenda já estará sendo encaminhada para a entrega. O cliente estará a par da situação de sua encomenda através de e-mails enviados pela empresa que informará o dia previsto de entrega e a situação da encomenda em relação ao pagamento.

Figura 17 – Compra de produtos

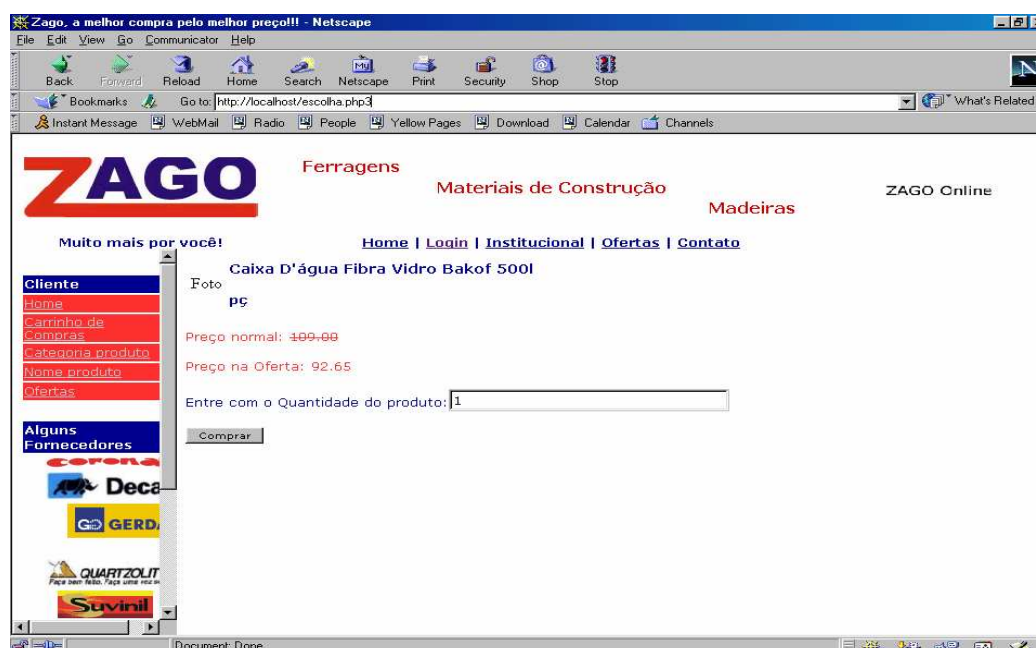


Figura 18 – Carrinho de compras

ZAGO Ferragens Materiais de Construção Madeiras ZAGO Online

Muito mais por você! [Home](#) | [Login](#) | [Institucional](#) | [Ofertas](#) | [Contato](#)

Carrinho de Compras

Item	Produto	Quantidade	Valor Unit	Valor Total
1	Torneira Bóia Bometal Balão Metal 1/2	6	12.42	74.52
2	Caixa D'água Fibra Vidro Bakof 250l	1	68.90	68.90
3	Cinto Eletricista	2	11.00	22.00
4	Vassoura Bettanin	4	6.00	24.00
5	Cal p/ Construção 20kg	20	2.70	54.00

Data da Compra: 13/12/2001
Valor Total: 243.42

Document: Done

Figura 19 – Escolha da forma de pagamento e local de entrega

ZAGO Ferragens Materiais de Construção Madeiras ZAGO Online

Muito mais por você! [Home](#) | [Login](#) | [Institucional](#) | [Ofertas](#) | [Contato](#)

Forma de Pagamento: Cartão de crédito Boleto bancário digital

Local de entrega: Mesmo local do cadastro Outro local

Document: Done

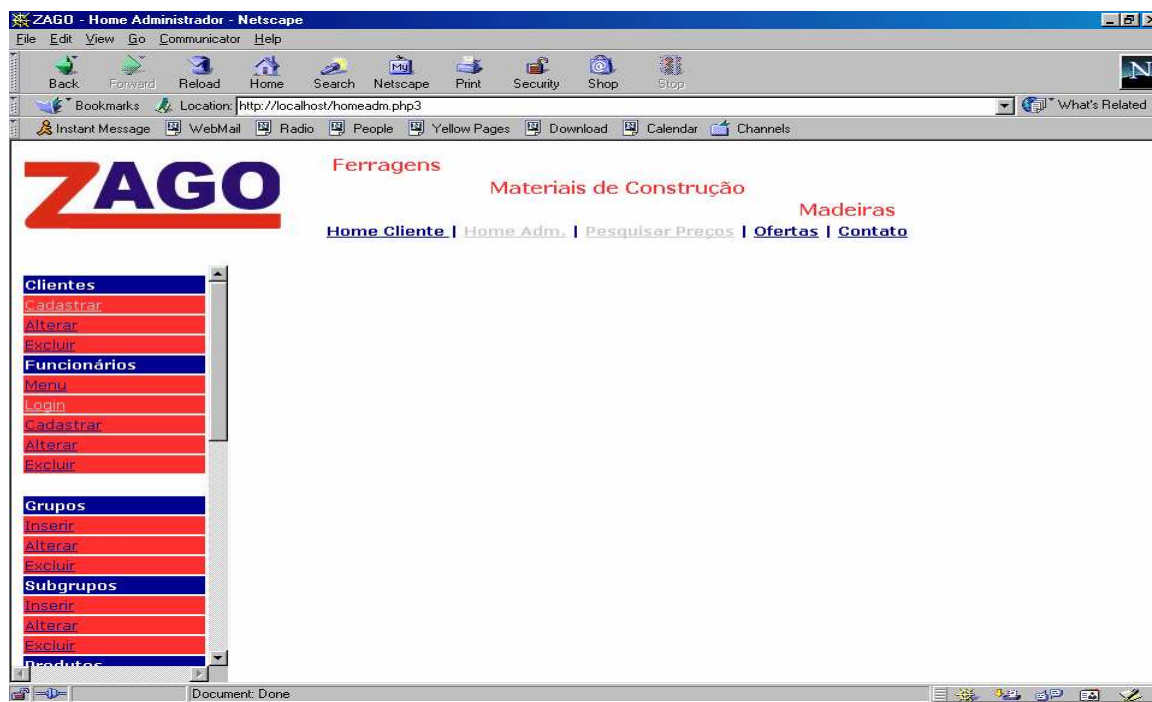
Figura 20 – Enquête satisfação do cliente



Já para a parte administrativa do aplicativo a interface muda bastante, mas o processo é o mesmo (Figura 21). É necessário que o administrador do sistema entre com seu CPF e senha. Com isto o administrador tem acesso a todas as opções disponíveis para o controle do sistema.

As suas principais opções são inserir, alterar e excluir clientes, funcionários, grupos, subgrupos, produtos, formas de pagamento, além das consultas por nome, grupo e subgrupo de produtos, e também as estatísticas de vendas e clientes.

Figura 21 – Área administrativa



5 CONCLUSÕES

A realização deste trabalho possibilitou o estudo e aprendizagem da linguagem de *script* PHP. Esta linguagem é de fácil entendimento e possibilita criar bons sites dinâmicos com uma certa facilidade. Outro ponto importante é o banco de dados MySQL, que possibilitou a migração dos registros de produtos do banco de dados *Cobol* existente na empresa. Apesar de não possuir a versão *full* do software *Acces-to-MySQL* utilizado para o último passo da migração dos dados, ainda foi possível comprovar a funcionalidade do mesmo, pois possui uma interface simples e útil para quem necessite este tipo de conversão mais freqüentemente.

Todas as ferramentas utilizadas foram adequadas para o desenvolvimento do trabalho. Outro grande motivo para a utilização destas ferramentas é seu custo. O PHP e o MySQL tem custo zero, isto é, estão disponíveis para quem quiser na Internet. Sua portabilidade, que permite que uma aplicação seja desenvolvida em uma plataforma para ser executada em outra. A simplicidade dos *scripts* e suas conexões com o banco de dados, dentro do próprio código HTML e tendo a possibilidade de trabalhar com diversos bancos de dados e servidores *http*, além do grande número de funções pré-definidas, entre outras coisas.

O uso do PWS foi importante por ter uma interface simples, mas possui muitos recursos importantes, desde a forma de publicação do *site* tanto na Intranet como na Internet até as estatísticas geradas, as quais são importantes para acompanhar o desempenho do *site*, os dias mais movimentados, o número de visitantes entre outras.

Com este aplicativo os clientes terão acesso mais rápido e fácil aos produtos que necessitem. As pesquisas de produtos feitas na Internet, podem ser feitas da mesma maneira na Intranet da empresa em terminais que estarão disponíveis ao público exclusivamente para este fim.

O aplicativo também facilitará também o processo de venda através dos funcionários dentro da empresa que terão terminais disponíveis para consultas de produtos e efetivação de vendas, além de agilizar o processo e melhorar o atendimento ao cliente.

O acesso ao catálogo de produtos disponível no site 24 horas ao cliente, além de ajudar na escolha dos produtos. Com este serviço o cliente pode conhecer melhor a linha de produtos oferecida pela empresa.

Com isto, o relacionamento entre empresa e cliente se torna mais intenso, fazendo com que ele fique sabendo das ofertas e promoções oferecidas pela empresa simplesmente acessando o seu caixa de mensagens. Através das mensagens os clientes conseqüentemente acessaram mais vezes o *site*, fazendo disto quase que uma rotina.

Por ser um serviço diferenciado e de baixo custo de implantação para a empresa, foi possível desenvolver este projeto oferecendo uma nova opção de compra para clientes diferenciados que talvez não possam ir até a loja fisicamente, mas sim virtualmente tendo a opção de comprar o que quiser quando quiser através do *site* na *Web*.

Deve-se levar em conta que o desenvolvimento deste aplicativo visou também a melhor utilização do maquinário e da infra-estrutura oferecida pela empresa, que até então utiliza todo seu potencial tecnológico somente para controle interno. Com isto, além da empresa invadir um mercado em constante expansão, faz melhor uso de seu potencial tecnológico alcançando novos objetivos traçado por uma boa estratégia de comércio eletrônico.

Com as informações adquiridas e estudadas para o desenvolvimento deste trabalho conclui-se que para continuar lutando e concorrendo pelo espaço de uma empresa no mercado é necessário inovar, sempre visando o melhor para seus clientes. É necessário simplicidade, facilidade e principalmente confiabilidade no processo. Com isto tudo, quem sai ganhando mais ainda é a empresa que mantém sua imagem como sendo um bom negócio.

5.1 EXTENSÕES

Como sugestão de continuidade, pode-se aplicar um sistema de segurança, com troca de mensagens com assinatura digital. Implementar um sistema de logística para a parte de entregas e expandir o serviço para todo o Brasil. Outra sugestão é utilizar sistemas especialistas para criar o perfil dos clientes para saber o que eles mais gostam quando acessam o *site*.

ANEXO 1 – LOGIN DO CLIENTE

```

<html>
<? require("config.ini"); ?>
<head>
  <title>ZAGO - Identificação do Cliente</title>
</head>
<?
if (!isset($login)) {
?>
<p class="headcadastro">ZAGO & Cia. Ltda.</p>
<table border=0 width=500 cellpadding=0 cellspacing=8>
  <tr>
    <td colspan=2>
      <p class="textos">É necessário que seja efetuado o login no sistema para poder
efetuar sua compra.
    </td>
  </tr>
  <tr>
    <td align=center>
      <form method=post action="<? echo $PHP_SELF; ?>">
        <table width=100%>
          <tr>
            <td>
              <p class="opctabcadastro" align=right>CPF:
            </td>
            <td>
              <input type=text size=20 maxlength=11 name="edcpf" style="border: 1
solid #000099">
            </td>
          </tr>
          <tr>
            <td>
              <p class="opctabcadastro" align=right>Senha:
            </td>

```

```

        <td>
            <input type=password size=20 maxlength=16 name="edsenhacli">
        </td>
    </tr>
    <tr>
        <td>&nbsp;&nbsp;&nbsp;</td>
        <td>
            <input type=submit name="login" value="Login">
        </td>
    </tr>
</table>
</form>
</td>
</tr>
</table>
</html>
<?
die();
}
elseif($login == "Login") {
    $db = mysql_connect("$host", "$user", "$senha") or die("Erro de Conexão");
    mysql_select_db("$dbName", $db) or die("Base de dados não existe");
    require("config.ini");
    $query = "SELECT nomcliente FROM cliente WHERE "
        ."(cpfcliente='$edcpf') and (senhacliente='$edsenhacli)";
    $resulta = mysql_query($query, $db);
    if (mysql_num_rows($resulta) == 0) {
?>
<body>
<p class="headcadastro">ZAGO & Cia. Ltda.</p>
<table>
<tr><td><p class="textos">
    Inválida combinação do CPF ou Senha.<br>
    <a href=contato.php3>Esqueceu sua senha?</a>

```

```

        </td></tr>
</table>
<?
    mysql_free_result($resulta);
    mysql_close($db);
    die();
}
$nome = mysql_result($resulta,0,"nomcliente");
?>
<form method=post action="mainframelogok.php3" name=loginOK target="menu">
    <input type="hidden" name="cpfcliente" value="<? echo $edcpf;?>">
    <p class="headcadastro">ZAGO & Cia. Ltda.</p>
    <br>
    <p class="textos">Seja Bem Vindo <? echo "$nome";?>. A partir deste momento
você pode efetuar sua compra.<br><br>
    <br><br>
    <p align="right"><input type=submit name="entrada" value="Click Aqui p/
Continuar"></p>
</form>
<?
    mysql_free_result($resulta);
    mysql_close($db);
}
?>
</html>

```

ANEXO 2 – CADASTRO DE CLIENTES

```

<head>
<title>ZAGO - Cadastro de Clientes</title>
<style type="text/css">
<!--
    .marcaregistrada { font-family: Verdana, Arial, Helvetica, sans-serif; font-size: 8pt;
font-style: normal; color: #000099; font-weight: bold}
    .opcadastro { font-family: Verdana, Arial, Helvetica, sans-serif; font-size: 10pt; font-
style: normal; color: #FF3333}
    .textos { font-family: Verdana, Arial, Helvetica, sans-serif; font-size: 10pt; color:
#000099}
    .headcadastro { font-family: Verdana, Arial, Helvetica, sans-serif; font-size: 12pt;
font-style: normal; font-weight: bold; color: #000099}
-->
</style>
</head>
<script language="JavaScript">
function checkForm(form) {
    //Verifica todos os campos
    for (var i =0; i < form.elements.length; i++) {
        if (form.elements[i].value == "") {
            alert("Preencha todos os campos")
            return false
        }
    }
    //Valida CPF
    if ( (!checkCPF(form.cpfcliente.value)) || (!eNumero(form.cpfcliente.value))) {
        alert("Número de CPF inválido")
        form.cpfcliente.focus()
        form.cpfcliente.select()
        return false
    }
    //Valida o telefone

```

```

if (!eNumero(form.telefone.value)) {
    alert("O número de telefone deve conter somente números")
    form.telefone.focus()
    form.telefone.select()
    return false
}
//Verifica as senhas
if (form.senhacliente.value != form.senhaclientev.value) {
    alert("As senhas devem possuir o mesmo valor")
    form.senhacliente.focus()
    form.senhacliente.focus()
    return false
}
return true
}
function checkCPF(numcpf) {
    x = 0;
    soma = 0;
    dig1 = 0;
    dig2 = 0;
    texto = "";
    numcpf1="";
    len = numcpf.length; x = len - 1;
    for (var i=0; i <= len - 3; i++) {
        y = numcpf.substring(i,i+1);
        soma = soma + ( y * x);
        x = x - 1;
        texto = texto + y;
    }
    dig1 = 11 - (soma % 11);
    if (dig1 == 10) dig1=0 ;
    if (dig1 == 11) dig1=0 ;
    numcpf1 = numcpf.substring(0,len - 2) + dig1 ;
    x = 11; soma=0;

```



```

for (var i=0; i <= len - 2; i++) {
    soma = soma + (numcpf1.substring(i,i+1) * x);
    x = x - 1;
}
dig2= 11 - (soma % 11);
if (dig2 == 10) dig2=0;
if (dig2 == 11) dig2=0;
if ((dig1 + "" + dig2) == numcpf.substring(len,len-2)) {
    return true;
}
return false;
}

function eNumero(valor) {
    for (i=0; i<valor.length; i++) {
        if (isNaN(parseInt(valor.charAt(i)))){
            return false;
        }
    }
    return true;
}
</script>
<?
require("config.ini");
$res1 = mysql_connect("$host", "$user", "$senha");
$res2 = mysql_db_query("$dbNome", $res1);
?>
<form onSubmit="return checkForm(this)" name="cadcliente"
action="cadcliente.php3" method="post">
    <p class="headcadastro">Cadastro de Cliente</p>
    <table width="50%" border="0">
        <tr>
            <td width="26%" class="opcadastro">
                <div align="right">C&ocirc;digito:</div>
            </td>

```

```

        <td width="74%">
            <input type="hidden" name="codcliente" value="<? echo "$codcliente" ?>
size="30" maxlength="15">
        </td>
    </tr>
    <tr>
        <td width="26%" class="opcadaastro">
            <div align="right">CPF:</div>
        </td>
        <td width="74%">
            <input type="text" name="cpfcliente" size="30" maxlength="11">
        </td>
    </tr>
    <tr>
        <td width="26%" class="opcadaastro">
            <div align="right">Identidade:</div>
        </td>
        <td width="74%">
            <input type="text" name="idcliente" size="30" maxlength="15">
        </td>
    </tr>
    <tr>
        <td width="26%" class="opcadaastro">
            <div align="right">Nome: </div>
        </td>
        <td width="74%">
            <input type="text" name="nomcliente" size="30" maxlength="60">
        </td>
    </tr>
    <tr>
        <td width="26%" class="opcadaastro">
            <div align="right">Endere&ccedil;o:</div>
        </td>
        <td width="74%">

```

```
<input type="text" name="endereco" size="50" maxlength="60">
</td>
</tr>
<tr>
<td width="26%" class="opcadaastro">
  <div align="right">Bairro:</div>
</td>
<td width="74%">
  <input type="text" name="bairro" size=30 maxlength=30>
</td>
</tr>
<tr>
<td width="26%" class="opcadaastro">
  <div align="right">Cidade:</div>
</td>
<td width="74%">
  <input type="text" name="cidade" size="30" maxlength="40">
</td>
</tr>
<tr>
<td width="26%" class="opcadaastro">
  <div align="right">Estado:</div>
</td>
<td width="74%">
  <select name="estado">
    <option value="SC" selected>SC</option>
    <option value="PR">PR</option>
    <option value="RS">RS</option>
    <option value="RJ">RJ</option>
    <option value="MG">MG</option>
    <option value="ES">ES</option>
    <option value="MT">MT</option>
    <option value="MS">MS</option>
    <option value="SP">SP</option>
```

```

        </select>
    </td>
</tr>
<tr>
    <td width="26%" class="opcadaastro">
        <div align="right">Telefone:</div>
    </td>
    <td width="74%">
        <input type="text" name="telefone" size="30" maxlenght="10" maxlength="15">
    </td>
</tr>
<tr>
    <td width="26%" class="opcadaastro">
        <div align="right">Email:</div>
    </td>
    <td width="74%">
        <input type="text" name="email" size="50" maxlength="80">
    </td>
</tr>
<tr>
    <td width="26%" class="opcadaastro">
        <div align="right">Digite uma senha:</div>
    </td>
    <td width="74%">
        <input type="password" name="senhacliente" size="30" maxlength="16">
        <? $senha_criptigrafada=md5('senhacliente'); ?> </td>
</tr>
<tr>
    <td width="26%" class="opcadaastro">
        <div align="right">Confirmar Senha:</div>
    </td>
    <td width="74%" class="textos">
        <input type="password" name="senhaclientev" size="30" maxlength="16"></td>
</tr>

```

```

<tr>
  <td width="26%" class="opcadastro">
    <div align="right">Promoções:</div>
  </td>
  <td width="74%" class="textos">
    <input type="checkbox" name="promos" value="S" checked>
    desejo receber emails com as ofertas da semana.</td>
</tr>
<tr>
  <td width="26%" class="opcadastro">
    <div align="right">Publicidade:</div>
  </td>
  <td width="74%" class="textos">
    <input type="checkbox" name="publicidade" value="S" checked>
    desejo receber emails publicitários do site.</td>
</tr>
<tr>
  <td width="26%" class="opcadastro">
    <div align="right"></div>
  </td>
  <td width="74%">
    <input type="submit" name="Submit" value="Confirmar Cadastro">
    <input type="reset" name="Reset" value="Limpar Formulário">
  </td>
</tr>
</table>
<span class="headcadastro"></span>
</form>
<p>&nbsp;</p><table width="100%" border="0">
  <tr>
    <td>
      <div align="center" class="marca registrada">&copy; 2001 Zago & Cia.
      Ltda.</div>
    </td>
  </tr>

```

```
</tr>
</table>

<?
mysql_close($res1);
?>
</html>
```

ANEXO 3 – CONSULTA POR NOME OU TRECHO DO NOME DE PRODUTOS

```

<html>
<? require("config.ini"); ?>
<head>
  <title>ZAGO - Consulta de Produtos</title>
</head>
<body bgcolor="<? echo $bgForm ?>">
<p class="headcadastro">Consulta de Produtos</p>
<?php
echo "<p class='textos'>Entre com o trecho do produto para pesquisar.</p>";
echo "<form action='\$PHP_SELF' method='POST'>";
echo "<p class='opctabcadastro'>Trecho:";
echo "<input type='text' name='trecho' value='\$trecho' ";
echo " style='border:1 solid #000099'></p>";
echo "<input type='submit' value='Pesquisar'>";
echo "</form>";
if ($trecho == "") { die(); }
$db = mysql_connect("$host", "$user", "$senha") or die("Erro de Conexão");
mysql_select_db("$dbNome", $db) or die("Base de dados não existe");
$sql = "SELECT * FROM produto WHERE (nome LIKE '%$trecho%')";
$resSql = mysql_query($sql, $db);
$totProd = mysql_num_rows($resSql);
if ($totProd == 0) {
  echo "<p class='textos'>Nenhum produto encontrado para esta informação</p>";
}
elseif ($totProd > 0) {
  echo "Encontrado(s) $totProd produto(s) no trecho \"<b>$trecho</b>\".<br><br>";
  $counter = 0;
  if ($dir=="") { $dir = "Próximo"; }
  $maxPag = 10;
  if ($offset==0) { $offset=0; }
}

```

```

if ($dir=="Próximo") {
    if ($totProd > $maxPag) {
        $counter = $offset;
        $offset += $maxPag;
        $num = $offset;
        if ($num > $totProd) { $num = $totProd; }
    }
    else { $num=$totProd; }
}
elseif ($dir=="Anterior") {
    if ($totProd > $maxPag) {
        $offset -= $maxPag;
        if ($offset < 0) { $offset = 0; }
        $counter = $offset - $maxPag;
        if ($counter < 0) { $counter = 0; }
        $num = $counter + $maxPag;
    }
    else { $num = $totProd; }
}
echo "<table width=100%>\n";
echo "<tr><th>Código<th>Nome<th>Descrição<th>Preço</tr>";
while ($counter < $num) {
    $nid = mysql_result($resSql,$counter,"codproduto");
    echo "<tr><td><a href=\"escolhateste.php3?codproduto=$nid\">\".$nid.\"</a>";
    echo "<td>".mysql_result($resSql,$counter,"nome");
    echo "<td>".mysql_result($resSql,$counter,"descricao");
    echo "<td>".mysql_result($resSql,$counter,"preco").\"</tr>\n";
    $counter++;
}
print "</table><br>";
}
mysql_close();
if ($totProd > $maxPag) {
    echo "<form action=\"\$PHP_SELF\" method=\"POST\">\n";
}

```



```
        if ($offset > $maxPag) { echo "<input type=\"submit\" value=\"Anterior\"
name=dir>\n"; }
        if ($offset < $totProd) { echo "<input type=\"submit\" value=\"Próximo\"
name=dir>\n"; }
        echo "<input type=hidden name=\"offset\" value=\"$offset\">\n";
        echo "<input type=hidden name=\"trecho\" value=\"$trecho\">\n";
        echo "</form>";
    }
?>
</body>
</html>
```

REFERÊNCIAS BIBLIOGRÁFICAS

ANSELMO, Fernando. **PHP e MySQL para Windows**. Florianópolis: Visual Books, 2000.

BARRETO, Maurício Vivas de Souza. **Curso de Linguagem PHP**. Sergipe: Universidade Federal de Sergipe, 2000.

EANBRASIL, Ean Brasil. **Automação comercial**, São Paulo, 1999. Disponível em: <www.eanbrasil.org.br>. Acesso: 15 abr. 2001.

FISCHER, Herber. **PHP - Guia de consulta rápida**. Rio de Janeiro: Novatec, 2000.

GONÇALVES, Cid. **Comércio eletrônico na Internet**: uma pesquisa exploratória do mercado consumidor. Disponível em: <<http://orion.planetarium.com.br/ebusiness/comelet/>>. Acesso: 23 out. 2001.

HARTMAN, Amir. **Pronto para a web**: estratégias para o sucesso na economia. Tradução Carlos Antônio Moura. Rio de Janeiro: Campus, 2000.

MICROSOFT CORPORATION. **Tour do Personal Web Server**. Microsoft, 1998.

SQL SERVER. **O que é SQL Server**. Disponível em: <<http://www.sql.com.br>> Acesso: 27 set. 2000.

STOCO, Lucio M. **Integrando PHP com MySQL**: guia de consulta rápida. São Paulo: Novatec, 2000

STRANGE, Valda. **Protótipo de sistema de apoio a gestão para corretora de seguros**. 2000. 71 f. Trabalho de Conclusão de Curso (Bacharelado em Ciências da Computação) – Centro de Ciências Exatas e Naturais, Universidade Regional de Blumenau, Blumenau.

VIVAS, Maurício. **Aplicações Web utilizando PHP**. Disponível em: <<http://www.mauricio.vivas.com.br>> Acesso: 13 set. 2001.