

CENTRO UNIVERSITÁRIO DO SUL DE MINAS - UNIS-MG

BACHARELADO EM SISTEMAS DE INFORMAÇÃO

ROGÉRIO RAMOS CORGOSINHO

N. CLASS.	M006.786
CUTTER	C787a
ANO/EDIÇÃO	2014

APLICAÇÃO WEB: Compra Café

Varginha/MG

2014

ROGÉRIO RAMOS CORGOSINHO

FEPESMIG

APLICAÇÃO WEB: Compra Café

Monografia apresentada ao curso de graduação em Bacharelado em Sistemas de Informação do Centro Universitário do Sul de Minas – UNIS/MG, como pré-requisito para conclusão do curso, sob orientação do Prof. Rodrigo Gomes da Silva.

Varginha/MG

2014

ROGÉRIO RAMOS CORGOSINHO

APLICAÇÃO WEB: Compra Café

Monografia apresentada ao curso de Bacharelado em Sistemas de Informação do Centro Universitário do Sul de Minas – UNIS/MG, como pré-requisito para obtenção do grau de bacharel pela Banca Examinadora composta pelos membros:

Aprovado em / /

Prof. (Especialista) Rodrigo Gomes da Silva

Prof. (Especialista) César Fernandes Ribeiro Filho.

Prof. (título ex.: Dr./Ms./Esp.) Nome do professor

OBS.:

Dedicatória Dedico esse trabalho a minha esposa e filhas que me suportaram todo esse tempo estudando nas madrugadas e a todos aqueles que dedicaram o seu tempo me passando algum tipo de conhecimento.

AGRADECIMENTOS

Agradeço aos meus coordenadores deste trabalho e do curso de Sistemas de Informação, a todos os professores desse curso e a minha família por terem ajudado na construção deste trabalho.

RESUMO

Esse trabalho mostra a abordagem do desenvolvimento do protótipo de uma Aplicação web com o nome de “Compra Café” que visa possibilitar a uma melhor agilidade ao processo de comercialização dos produtos da Cooperativa dos Cafeicultores da Zona de Três Pontas – Cocatrel, visando auxiliar principalmente controle do faturamento, no planejamento das entregas, na formação de um banco de dados e aos clientes na aquisição dos produtos da cooperativa. Foram utilizados alguns dos conhecimentos conceituais de Engenharia de Software, Engenharia de Requisitos e a modelagem do Banco de Dados que foram apresentados no curso de Sistemas de Informação. Foi utilizada a Metodologia Científica Experimental em que foram apresentadas as etapas da observação do problema, a coleta de dados, a análise dos requisitos, a descrição da pesquisa e a comprovação das hipóteses levantadas com a apresentação dos resultados do trabalho: um “wireframe web” com as partes mais básicas do projeto, Diagramas Entidade Relacionamento Conceitual e Lógico, Especificações e Diagramas de Caso de Uso de todo o projeto e os relatórios do estoque, de marketing e do planejamento das entregas.

Palavras-chave: Aplicação web, controle, planejamento, relatórios, banco de dados.

ABSTRACT

This work shows the application of a web prototype development approach with the name of "Coffee Shop" which aims to enable better streamline the process of marketing the products Cooperative of Coffee Growers of Three Points area - Cocatrel, aimed mainly auxiliary control billing, planning of deliveries in the formation of a database and clients in the acquisition of the products of the cooperative. Some of the conceptual knowledge of Software Engineering, Requirements Engineering and modeling the database that were presented in the course of Information Systems were used. Were used the Experimental Scientific Methodology they were presented the steps of the observation of the problem, data collection, analysis of requirements, the description of the research and proof of the hypotheses with the presentation of the results of the work: a 'web wireframe "with the most basic parts of the project, Entity Relationship Diagrams conceptual and logical, specifications and diagrams Use Case of the entire project and stock reports, marketing and planning of deliveries.

Keywords: *Web application, control, planning, reporting, database.*

LISTA DE ILUSTRAÇÕES

Figura 01 – Quadro comparativo das versões Android.....	04
Figura 02 – Representação Gráfica – Modelo Cascata.....	16
Figura 03 – Representação Gráfica – Prototipação.....	18
Figura 04 – Representação Gráfica – Modelo Espiral Completa.....	19
Figura 05 – Diagrama de Caso de Uso.....	26
Figura 06 – Diagrama de Classes.....	27
Figura 07 – Diagrama Entidade Relacionamento – Conceitual.....	35
Figura 08 – Diagrama Entidade Relacionamento – Lógico.....	36
Figura 09 – Página inicial da Aplicação Web – Compra Café.....	51
Figura 10 – Formulário – Cadastro de Clientes.....	52
Figura 11 – Formulário para o Cadastro de Produtos.....	53
Figura 12 – Página inicial do Ambiente de Compras.....	54
Figura 13 – Formulário para o Cadastro Administrativo.....	55
Figura 14 – Página para Abastecer o Estoque.....	56

LISTA DE TABELAS

Tabela 01 – Tabela / Relatório do Estoque.....	55
Tabela 02 – Tabela / Relatório Marketing.....	56
Tabela 03 – Tabela / Relatório do Planejamento das entregas.....	56

LISTA DE ABREVIATURAS E SIGLAS

HTML – Linguagem de Marcação de Texto (Hipertext Markup Language).....	11
CSS – Folhas de Estilo em Cascaca (Cascading Style Sheets).....	11
PHP – Hipertexto Pré-processado – (Hipertex Preprocessor).....	12
SQL – Structured Query Language (Linguagem de Consulta Estruturada).....	13

SUMÁRIO

1 INTRODUÇÃO	1
2 Revisão da literatura	3
2.1 Sistemas de Informação	5
2.1.1 Entrada.....	6
2.1.2 Processamento	6
2.1.3 Saída	6
2.1.4 Realimentação (“ <i>feedback</i> ”).....	6
2.2 Sistemas Web	6
2.2.1 Características básicas dos sistemas web	7
2.3 Linguagens de programação.....	7
2.4 Tipos de programação.....	8
2.4.1 Programação linear	9
2.4.2 Programação estruturada	9
2.4.3 Programação orientada a objetos.....	9
2.5 Comércio Eletrônico.....	10
2.5.1 Modelos de e-commerce.....	11
2.6 HTML.....	11
2.7 CSS.....	12
2.8 PHP	12
2.9 Banco de dados.....	13
2.9.1 Banco de Dados MySQL.....	13

2.10 Modelo Entidade Relacionamento	14
2.11 Engenharia de Software.....	15
2.12 Engenharia de Requisitos	20
2.13 Requisitos Funcionais.....	22
2.14 Requisitos Não Funcionais.....	22
2.15 Regras de Negócio e Restrições	23
2.16 Modelagem de Software com UML	24
2.17 Diagrama de Casos de Uso.....	25
2.18 Especificação de Casos de Uso.....	25
2.19 Diagrama de Classes.....	27
2.20 Segurança da informação	29
2.20.1 Segurança e o Ciclo de Vida da Informação	29
2.20.2 A Segurança no Contexto da Governança de Tecnologia da Informação (TI)	30
3 MATERIAL E MÉTODOS.....	32
3.1 Descrição.....	32
3.1.1 Entrevista	33
3.1.2 Modelagem	34
3.1.3 Desenvolvimento	34
4 RESULTADOS	34
4.1 Diagrama Entidade Relacionamento - Conceitual	35
4.2 Diagrama Entidade Relacionamento - Lógico	35
4.3 Especificação do Caso de Uso – Cadastrar Cliente	36
4.4 Especificação do Caso de uso – Cadastrar usuário administrador.....	39
4.5 Especificação do Caso de uso – Cadastrar Produtos.....	41
4.6 Especificação do Caso de uso – Abastecer o estoque.....	43
4.7 Especificação do Caso de uso – Comprar produtos	46
4.8 Especificação do Caso de Uso – Verificar agendamento de entregas	48

4.9	Página inicial da Aplicação Web – Compra Café.....	50
4.10	Formulário – Cadastro de clientes	50
4.11	Formulário – Cadastro de Produtos	51
4.12	Página inicial do Ambiente de Compras	52
4.13	Formulário do Cadastro Administrativo	53
4.14	Página Abastecer o Estoque.....	54
4.15	Apresentação do Relatório do Estoque.....	55
4.16	Apresentação do Relatório Marketing	56
4.17	Apresentação do Relatório do Planejamento das Entregas.....	56
5	CONCLUSÃO.....	57
	REFERÊNCIAS.....	58

1 INTRODUÇÃO

O objetivo desse trabalho é desenvolver uma aplicação web para fins comerciais que viabilize a sua utilização em dispositivos móveis e em micro computadores, utilizando-se das ferramentas tecnológicas ministradas no curso de Bacharel em Sistemas de Informação.

Atualmente, as ferramentas tecnológicas estão cada vez mais presentes em nosso cotidiano, como também, mais próximas de nosso alcance. A essa disponibilidade são criados inúmeros aplicativos que informam, dão suporte a negociações, orientam investidores e possibilitam uma infinidade de recursos que auxiliam o cidadão contemporâneo em diversas atividades utilizando o seu computador ou dispositivos móveis como: tablets, smartphones e outros.

Visando proporcionar informações de qualidade para tomada de decisões comerciais entre empresas de médio e grande porte no cenário mercadológico do café empacotado é que esse trabalho será desenvolvido. Tendo por foco a interação entre empresas, com a utilização do modelo de “*e-commerce B2B*”, que se interagem por meio de uma aplicação web.

A Cocatrel (Cooperativa dos Cafeicultores da zona de Três Pontas Ltda) foi a empresa que, após uma reunião com a gerência comercial, despertou o interesse em compartilhar informações, e se a aplicação lhe for convincente, adequar o processo de comercialização de seus produtos com a utilização da aplicação web “Compra Café”.

O desenvolvimento de uma aplicação web foi a solução encontrada para agilizar o processo de comercialização e proporcionar uma melhor interação comercial entre o fornecedor atacadista de café empacotado o Sul de Minas Gerais (“Cocatrel”) e os seus clientes que são os supermercadistas e comercializam o produto (café “*gourmet*” empacotado).

Esse trabalho possui por objetivos específicos: definir as regras do negócio e levantar requisitos funcionais com a utilização de entrevistas; desenvolver uma modelagem de banco de dados que viabilize a uma eficiente utilização com a linguagem MySQL. Bem como, a aplicação web “Compra Café” será desenvolvida tendo por meta alcançar outros objetivos específicos concernentes às suas funcionalidades que auxiliarão a logística das entregas, a emissão dos pedidos e o controle do faturamento. Bem como, proporcionarão agilidade dos procedimentos do ciclo comercial compreendido entre a compra, a venda, a entrega e o faturamento. Bem como, economia dos processos mercadológicos e a uma melhor apresentação e divulgação dos produtos comercializáveis.

Ainda por objetivos específicos a aplicação web possuirá uma versão para computadores que utilizará as linguagens HTML5, PHP, Javascript e layout em CSS e outra para dispositivos móveis desenvolvida com a linguagem Bootstrap . Ambas as versões alimentarão o Xampp que será o único servidor de banco de dados com a utilização da linguagem MySQL desse trabalho de pesquisa.

Tendo ainda por objetivo específico a arquitetura da aplicação web será definida em três camadas: a camada de apresentação, a camada de negócio e a camada de apresentação.

Sendo que a camada de apresentação é aquela responsável pela comunicação mais próxima e direta com o usuário, sendo a camada que o usuário faz as requisições ao aplicativo através de um browser web.

A camada de negócio será onde se definirá a abordagem da aplicação web em subsistemas que definirão as propostas a serem trabalhadas e que se comunica com a camada de apresentação. Receberão as requisições dos usuários estabelecendo uma demanda de atividades programadas.

Sendo a camada de negócio a responsável pela autenticação dos usuários com o aplicativo.

A camada de persistência que será a responsável pelo armazenamento dos dados da aplicação web.

Como problema de pesquisa foi levantado o seguinte questionamento: De qual forma uma aplicação web poderá agilizar o processo de comercialização entre fornecedores e comerciantes de café?

Foi levantada a hipótese de que pelo menos um fornecedor do comércio atacadista de café empacotado do Sul de Minas Gerais terá uma aplicação web que agilizará o processo de comercialização com os seus clientes.

A justificativa para o desenvolvimento desse trabalho está embasada na melhor interatividade entre empresas, que em tempo real, poderão alcançar melhores resultados em suas atividades comerciais. Obtendo maior agilidade dos procedimentos, economia de processos e uma melhor divulgação dos produtos com a utilização de uma aplicação web.

2 REVISÃO DA LITERATURA

Uma aplicação web é do tipo “cliente-servidor” e pode ser acessada por um programa cliente em um browser. Sendo que uma aplicação responde as requisições de um usuário com a informação que lhe for solicitada.

As aplicações web possuem estruturas de navegação bem definidas e disponibilizam uma abordagem de hiperlink integrando hipermídia e transação.

Segundo Souza (2008, p.15), “Aplicações para web, *“ApsWeb”*, cobrem uma ampla gama de aplicações. Pode ser pouco mais que um conjunto de arquivos ligados por hipertexto ou ambientes complexos integrados ao banco de dados da empresa.”

As aplicações web possuem algumas características próprias, como por exemplo:

Características técnico estruturais:

- Conteúdo: sabendo-se que a web é um importante meio de informação. Uma aplicação web pode ter diferentes tipos de funcionalidades, contudo esse tipo de aplicação sempre é orientado a conteúdo. Em que se considera conteúdo por dados estruturados (banco de dados e outros) e não estruturados (vídeos, imagens, sons, arquivos de textos e outros). Sendo que, o conteúdo é dinâmico e necessita de atualizações frequentes, o que por consequência, uma aplicação web necessita de um gerenciamento de conteúdo apropriado.

- Hipertexto: o paradigma fundamental para estruturar uma informação no ambiente da web é a noção de hipertexto, em que elementos básicos são: os nós, os elos (links) e as âncoras que ativam os elos para que o usuário possa navegar por entre os nós. Essa semântica requer planejamento e estratégia para se estruturar um projeto que preserve a qualidade da navegação, evitando a desorientação do conteúdo e a perda de sentido.

- Apresentação: o aspecto visual da aplicação web é um fator de qualidade de essencial importância, tendo em vista a ação atuante da concorrência. É fundamental que a interface com o usuário seja autoexplicativa, intuitiva e consistente com o estilo interativo. Assim como, a aparência visual esteja sujeita a modismos, tendências e à disponibilidade de novas ferramentas tecnológicas a serem incorporadas ao ambiente.

- Requisitos não funcionais: performance, usabilidade, escalabilidade, robustez e segurança são alvos de convergência para uma maior atenção no desenvolvimento do projeto, visto que, uma aplicação web somente deve ser exposta ao público se estiver bem estruturada.

Características relacionadas ao uso:

- Ubiquidade: é a possibilidade de acesso da aplicação em diferentes tipos de dispositivos. Sendo que nesses casos existe a necessidade de adequações no projeto para disponibilizar a aplicação em diferentes contextos de uso.

- Infraestrutura tecnológica imprevisível: devido à grande quantidade de possíveis dispositivos do usuário final, com a existência de diferentes sistemas operacionais, configurações e tipos de hardware e software. Bem como, variados tipos de conexões de rede, cobertura e largura de banda, estabilidade, e vários outros fatores que possam até mesmo desabilitar alguma funcionalidade importante e comprometer a qualidade do serviço da aplicação web. (LOWE et SELLERS, 2001)

Para exemplificar as mudanças que ocorrem na área tecnológica e que influenciam na definição de qual versão a ser utilizada no desenvolvimento de um programa para um aplicativo. Veja na figura abaixo a evolução do sistema operacional Android e as diferentes possibilidades de funcionalidades que foram paulatinamente sendo incorporadas ao sistema operacional, bem como os comandos foram se diferenciando com o passar dos tempos. Contudo, a cada dia uma nova funcionalidade é disponibilizada aos usuários do sistema Android.

Veja abaixo, na figura 01, um quadro comparativo que demonstra a evolução das funcionalidades do sistema operacional Android.

Figura 01 – Quadro comparativo

Versão	Nome	Ano	Funcionalidades principais
1.5	CupCake	2008	Correção automática de textos e as funções Widgets (marcadores, calendários, navegador de conteúdo e outros).
1.6	Donut	2009	Busca online automática na página inicial do aparelho.
2.0	Eclair	2009	Uso de várias contas do Google no mesmo aparelho. Sincronismo de e-mails e contatos.
2.2	Froyo	2010	Acesso via wifi usando a rede 3G. Introdução do Flash no navegador.
2.3	Gingerbread	2010	Suporte a voz por IP (VOIP). Uso de aplicativos de

			voz (Skype).
3.0	Honeycomb	2011	Utilização do sistema em telas maiores (tablets).
4.0	Ice Cream Sandwich	2011	Criação de pastas na tela principal do aparelho.
4.1/4.2/4.3	Jelly Bean	2012/2013	Novos elementos internos e novo assistente de pesquisa por voz. Melhora da segurança do sistema e a função de múltiplos usuários com restrições de uso.
4.4	KitKat	2014	Novos recursos ao assistente pessoal Google Now se integram aos aplicativos instalados no aparelho.

Fonte: o autor

É notável que as versões do android se modificaram e disponibilizaram funcionalidades cada vez mais interessantes, bem como para o programador as diferentes versões dificultam um pouco, porque apresentam alguns comandos específicos a cada versão.

2.1 Sistemas de Informação

Um sistema de informação pode ser considerado como uma estrutura de softwares que trabalham em conjunto e são desenvolvidos para atenderem às solicitações específicas e que possibilitem a um melhor atendimento dos objetivos identificados ou propostos por uma organização.

Vários dispositivos podem fazer parte de um sistema e ajudar pessoas ou participar ativamente de uma organização, auxiliando em todos os níveis de atuação empresarial. Porque tanto no nível operacional, tático ou estratégico um analista pode encontrar formas de intervenção sistêmica que possa trazer benefícios a empresa.

Sistemas de informação é um conjunto de componentes inter-relacionados, desenvolvidos para coletar, processar, armazenar e distribuir informação para facilitar a coordenação, o controle, a análise, a visualização e o processo decisório. (RODRIGUES, 2013, p.36)

As atividades de um sistema de informação possuem características próprias, conforme a sua especificidade, mas que são estruturalmente comuns ao escopo desse tipo de sistema. Sendo que, as atividades comuns associadas a um sistema de informação são definidas por:

2.1.1 Entrada

É a captação e junção de dados primários que visam a uma meta de saída após o processamento.

2.1.2 Processamento

Converte os dados obtidos em informações (saídas) úteis e desejáveis aos usuários. Fase que demanda muita atenção dos programadores.

2.1.3 Saída

É a meta a ser atingida, ou o objetivo planejado a ser alcançado, além de ser a emissão ou a impressão de informações de qualidade geradas pelo processamento.

2.1.4 Realimentação (“*feedback*”)

Muitas vezes a informação gerada não foi o suficiente, ou o esperado, ou teve uma margem significativa de erros e inconsistências. A realimentação é uma saída alternativa revisional para ajustes e correções na entrada ou no processamento, e em todos os itens, nesses estágios, que possam comprometer o alcance da meta principal de disponibilizar informações de qualidade.

2.1.5 Controle

Nessa atividade de controle as devidas correções e modificações são alteradas para que o sistema funcione conforme os objetivos a serem alcançados.

Segundo Rodrigues (2014, p.48), o “*feedback* sempre deve ser acompanhado e monitorado para garantir que o sistema esteja realizando todas as suas tarefas e atendendo às expectativas a que se propõe.”

2.2 Sistemas Web

Após a popularização da internet nos anos de 1990 vários sistemas web passaram a beneficiar a população mundial em proporções cada vez maiores. E a possibilidade de

usufruir de serviços com maior conforto e agilidade traduz a viabilidade do surgimento de sistemas web em diversos segmentos de mercado, serviços, produtos e setores da sociedade.

Segundo Reis, a utilização da internet tem favorecido a aproximação entre clientes, fornecedores, órgãos públicos e a população de usuários em geral, facilitando operações comerciais e bancárias. Além do fornecimento de vários outros serviços entre os usuários, empresas e diversos tipos de instituições, que de certa forma, criou-se um vínculo de dependência com as tecnologias digitais entre eles.

É notável que na proporção em que os sistemas de gestão evoluem e o vínculo de suas funcionalidades com as da internet aumenta. Possibilitando o surgimento de diversos sistemas web. Como por exemplo: O correio web como suporte ao marketing ativo. O fornecimento de extratos de contas bancárias. Os sistemas de controle de estoque que avisam, de forma automática e via e-mail, aos fornecedores quando os níveis de estoque estiverem críticos. Os sistemas educacionais à distância (EAD's) e muitos outros.

2.2.1 Características básicas dos sistemas web

A capacidade de abrangência de um sistema web pode ser flexível e controlável, visto que, dispositivos e ferramentas de controle de acesso podem ser implementadas ao sistema, fazendo com que somente pessoas previamente cadastradas possam ter acesso e fazer alterações que lhes forem convenientes. O que viabiliza ao administrador a seleção e o controle da abrangência do sistema e a quais possíveis usuários que lhes for de interesse.

Segundo Reis, (2004 – acessado em 30/05/2014) “sistemas web têm como características básicas a utilização da internet como plataforma, o processamento é feito em tempo real e em escala de abrangência mundial e utilizando-se do compartilhamento de dados.”

A criação dos sistemas web são embasados em programas estruturados e que se utilizam de linguagens específicas (PHP, Java, ASP.NET e várias outras de última geração que facilitam, otimizam e dinamizam o acesso e a navegação dos usuários) e com o armazenamento de informações em servidores de bancos de dados. (REIS, 2004)

2.3 Linguagens de programação

Linguagem de programação é uma sintaxe padrão predeterminada por um tipo específico de linguagem (PHP, Java dentre outras), para que um programador possa utilizá-la no intuito de alcançar um objetivo definido em computadores ou dispositivo móveis. Estabelecendo um vínculo comunicativo que interage os objetivos a serem alcançados com a programação e a

execução de programas em computadores, ou dispositivos móveis, ou até mesmo por uma rede de computadores e a elementos de hardware interligados pela internet.

Segundo Aguilar, existem dois tipos principais de linguagem de programação: a Linguagem de Baixo Nível e a Linguagem de Alto Nível.

A Linguagem de Baixo Nível: é o tipo de linguagem que se aproxima da arquitetura do hardware. Como por exemplo:

- A Linguagem de Máquina: que é a linguagem que coordena o funcionamento do hardware da máquina através de combinações de padrões de bits (0's e 1's).

- A Linguagem Assembler: é aquela que com a utilização de uma simbologia chamada de mnemotécnica foi possível criar programas tradutores que interpretam a Linguagem de Máquina. Tornando possível se programar o hardware da máquina utilizando comandos em Linguagem Assembler.

As Linguagens de Alto Nível são aquelas mais facilmente interpretadas e utilizadas pelos programadores por serem mais próximas da linguagem natural (a língua inglesa principalmente).

As linguagens de programação servem para escrever programas que permitem a comunicação entre o usuário e a máquina. Programas especiais chamados tradutores (compiladores e interpretadores) convertem as instruções escritas em linguagens de programação em linguagens de máquina (0 e 1, bits) que a máquina pode entender. (AGUILAR, 2008, p.19)

Como atualmente a usabilidade dos computadores e de dispositivos móveis é bem extensa. Como também, existem vários tipos de hardwares, de softwares e de sistemas que são desenvolvidos para diversos tipos de usuários, ramos empresariais e organizações. Bem como, devido a existência de uma diversidade de dispositivos móveis que se utilizam de vários tipos de aplicativos com múltiplas funcionalidades. Tornou-se necessário o desenvolvimento de linguagens de programação que possam atender às várias tendências e objetivos diferentes, como também a satisfazer o vislumbre de usabilidade de todos os tipos de contexto, com os mais diversos tipos de equipamentos e de solicitações que surgem diariamente no dinâmico universo tecnológico disponível para a sociedade atual.

2.4 Tipos de programação

Dependendo da extensão ou das características do resultado a ser alcançado é recomendável a utilização de um tipo de programação mais apropriado. Os tipos de programação mais utilizados são: a programação linear, a programação estruturada e a programação orientada a objetos.

2.4.1 Programação linear

Esse tipo de programação é comum quando se utiliza linguagens de programação de alto nível, como: Fortran, Pascal, Basic, e outras. Bem como, o resultado esperado pelo programa depende de um processamento sequencial de uma programação desenvolvida linha à linha.

Segundo Magalhães (2009, p.18), “ a execução dos passos dos programas é sequencial com início e fim predeterminados a uma finalidade a ser alcançada. Os códigos são grandes e de difícil interpretação.”

A programação linear não é indicada para programas muito extensos.

2.4.2 Programação estruturada

Quando a resolução de um problema ou solicitação de um desempenho para se alcançar alguma meta informacional exige o desenvolvimento de um programa muito extenso, composto por um grande número de linhas de código de alguma linguagem. Existe a possibilidade da divisão de tarefas de programação em módulos, que além do aspecto funcional da utilização desse tipo de programação, é também uma forma de se trabalhar em equipe.

A programação estruturada significa escrever um programa conforme as seguintes regras: o programa tem um projeto modular; os módulos são projetados de modo descendente; cada módulo é codificado utilizando-se as três estruturas de controle básicas: sequência, seleção e repetição. (AGUILAR, 2008, p.45)

Segundo Magalhães, é o tipo de programação em que existe a modularização, que é a divisão do código em tarefas ou módulos que são chamados de procedimentos ou de funções que executam determinadas tarefas.

2.4.3 Programação orientada a objetos

A programação orientada a objetos divide o problema em partes chamadas de objetos, em que cada objeto possui atributos e métodos com particularidades que podem se interagir com outros objetos por meio de envio de mensagens.

Definir um programa em termos de objetos é uma maneira profunda de ver o software. Os objetos o obrigam a ver tudo, em nível conceitual, do que um objeto faz: seus comportamentos. Ver um objeto a partir do nível conceitual é um desvio da observação de como algo é feito: a implementação. Essa mentalidade o obriga a pensar em seus programas em termos naturais e reais. Em vez de modelar seu programa como um conjunto de procedimentos e dados separados (termos do mundo do computador), você modela seu programa em objetos. (SINTES, 2002, p.6)

Programação Orientada a Objetos é o tipo de programação utilizada em aplicativos que utilizam a linguagem Java, como também PHP e Android. Que são linguagens muito comumente utilizadas na programação de aplicações web que tenham por paradigma a noção estrutural de objetos que se interagem.

2.5 Comércio Eletrônico

O comércio eletrônico é um ramo de atividade promissor, mas que depende de um bom planejamento. Para que seja possível estruturar e viabilizar um negócio confiável, trazendo um retorno comercial positivo e saudável em suas transações. Em que, o usuário possa ter as suas solicitações atendidas dentro dos prazos estipulados, bem como, terem todas as condições de negociação atendidas. Assim como, as empresas que se propõem a trabalhar oferecendo produtos e ou serviços em sites na Internet possam faturar devidamente, pelos seus produtos e ou serviços comercializados.

O desenvolvimento informal é particularmente apropriado para o desenvolvimento de sistema de comércio eletrônico (e-commerce), com base na Web, que requerem uma combinação de habilidades de software e projetos gráficos. (SOMMERVILLE, 2003 – p.09).

Segundo Fonseca, “comércio eletrônico (CE) é a realização de toda a cadeia de valor dos processos de negócio num ambiente eletrônico, por meio da aplicação intensa das tecnologias de comunicação e de informação, atendendo aos objetivos de negócios” (FONSECA, 2010 – p.8).

Segundo Vasconcelos, o e-commerce pode ser considerado uma ferramenta estratégica de negociação, com a utilização das tecnologias que se associam à internet, visando a vantagens competitivas, melhoras nas negociações, redução de custos, melhor qualidade no atendimento aos clientes dentre outras. Contudo, o planejamento de sua implementação e utilização, como sendo a extensão da empresa na internet precisa de ser bem elaborado. Visto que, após implantado um e-commerce existe a prospecção de aumento no consumo dos produtos e conseqüentemente a necessidade de um melhor controle dos estoques, um eficiente apoio logístico, um atendimento ao consumidor diferenciado, um eficiente dispositivo de segurança das informações, bem como um suporte técnico integrado com frequente manutenção acessível.

2.5.1 Modelos de e-commerce

Existem quatro tipos de modelos de comércio eletrônico:

- B2B (Negócio-a-Negócio), que é o modelo de negócio quando somente empresas se envolvem nas negociações. Por exemplo, quando empresas de informática fornecem suprimentos diretamente a empresas em geral.

- B2C (Negócio-a-Consumidor), é quando acontecem vendas entre empresas e consumidor final. Por exemplo, o site de e-commerce das lojas americanas.

- Intra-organizacional, que é o tipo de negócio que acontece no ambiente interno das organizações, entre a empresa e seus funcionários. Em que a empresa disponibiliza condições especiais aos funcionários através de um portal exclusivo.

- C2C (Consumidor-a-Consumidor), em que as negociações acontecem entre os próprios consumidores finais (FONSECA, 2008).

O e-commerce B2B é considerado pelo nome de e-hubs, devido ao papel de ponto de distribuição e afirmam que para entendê-los é necessário saber quais são os produtos e de qual maneira eles são comprados pelas empresas. (KAPLAN e SAWHNEY, 2000)

2.6 HTML

O HTML é uma linguagem acessível, versátil e de fácil entendimento, utilizada para criar páginas web e que disponibiliza várias maneiras de se criar e organizar o conteúdo das apresentações que são interpretadas pelos navegadores “*browsers*”. Possibilitando ao desenvolvedor a escolha do tamanho e do tipo da fonte a ser utilizada. O padrão de cor da fonte e do fundo. Enfim, toda a aparência da página web em todos os aspectos, bem como a possibilidade de configurar a presença de imagens, links, sons vídeos.

Tim Beners-Lee [...] Tanto estudou que acabou unindo e criando as chamadas três tecnologias elegantes da “*Word Wide Web*”: a URL (ou URI), que é o endereço para identificar os recursos e saber onde encontrá-los; o HTTP, que é o protocolo para mover dados na web e em toda a internet; e o HTML, que é a linguagem para representar o conteúdo em termos de páginas na web e promover sua união através de links. (GOMES, 2010 – p.9)

O HTML quer dizer “*Hipertext Markup Language*” é uma linguagem que formata um texto digital chamado de hipertexto, com o qual se cria páginas para a internet, podendo se verificar os resultados de sua utilização em vários navegadores.

Segundo Carril, “o HTML é uma linguagem de marcação, utilizada para formatar páginas Web com texto e informações separadamente. Estas páginas são acessíveis qualquer tipo de browser como: Internet Explorer, Firefox, Chrome, Opera ou Safari.” (CARRIL, 2012 – p.3).

2.7 CSS

O CSS é uma abreviação de “*Cascading Style Sheets*”, sendo portanto uma forma acessível de criar estilos a documentos web criados com HTML ou XHTML a partir de folhas de estilo que se compõem de regras que determinam a aparência do portal.

Segundo Gomes (2010, p.11), “o CSS é o responsável pelo aspecto da página, pelo layout, mas não somente para isso. Ele nasceu também com outra responsabilidade: garantir acessibilidade, tanto para máquinas, quanto para pessoas.”

Existem três formas de utilização do CSS:

Na primeira forma, conhecida como in-line, o CSS é introduzido por meio de um atributo do HTML.

Na segunda forma, conhecida como CSS interno, define-se previamente as formatações que se pode utilizar em todo o documento.

Na terceira forma, chamada de Método Externo, é a mais utilizada de todas. Em que cria-se um arquivo com extensão “*.css” e nele é adicionado todas as formatações necessárias para a definição do layout do site a ser criado. Na construção do site e feito um link entre os arquivos HTML e o arquivo CSS, permitindo que um único arquivo CSS controle a formatação de quantas páginas HTML forem necessárias. (PRADO et SILVA, 2008)

2.8 PHP

A abreviatura PHP significa “*Hipertext Preprocessor*” (hipertexto pré-processado), que é uma linguagem de programação muito utilizada em desenvolvimentos para Web, sendo trabalhada dentro do código HTML.

O PHP é uma linguagem de programação interpretada pelo servidor que fica embutida junto do código HTML. Quando um arquivo é disponibilizado através da Internet por um servidor de páginas capaz de gerenciar a execução de “*scripts*” PHP algum cliente (“*browser*”) pede que tal arquivo lhe seja enviado, o servidor de páginas checa se existe, dentro deste arquivo, algum código escrito na linguagem PHP. (ESTROZI, 2010 - p. 29)

A sintaxe da linguagem PHP é semelhante à linguagem C e Java, sendo de fácil aprendizado. O objetivo da linguagem PHP é proporcionar aos desenvolvedores a criação de páginas que serão geradas dinamicamente.

O PHP é uma das mais poderosas ferramentas para os profissionais de desenvolvimento e programação voltada para a internet, sendo capaz de reduzir bruscamente a quantidade de tempo necessário para a realização de um projeto. Permitindo que novos mecanismos sejam criados proporcionando um produto final diferenciado e digno de destaque. (ESTROZI, 2010)

2.9 Banco de dados

Banco de dados é um conjunto de dados integrados que se destina a atender vários usuários. Sendo que, o modelo mais usado é o relacional que se utiliza de tabelas que se interagem para representar os dados e as relações entre eles.

Segundo Date, “o banco de dados, por si só, pode ser considerado como o equivalente eletrônico de um armário de arquivamento; ou seja, ele é um repositório ou recipiente para uma coleção de arquivos de dados computadorizados.” (DATE, 2003 – p.3)

O aumento do uso da internet viabilizou a oportunidade de criação de várias interfaces web que disponibilizam vários tipos de serviços online. Como por exemplo: as transações bancárias, bibliotecas virtuais e vários outros tipos de informação que podem ser recuperadas de um banco de dados. Como as interfaces de usuário ocultam os detalhes do acesso ao banco de dados, muitas das pessoas que utilizam dos serviços disponibilizados na internet, não tem consciência de estarem acessando um banco de dados de forma rotineira em suas atividades online na web.

Segundo Heuser (1998 – p.14), “ao conjunto de arquivos integrados que atendem a um conjunto de sistemas dá-se o nome de banco de dados.”

2.9.1 Banco de Dados MySQL

Um pesquisador da IBM chamado de Ted Codd idealizou em um artigo sobre banco de dados relacionais um sistema que pudesse manipular um banco de dados através de comandos em inglês. Esse sistema permitiria armazenar e recuperar uma grande quantidade de informações.

Essa teoria fez com que a IBM organizasse um grupo de pesquisa denominado “System R”, com o objetivo de criar um sistema de banco de dados relacional. Dando origem à criação da linguagem “Structured Query Language” (SQL) ou Linguagem de Consulta Estruturada.

O SQL tornou-se um padrão no mercado de banco de dados relacionais. Mas foi com a empresa Oracle que surgiram os primeiros sistemas de banco de dados construídos nos padrões SQL.

O Sistema R evoluiu para SQL/DS, o qual, posteriormente, tornou-se o DB2. A linguagem criada pelo grupo do Sistema R foi a Structured Query Language (SQL) – Linguagem de Consulta Estruturada. Essa linguagem tornou-se um padrão na indústria de banco de dados relacionais e, hoje em dia, é um padrão ISO (International Organization for Standardization). (ALVES, 2012 – p.27)

Os bancos de dados que utilizam da linguagem SQL e as suas vertentes são os mais utilizados e entre estes os MySQL é o mais utilizado deles.

O MySQL percorreu um longo caminho desde o início de sua existência, onde era um simples armazenador de dados brutos sem grandes tratamentos ou recursos. Hoje, com recursos como Unicode, buscas por texto em qualquer campo (FulltextSearch), triggers, stored procedures, replicações e uma enorme gama de outros recursos, que não deixam absolutamente nada a desejar frente aos grandes e caríssimos bancos de dados concorrentes. (JOBSTRAIBIZER, 2010 – p. 6 e p. 7).

O MySQL surgiu na Suécia na década de 1980 e foi criado por três desenvolvedores; Allan Larsson, David Axmark e Michael Widenius. No intuito de controlar o banco de dados relacionais da própria empresa.

Posteriormente, o MySQL foi comprado pela empresa SUN e depois pela Oracle e hoje, embora tenha sido “open source” desde o início, vários desenvolvedores oficiais são os principais responsáveis por constantes melhorias.

2.10 Modelo Entidade Relacionamento

O modelo Entidade Relacionamento baseia-se na percepção do mundo real, em que vários objetos chamados de entidades que possuem atributos que são as qualidades e características comuns à sua existência. Sendo que, esses objetos se relacionam entre si.

A abordagem Entidade Relacionamento (ER) é uma técnica de modelagem conceitual, sendo representada por um diagrama que é chamado de Diagrama Entidade Relacionamento (DER). A figura 02 apresenta um exemplo de uma Diagrama Entidade Relacionamento

Segundo Heuser, a entidade em um banco de dados representa o conjunto de objetos com as mesmas características e o relacionamento é o conjunto de associações entre as entidades.

No modelo Entidade Relacionamento; a Entidade é o conjunto de objetos da realidade modelada dos quais se deseja manter informações no banco de dados e Relacionamento é o

conjunto de associações entre as entidades.

A técnica de modelagem de dados mais difundida e utilizada é a abordagem entidade-relacionamento (ER). Nesta técnica, o modelo de dados é representado através de um modelo entidade-relacionamento (modelo ER). Usualmente, um modelo ER é representado graficamente, através de um diagrama entidade-relacionamento (DER). A abordagem ER foi criada em 1976 por Peter Chen. Ela pode ser considerada como um padrão de fato para modelagem conceitual. (HEUSER, 1998 – p.22)

2.11 Engenharia de Software

A engenharia de software apoia o desenvolvimento profissional de software em todos os aspectos, inclusive os aspectos técnicos de apoio à especificação, projeto e evolução de programas. Como também, trata de toda a documentação relacionada à operacionalidade do software, para que esteja de acordo com as expectativas do projeto.

Segundo Sommerville (2003, p.4), “a engenharia de software é uma disciplina de engenharia relacionada com todos os aspectos da produção de software, desde os estágios iniciais de especificação do sistema até sua manutenção, depois que este entrar em operação.”

Segundo Sommerville, os atributos essenciais de um software são:

Manutenibilidade: a existência de um planejamento para a escrita do software que vise a possibilidade de evolução, visto que as necessidades dos clientes mudam e se evoluem e precisam constantemente de serem atendidas.

Confiança e proteção: características como confiabilidade, proteção e segurança despertam a confiança dos clientes. Prejuízos no caso de falha do sistema não podem acontecer. Bem como, medidas de segurança precisam de ser tomadas para se evitar o acesso de usuários maliciosos.

Eficiência: além de ser o significado de capacidade de resposta, tempo de processamento, uso de memória. Dentre outras características operacionais um software deve economizar os recursos do sistema como memória, ciclos do processador (clock), e outros.

Aceitabilidade: O usuário precisa aceitar o software que foi projetado para ele. Para que isso aconteça, o software precisa ser compreensível e ser compatível com os outros sistemas já utilizados anteriormente pelo usuário.

Segundo Silva, a Engenharia de Software é uma abordagem sistemática que disciplina e quantifica o desenvolvimento, operação e manutenção de software. É considerada sistemática uma vez que possui processos definidos. Disciplinada tendo em vista que orienta que tais processos sejam seguidos e quantificáveis, uma vez que define um conjunto de medidas a

serem aplicadas no processo.

Engenharia de Software é uma área da informática voltada para a especificação, desenvolvimento e manutenção de sistemas de software usando tecnologias e práticas de sistemas de informação, gerência de projetos e outras disciplinas, buscando organização produtividade e qualidade. (SOUZA, 2008, p. 11)

Segundo Souza, o processo de software possui quatro atividades fundamentais:

Especificação de software: em que a equipe técnica define o software a ser produzido e as devidas restrições de sua operação.

Desenvolvimento de software: etapa que se caracteriza pela elaboração de um projeto, como também é quando o software é programado.

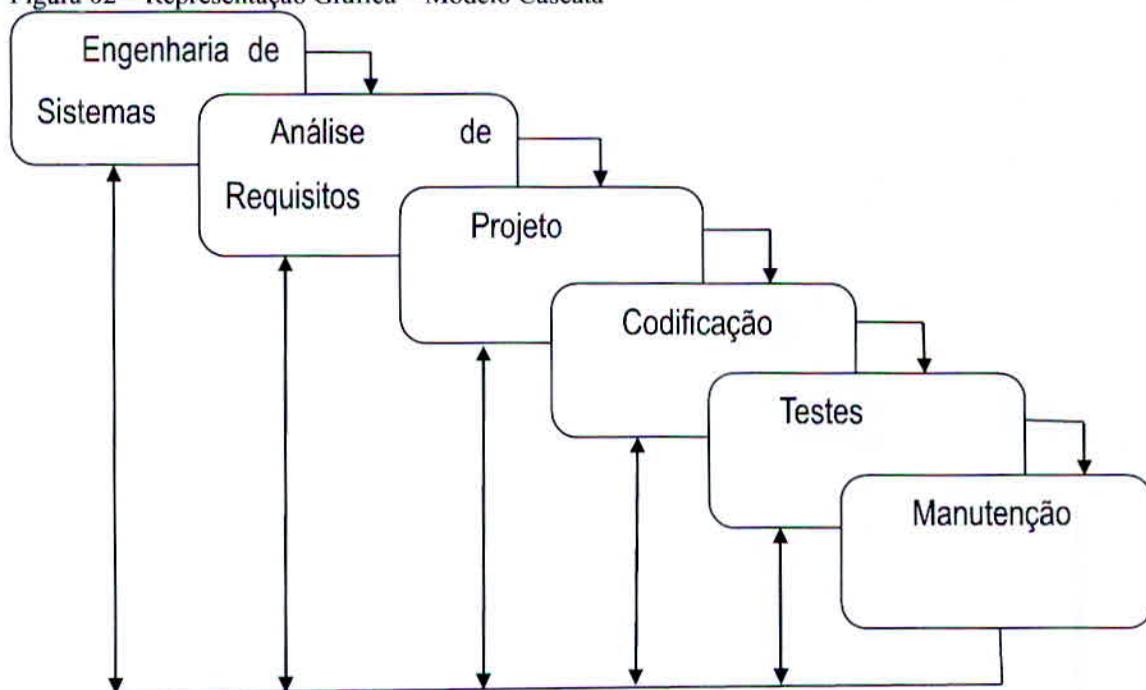
Validação de software: é a fase que garante, por meio de verificações, que o software atenderá as expectativas do cliente.

Evolução de software: é uma etapa mais complexa e que requer uma constante análise de tendências, para constantemente identificar as mudanças de requisitos do cliente e de seu nicho de mercado.

Segundo Souza, na engenharia de software existem abordagens usuais chamadas de, “Ciclos de Vida do Software”, que são realizadas na produção de um sistema, que vão desde a análise do problema até o resultado final. Os principais Ciclos de Vida do Software são: Ciclo de Vida Clássico (também chamado de Cascata), Prototipação, Ciclo de Vida em Espiral.

- **Ciclo de Vida Clássico (Cascata):** é o modelo mais antigo e mais usado na engenharia de software. Este ciclo é modelado em função do ciclo da engenharia convencional e requer uma abordagem sistemática, sequencial ao desenvolvimento de software.

Figura 02 – Representação Gráfica – Modelo Cascata



Fonte: o autor

As atividades do Ciclo de Vida Clássico são:

- Análise e Engenharia de Sistemas – essa atividade envolve a coleta de requisitos em nível de sistema, possui um pouco do desempenho de projeto e da análise de alto nível. Esta visão é essencial quando o software deve fazer interface com outros elementos (hardware, pessoas e banco de dados).

- Análise de Requisitos do Software – é quando acontece o processo de coleta dos requisitos mais intensificado e concentrado especificamente no software. Nessa atividade deve-se compreender o domínio da informação, a função, o desempenho e as interfaces exigidas. Nessa fase os requisitos são documentados e revistos com o cliente.

- Projeto – é considerado a tradução dos requisitos do software para um conjunto de representações, em que são avaliados quando ao nível de qualidade, antes da codificação. Esta atividade concentra-se em quatro atributos do software: Estrutura de Dados; Arquitetura de Software; Detalhes Procedimentais e Caracterização de Interfaces.

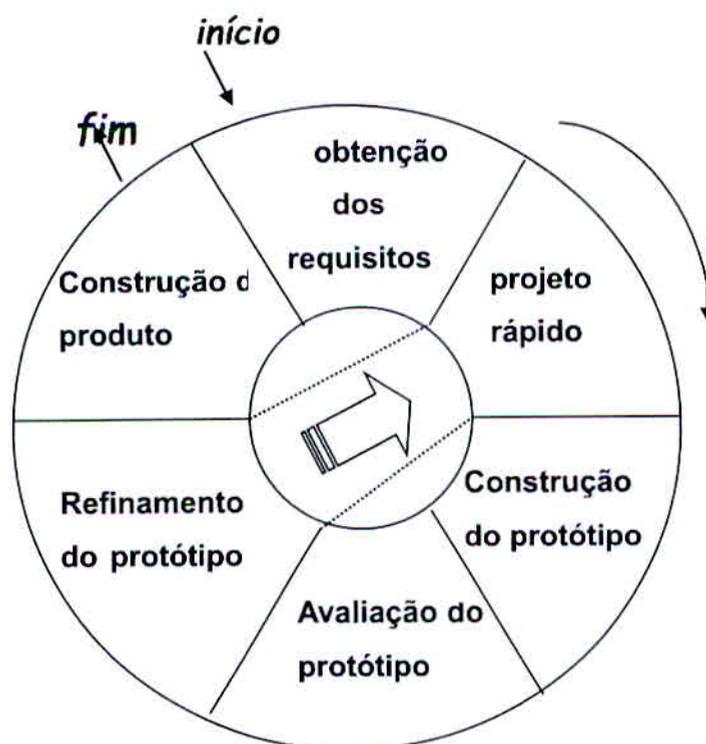
- Codificação – que é a tradução das representações do projeto para uma linguagem de programação, tendo por resultado instruções executáveis pelo computador ou dispositivos móveis.

- Testes – os testes são concentrados nos aspectos lógicos internos do software para garantir que todas as instruções sejam testadas. Como também, nos aspectos funcionais externos, para descobrir erros. Garantindo dessa forma, que a entrada definida produza os resultados planejados.

- Manutenção – nessa atividade acontece a adaptação do software para acomodar as mudanças em seu ambiente externo, para corrigir os erros, bem como para atender as exigências do cliente (“stakeholder”) quanto a acréscimos funcionais ou de desempenho.

▪ Prototipação: nessa abordagem torna-se possível ao desenvolvedor criar um modelo (protótipo) do software a ser construído. O protótipo torna-se um mecanismo para identificar os requisitos de software. Esse tipo de ciclo de vida é apropriado para situações em que o cliente já tenha definido um conjunto de objetivos gerais para o software, mas ainda não identificou os detalhes dos requisitos de entrada, processamento e saída.

Figura 03 – Representação Gráfica - Prototipação



Fonte: (SOUZA, 2008 – p. 30)

As atividades da Prototipação são compreendidas por:

- Obtenção dos requisitos – é quando o desenvolvedor e o cliente definem os objetivos gerais do software identificando quais os requisitos são conhecidos e as áreas que necessitam de definições adicionais.

- Projeto rápido – é definida a representação dos aspectos do software que são visíveis ao usuário (abordagens de entrada e formatos de saída).

- Construção do Protótipo – é quando se faz a implementação do projeto rápido.

- Avaliação do Protótipo – fase em que o cliente e o desenvolvedor avaliam o protótipo.

- Refinamento dos Requisitos – fase em que o cliente e o desenvolvedor refinam os requisitos do software a ser desenvolvido. Nesse ponto ocorre um processo de iteração com a atividade primeira do ciclo (Obtenção dos requisitos), para que todas as necessidades do cliente sejam atendidas e o desenvolvedor compreenda melhor a execução do projeto.

- Construção do produto – após identificados todos os requisitos, o protótipo deve ser descartado e a versão de produção deve ser construída considerando os critérios de qualidade dos requisitos funcionais e não funcionais do produto.

- Ciclo de Vida Espiral: essa abordagem engloba as melhores características do ciclo de vida Clássico e da Prototipação, adicionando um novo elemento que é a Análise de Risco. Como também, segue a abordagem de passos sistemáticos do Ciclo de Vida Clássico incorporando-os numa estrutura iterativa que se aproxima bem mais ao mundo real. Assim como, utiliza a Prototipação em qualquer etapa da evolução do produto como mecanismo de redução de riscos.

As atividades básicas do Ciclo de Vida em Espiral que são sequencialmente alteradas são:

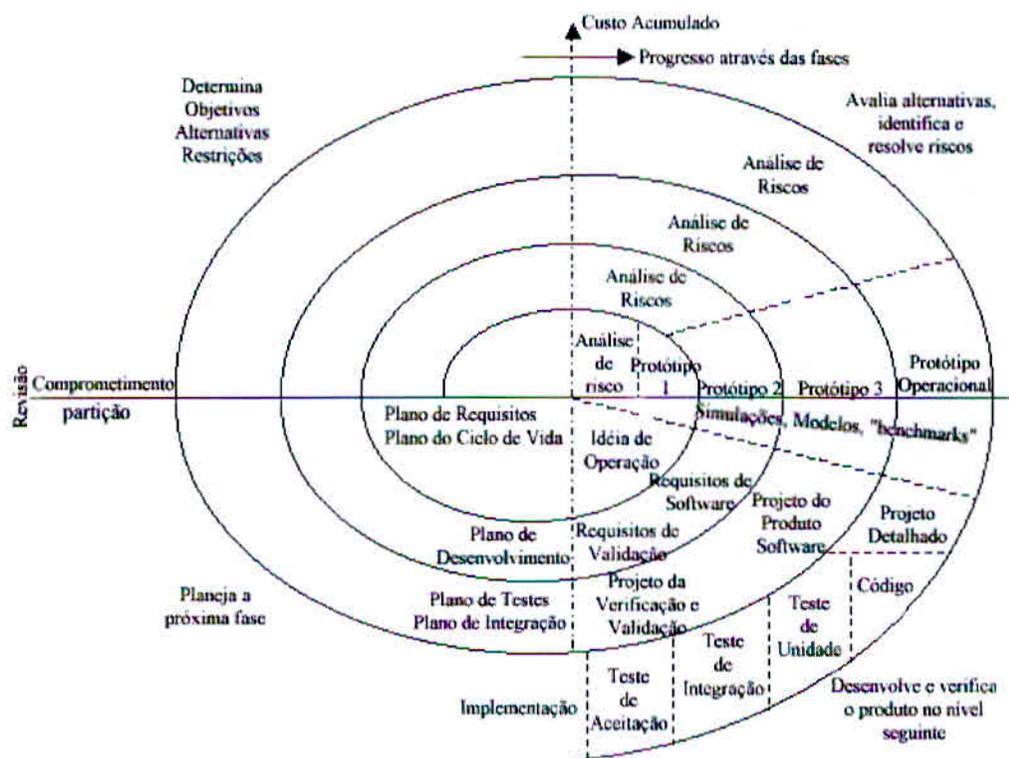
- Planejamento – fase em que acontece a determinação dos objetivos, alternativas e restrições.

- Análise de Risco – são desenvolvidas análises das melhores alternativas para o desempenho do software em que se consuma a identificação e resolução dos riscos.

- Construção – nessa atividade acontece o desenvolvimento do produto (Protótipo) para o nível seguinte.

- Avaliação do Cliente – nessa fase o cliente avalia o produto e o desenvolvedor tem condições de se aproximar os interesses do cliente e estabelecer um planejamento para as novas fases.

Figura 04 – Representação gráfica – Modelo Espiral Completo



Fonte: (SOUZA, 2008 – p. 33)

2.12 Engenharia de Requisitos

A engenharia de requisitos visa identificar os requisitos de um sistema que traduzem as necessidades e desejos dos “stakeholders” (interessados e envolvidos na elaboração do sistema). Documentando o gerenciamento os requisitos no intuito de minimizar o risco de não atender as expectativas das partes interessadas.

Segundo Silva, as quatro principais atividades da Engenharia de Requisitos são:

Elicitação: que é o levantamento dos requisitos, fase em que se utiliza várias técnicas de identificação e detalhamento de requisitos.

Documentação: atividade de descrição dos requisitos em linguagem natural e também em modelos formais (UML, casos de uso, etc).

Validação e negociação: fase em que acontece a garantia da qualidade, a resolução de conflitos e a consistência das informações.

Gerenciamento é uma atividade presente em todas as outras atividades da Engenharia de Requisitos. Mantendo a qualidade da consistência dos dados para que eles possam ser implementados com credibilidade para que possa gerar os resultados esperados.

A engenharia de requisitos é uma subdisciplina da Engenharia de Software. Sua principal responsabilidade é dividida em Produção e Gerência de Requisitos. A Produção de Requisitos é dividida em Levantamento, Registro, Validação e Verificação dos requisitos. A Gerência de Requisitos é dividida em Controle de Mudanças, Gerência de Configuração, Rastreabilidade e Gerência de Qualidade. (SILVA, p. 23)

Segundo Silva, na produção de requisitos:

- Levantamento de requisitos relaciona-se com a obtenção dos requisitos do software em que utiliza-se de algumas técnicas: Entrevistas, Prototipação, JAD e Workshop.

- Registro é a documentação para que possa servir de embasamento para o restante do processo de desenvolvimento. Sendo que, os requisitos são documentados no artefato chamado de Especificação de Requisitos.

- Verificação é fase em que se examina a especificação do software, com o objetivo de assegurar que não há ambiguidade de requisitos, inconsistências ou omissões. Permitindo a detecção e correção de possíveis problemas.

- Validação dos requisitos tem a finalidade de obter um aceite formal do cliente.

Na gerência de requisitos:

- Controle de Mudanças – é uma maneira de controlar as mudanças dos requisitos que ocorrem ao longo do tempo. Utiliza-se uma “*baseline*” de requisitos que demonstra como as mudanças dos requisitos, assim como o que foi introduzido e descartado. O controle de mudanças deve possuir um canal único de aprovação apoiado em um software.

- Gerência de Configuração – estabelece normas, ferramentas e templates que gerenciam satisfatoriamente os itens de configuração do sistema. A cada fase do processo de desenvolvimento um conjunto de itens de configuração é definido (baseline), que é guardado como um registro dos artefatos criados até o momento.

- Rastreabilidade – é acompanhar a vida de um requisito em ambas as direções (partindo do requisito e chegando ao artefato ou partindo do artefato e chegando ao requisito). Como ferramenta para a rastreabilidade destaca-se a Matriz de Rastreabilidade.

- Gerência de Qualidade de Requisitos – garante a qualidade dos requisitos especificados. Para tanto, considera-se alguns critérios de qualidade:

- Correção: os requisitos devem estar presentes no sistema. Os requisitos reais do usuário devem coincidir com os requisitos levantados. Sendo necessário uma boa atividade de validação.

- Não ambiguidade: não gera duplo entendimento.

- Completude: descreve todas as demandas de interesse dos usuários, incluindo requisitos

funcionais, não funcionais, restrições, atributos e interfaces externas.

- **Consistência:** é a garantia que nenhum subconjunto de requisitos entre em conflito com os demais requisitos do sistema.
- **Verificabilidade:** a possibilidade de se verificar por pessoas ou ferramentas, em termos de tempo e custo, se o requisito foi atendido.
- **Modificabilidade:** é a verificação se o estilo de estrutura do requisito permite alterações de forma simples e consistente com os demais requisitos.
- **Ranqueável:** é classificável de acordo com a importância para o cliente e quanto a estabilidade do próprio requisito.
- **Rastreável:** a origem de cada requisito pode ser encontrada.
- **Compreensível:** o requisito precisa de ser facilmente compreendido por usuários e desenvolvedores.

2.13 Requisitos Funcionais

Requisitos funcionais são aqueles que descrevem as funcionalidades ou os serviços disponibilizáveis pelo sistema.

Os atributos dos requisitos funcionais são:

Completude: que é a definição fiel e exata das funcionalidades ou serviços do sistema.

Consistência: os requisitos não podem se contradizer ou ter definições contraditórias.

Alguns exemplos de requisitos funcionais podem ser: pesquisa em banco de dados, cadastramento de dados pessoais em banco de dados, baixa de estoque automática quando alguma venda de produtos aconteça, autenticação por meio de login e senha.

Requisitos funcionais são as declarações de serviços que o sistema deve fornecer, como o sistema deve reagir a entradas específicas e como deve se comportar em determinadas situações. Em alguns casos, os requisitos funcionais também podem estabelecer o que o sistema não deve fazer. (SOMMERVILLE, p.80)

Segundo Silva, “requisitos funcionais descrevem as funcionalidades do software, ou seja, estão diretamente ligados às funções do software, como por exemplo: O sistema deve permitir o cadastro de cliente. O sistema deve permitir a geração de relatório de vendas por período.” (SILVA, 2014, p.22)

2.14 Requisitos Não Funcionais

Os requisitos não funcionais são aqueles que definem as propriedades e as restrições do

sistema.

Requisitos não funcionais são restrições sobre os serviços ou as funções oferecidas pelo sistema. Eles incluem restrições de timing, restrições sobre o processo de desenvolvimento e padrões. Os requisitos não funcionais aplicam-se frequentemente, ao sistema como um todo. Em geral, eles não se aplicam às características ou serviços individuais. (SOMMERVILLE, p.80 – 81)

Os requisitos não funcionais são classificados em:

Requisitos do produto: que são aqueles que especificam o comportamento do software, por exemplo:

- Usabilidade: é a característica do software de ser facilmente entendível e manipulável.

Bem como a de atender a uma base significativa de usuários.

- Eficiência: que é a possibilidade em alcançar alto rendimento em baixo tempo de retorno.

- Escalabilidade: a capacidade de usar recursos à medida que forem acrescentados.

- Extensibilidade: adapta-se bem a novas tecnologias e fornece capacidades para executar tarefas que vão além do projeto original.

- Robustez: é a característica de ser tolerante a falhas que possam ser ocasionadas por erros isolados de hardware e se caso aconteça alguma falha, o sistema minimiza as perdas de trabalho e danos ao hardware.

- Portabilidade: é a capacidade do sistema de operar em vários tipos de configuração de hardware.

- Segurança: o sistema possui recursos que impedem que usuários e softwares acessem a serviços e recursos sem autorização.

Requisitos organizacionais: que são os requisitos que determinados pelos padrões do cliente. Como estatutos empresariais, regimentos internos e específicos a uma determinada empresa, políticas e procedimentos.

Requisitos externos: que são os que se embasam em fatores externos ao sistema. Como legislações pertinentes a situações específicas de um determinado local ou aplicabilidade. Procedimentos jurídicos que alguma funcionalidade a ser disponibilizada pelo sistema tenha que considerar: os devidos parâmetros concernentes a jurisdição local e outros com características similares.

2.15 Regras de Negócio e Restrições

As regras de negócio podem ser automatizadas, mas elas existem sem a necessidade de

um sistema ou software. Até mesmo por pertencerem ao domínio do próprio negócio. Devido a essa particularidade, uma regra de negócio não pode ser considerada como um requisito funcional do sistema ou software. Como por exemplo, o setor administrativo de uma empresa pode estabelecer limites de compra mínimos ou que a venda a prazo seja feita somente a clientes em que as suas compras atinjam parcelas com valor mínimo. E esse fato iria ser possível de acontecer independente da atuação de um software.

Regras de negócio são restrições que controlam um procedimento operacional do negócio. São políticas impostas pela administração da empresa ou por leis governamentais, portarias, etc. Seriam como um requisito não funcional externo.

Segundo Prado, o conceito “regras de negócio” começou a ser desenvolvido nos anos 1990, com a criação do “*GUIDE Business Rule Project*”, que posteriormente evoluiu para a criação do Business RuleGroup. Sua declaração diz que regras de negócio definem a estrutura e controlam a operação das empresas. Representam uma mudança na forma como descrevemos uma organização, deixando de lado o foco nos dados que a empresa usa e a organização dos processos que executa, para se concentrar nas restrições sob as quais as empresas operam. Uma regra de negócio é uma declaração que, explicitamente, define ou restringe um aspecto do funcionamento da empresa. Sob essa ótica, nada existe se não estiver definido em uma regra de negócio, e tudo é permitido se não houver uma restrição estabelecida em uma regra de negócio.

2.16 Modelagem de Software com UML

UML é uma linguagem de semântica semiformal e dinâmica com sintaxe abstrata e regras bem definidas. Capturando a estrutura de sistemas orientados a objeto antes da determinação do código. Sua apresentação se dá com a utilização de diagramas de classe demonstrando as construções de herança, associações, composição e agregação. Além de outros o diagrama de sequência e o diagrama de casos de uso.

UML (Unified Modeling Language) é uma família de notações gráficas, apoiada por um metamodelo único, que ajuda na descrição e no projeto de sistemas de software, particularmente daqueles construídos utilizando o estilo orientado a objetos. (FOWLER, 2005, p.25).

A UML nasceu da unificação das muitas linguagens gráficas de modelagem orientadas a objetos que floresceram no final dos anos oitenta, início dos noventa. Desde sua aparição, em 1997, ela fez com que essa torre de Babel fosse resolvida. (FOWLER, 2005)

2.17 Diagrama de Casos de Uso

O diagrama de casos de uso mostra todos os trâmites dos requisitos funcionais de um sistema, demonstrando quais são os atores e toda a extensão das ações dos atores que se adequam às funcionalidades correlatas da macroatividade do sistema.

A melhor maneira de pensar um diagrama de caso de uso é como um sumário gráfico do conjunto de casos de uso. Ele também [...] mostra o limite do sistema e as interações com o mundo exterior. O diagrama de casos de uso mostra os atores, os casos de uso e os relacionamentos entre eles: Quais atores realizam quais casos de uso. Quais casos de uso incluem outros casos de uso. (FOWLER, 2005, p. 107)

Segundo Silva, o diagrama de caso de uso auxilia a comunicação entre os analistas e os clientes. Bem como, descreve um cenário que mostra as funcionalidades do sistema do ponto de vista do usuário. Descrevendo o sistema, seu ambiente e a relação entre os dois. Veja a figura 05.

2.18 Especificação de Casos de Uso

Especificação de caso de uso é a elaboração de um documento que descreve um caso de uso contendo um conjunto de definições relacionadas ao caso de uso e apresentadas segundo um organizacional específico.

Os casos de uso é uma técnica para captar os requisitos funcionais de um sistema. Eles servem para descrever as interações típicas entre os usuários de um sistema e o próprio sistema, fornecendo uma narrativa sobre como o sistema é utilizado. (FOWLER, 2005, p. 104)

Segundo Spínola, os parâmetros de uma especificação de casos de uso são os seguintes:

Objetivos: descrevem os objetivos do caso de uso.

Requisitos: indica qual requisito funcional que o caso de uso está associado.

Atores: define os atores associados ao caso de uso, sendo que o ator é qualquer entidade externa que interage com o sistema do caso de uso.

Prioridade: informação identificada junto ao usuário que auxilia na definição dos casos de uso que serão contemplados em cada iteração do desenvolvimento do software.

Pré-condições: neste campo se informa as condições que devem ser atendidas para que o caso de uso possa ser executado.

Frequência de uso: informação identificada junto ao usuário que auxilia na definição dos casos de uso que serão contemplados em cada iteração do desenvolvimento do software.

Criticalidade: informação identificada junto ao usuário que auxilia na definição dos casos de uso que serão contemplados em cada iteração do desenvolvimento do software.

Condição de entrada: neste campo é definida qual ação do autor dará início à interação com o caso de uso em questão.

Fluxo Principal: é onde se descreve os passos entre o ator e o sistema. O fluxo principal é o cenário que mais acontece no caso de uso e/ ou o mais importante.

Fluxo alternativo: é o caminho alternativo tomado pelo caso de uso a partir do fluxo principal, ou seja, dada uma condição de negócio o caso de uso seguirá outro cenário que não o principal caso essa condição seja verdadeira.

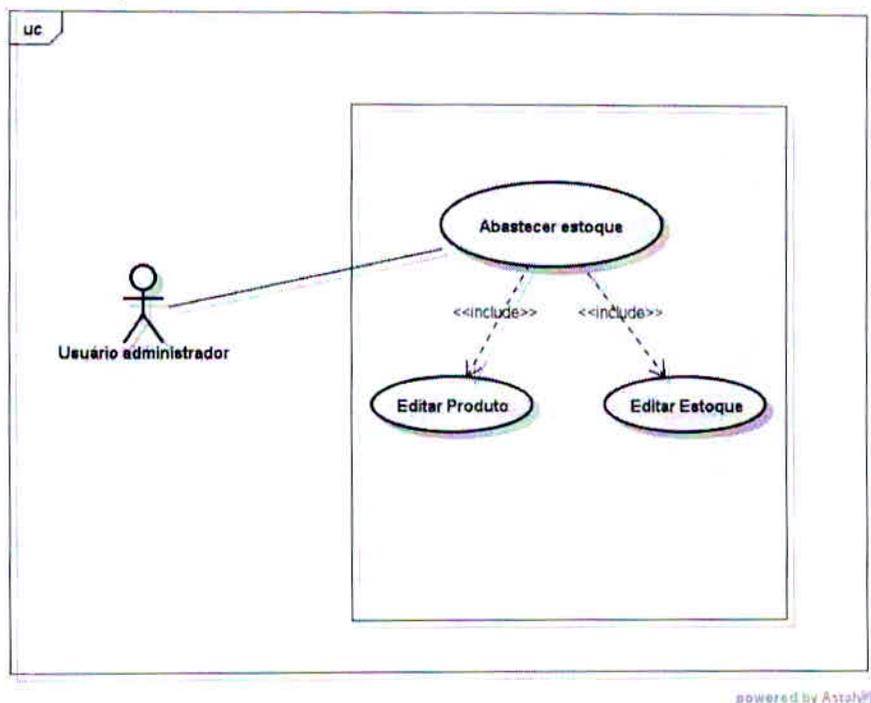
Extensões: nesta seção se coloca todos os casos de uso que estendem o caso de uso base e em quais pontos são chamados dentro dos fluxos de eventos.

Pós-condições: neste campo deve-se informar o estado em que o sistema (ou entidade manipulada no caso de uso) depois que o caso de uso for executado.

Regras de negócio: nesta seção se descreve todas as regras funcionais que o caso de uso deve cumprir durante sua execução.

Cada caso de uso tem um ator principal, que pede ao sistema para que execute um serviço. O ator principal é aquele cujo objetivo o caso de uso está tentando satisfazer e normalmente (mas nem sempre) é o iniciador do caso de uso. Podem existir outros atores com os quais o sistema se comunica enquanto executa o caso de uso. Eles são conhecidos como atores secundários. (FOWLER, 2005)

Figura 05 – Diagrama de Caso de Uso



Fonte: o autor.

2.19 Diagrama de Classes

É um diagrama que demonstra um conjunto de classes e seus relacionamentos, sendo o diagrama utilizado para modelagem orientada a objetos.

No diagrama de classes, as classes são representadas por retângulos em que são apresentados o nome da classe, os atributos e os métodos. Veja um exemplo na figura 06.

As classes se relacionam conforme as regras de negócio adquirindo relacionamentos que são conhecidos por:

- Associação: é uma relação que permite especificar quais objetos de uma classe se relacionam objetos de outra classe, sendo que é importante saber para cada objeto quais objetos lhe estão associados. Exemplo: um cliente pode ter várias contas bancárias, mas uma conta é relacionada a um cliente.
- Agregação: é um tipo de associação que se utiliza quando se pretende representar a noção de Todo/Parte. Exemplo: uma holding possui várias empresas. Mas se uma empresa sair de uma holding, ela continua existindo.
- Composição: é um caso especial de agregação em que os componentes apenas existem no contexto do Todo. Ou que, um objeto de uma classe (composição) inclui um conjunto de objetos de outra classe (componentes) e quando se remove um objeto composto todos os seus

2.20 Segurança da informação

Por ser a informação um bem precioso que necessita de cuidados com a sua segurança. Uma organização ou empresa que queira preservar computadores, softwares, aplicações, componentes de rede, sistemas operacionais, servidor de banco de dados, e qualquer dispositivo que faça parte de seu sistema de informações. Essa organização precisa estar devidamente resguardada por ferramentas de segurança, para que os usuários maliciosos, os curiosos, os vírus ou qualquer outra ameaça não altere ou danifique dados ou equipamentos. Proporcionando muitas vezes prejuízos irreparáveis.

Segundo Campos, um sistema de segurança da informação baseia-se em três princípios básicos:

Confidencialidade: garantindo que somente pessoas explicitamente autorizadas podem ter acesso a informação.

Integridade: garante que quanto a informação é acessada ela está completa, sem alterações indesejadas, e portanto confiável.

Disponibilidade: garante às pessoas autorizadas a acessibilidade da informação sempre que lhes for necessário.

Segundo Piurcosky, além destes três aspectos, ainda temos:

Autenticação: para garantir que um usuário é de fato quem alega ser.

Não-repúdio: capacidade do sistema de provar que um usuário executou uma determinada ação.

Legalidade: garantia de que o sistema esteja aderente à legislação pertinente.

Privacidade: capacidade de um sistema de manter anônimo um usuário, impossibilitando o relacionamento entre o usuário e suas ações.

Auditoria: capacidade do sistema em auditar tudo o que foi realizado pelos usuários, detectando fraudes ou tentativas de ataque.

2.20.1 Segurança e o Ciclo de Vida da Informação

A informação precisa ser trabalhada com segurança desde de a sua obtenção até o seu descarte. Visando sempre a autenticidade, a utilidade e o valor da informação para uma devida organização.

Segundo Piurcosky, no levantamento de informações para implantação ou desenvolvimento de um sistema é importante saber como a informação fluirá na Organização.

A capacidade em identificar as necessidades e os requisitos das informações é que determina o seu ciclo de vida.

Segundo Lyra, os componentes do ciclo de vida da informação são os seguintes:

- Identificação das necessidades e dos requisitos – devido a tendência ao alto grau de especificidade informacional é imprescindível conhecer bem as necessidades e as características que envolvem as situações que determinam a aplicabilidade de uma informação.

- Obtenção: etapa em que se desenvolve procedimentos para captura e recepção da informação proveniente de fontes externas. Considerando-se quanto a integridade, completude e compatibilidade com os requisitos já levantados.

- Tratamento: Antes da utilização de uma informação ela precisa ser classificada, formatada, organizada. Disponibilizando posteriormente fácil acessibilidade, integridade e confidencialidade quando necessário.

- Distribuição: Fase da entrega da informação ao usuário interessado e de vários cuidados com a segurança da rede de distribuição.

- Uso: Verificação da utilidade e do valor da informação para a organização. Aplica-se os conceitos de disponibilidade, integridade e confidencialidade.

- Armazenamento: Fase em que se determina a possibilidade de reutilização da informação e a necessidade de armazenamento da mesma. Prevalecendo a preocupação com a integridade, disponibilidade e confidencialidade da informação.

- Descarte: Fase auxiliar da Gestão da Informação em que se reavalia a utilidade, como também se ela já está obsoleta. E quando necessário ela é descartada, obedecendo a normas legais, políticas operacionais e exigências internas. Gerando um novo tipo de lixo: o lixo digital.

2.20.2 A Segurança no Contexto da Governança de Tecnologia da Informação (TI)

A necessidade de controle dos processos que estabelecem a utilização das informações, bem como dos sistemas que se utilizam dos mais variados recursos de TI. Evidenciou o surgimento de padrões de controle que são modelos de referência para gestão de Tecnologia da Informação.

Segundo Piurcosky, o *“Control Objectives for Information and Related Technology”* (COBIT) é um guia dirigido para a gestão da Tecnologia da Informação, sendo recomendado pelo *“Information System Audit and Control Foundation”* (ISACA).

Esse guia possui uma série de recursos que são modelos de referência para a gestão. Incluindo controle de objetivos, mapas de auditoria, ferramentas para a implementação e técnicas de gerenciamento.

O COBIT abrange a quatro domínios:

- Planejar e Organizar – os objetivos de alto nível desse domínio são: Definir um Plano Estratégico de TI e orientar; definir Arquitetura da Informação; determinar o Gerenciamento Tecnológico, definir os processos de TI; Gerenciar o investimento de TI; comunicar os objetivos do gerenciamento e orientar; gerenciar os recursos humanos de TI; gerenciar a qualidade; estimar e gerenciar os riscos de TI e gerenciar projetos.

- Adquirir e Implementar – os objetivos: Identificar soluções automatizadas; adquirir e manter o software de aplicação; habilitar operação e uso; obter os recursos de TI, gerenciar mudanças; instalar e credenciar soluções e mudanças.

- Entregar e dar Suporte – objetivos de controle: Definir e gerenciar os níveis de serviço; gerenciar serviços de terceiros; gerenciar performance e capacidade; assegurar serviço contínuo; assegurar a Segurança do Sistema; identificar e alocar recursos; treinar usuários; gerenciar serviços de escritório e incidentes; gerenciar a configuração; gerenciar problemas; gerenciar dados; gerenciar o ambiente físico e gerenciar operações.

- Monitorar e Avaliar – os objetivos desse domínio são: Monitorar processos; assegurar a avaliação dos controles internos; obter avaliação independente e prover auditoria independente.

3 MATERIAL E MÉTODOS

O método experimental foi o mais aplicável a esse tipo de trabalho. Visto que, as etapas comuns a esse tipo de método científico foram devidamente seguidas, no intuito de levantar hipóteses e propor soluções a questões levantadas com o desenvolvimento da aplicação de formas que possam solucionar as questões observadas.

Com a etapa da observação foi desenvolvida uma pesquisa de mercado para avaliar a possibilidade de uso de uma aplicação web, levando-se em consideração as hipóteses que levantam os questionamentos sobre a positividade dos resultados a serem obtidos.

A coleta de dados, em que a análise de requisitos torna possível a fundamentação estrutural da criação de uma aplicação web que atenda às expectativas dos clientes e as necessidades do mercado a que esse trabalho científico se destina, foi desenvolvida com sucesso.

A descrição da pesquisa está formalizada expondo todos os passos e as características gerais do trabalho científico. Desde as abordagens iniciais até a elaboração e execução da aplicação.

Foram feitas comparações com outras aplicações existentes no mercado, no intuito de se verificar semelhanças com a aplicação web que fora desenvolvida.

Foi desenvolvida uma análise desde o planejamento até a execução, em que todas as etapas foram analisadas, bem como todas as funcionalidades da aplicação, para que se possa comprovar, desde de sua origem até a sua conclusão, que a criação da aplicação teve por objetivo a comprovação das hipóteses levantadas.

3.1 Descrição

O trabalho teve o seu início com a busca das informações que pudessem estruturar uma fundamentação para o desenvolvimento da pesquisa. Foi feita a análise de requisitos do sistema e a coleta de dados com a utilização de entrevistas. Como também, uma pesquisa de mercado para se avaliar o grau de interatividade dos comerciantes com o “e-commerce”, confrontando com a disponibilidade e aceitabilidade do uso de dispositivos móveis para a aquisição de produtos. Foram executadas também algumas comparações com outras aplicações, sistemas, aplicativos e sites de e-commerce. No intuito de se pesquisar tendências, aparências de interfaces e potencial de interatividade.

Posteriormente foi feita a modelagem do software web para que se pudesse determinar a arquitetura e toda a estrutura do software. Com a definição de todas as ações, ferramentas, informações, funcionalidades e recursos disponíveis aos usuários e o grau de interação entre eles. Foram utilizados os conceitos de Entidade Relacionamento e os Diagramas de Entidade e Relacionamento (DER), os diagramas de Casos de Uso e os Diagramas de Classe (UML).

Após a modelagem, deu-se início a fase de se desenvolver o que foi planejado. Programando com a utilização dos códigos das linguagens de programação; PHP, HTML5 principalmente. E utilizando-se do paradigma de Programação Orientada a Objetos para a implementação da aplicação web.

Houve a fase final, que se entende pela verificação, para se confirmar quanto aos resultados apresentados, se realmente, estavam alcançando aos objetivos planejados, para que a viabilidade do software fosse atendida.

3.1.1 Entrevista

Para coletar e buscar dados e informações para o projeto foram utilizadas entrevistas.

Primeiramente foi marcada uma reunião com o gerente da empresa Cocatrel em Três Pontas, Minas Gerais. Situação tal que, previamente foi planejada uma sequência de perguntas que compõem um interrogatório que estrategicamente auxiliaram a obter os requisitos suficientes para o planejamento e o desenvolvimento de uma aplicação web que fosse satisfatória em termos de expectativas e resultados por parte desse fornecedor.

Como também, de posse das informações adquiridas com o levantamento dos requisitos, através das entrevistas com a gerência comercial, foi elaborado um planejamento de atividades para melhor conhecer o mercado prospectado, para que o desenvolvimento da aplicação web pudesse ter um melhor respaldo estrutural que fundamentasse atividades concernentes ao perfil do usuário e do tipo de atividade comercial analisada. De formas que, uma pesquisa de mercado voltada ao comércio do café, para que pudesse ser verificada a viabilidade e a aceitação de um produto web interagindo ao conceito de negociação digital utilizando a internet (e-commerce B2B), por meio de dispositivos móveis e ou computadores com a utilização de uma aplicação web que lhes possa proporcionar um diferencial em termos de praticidade, versatilidade e desempenho foi desenvolvida com a utilização de entrevistas devidamente planejadas e previamente agendadas.

A definição do escopo e de uma estrutura de desenvolvimento da aplicação web e a viabilidade de adaptações que possam torná-la mais interessante aos clientes foram melhor

identificadas, tendo por base os requisitos devidamente analisados, junto aos comerciantes, com a utilização dos recursos da prática de entrevista já supracitados.

3.1.2 Modelagem

A criação de um modelo que representasse a aplicação web em todos os níveis e sentidos foi possível com a utilização de alguns programas:

- O “brmodelo” foi utilizado para a modelagem do banco de dados com a utilização dos Diagramas Entidade Relacionamento, determinando o esquema conceitual e o lógico para o banco de dados utilizado na aplicação.
- O “ArgoUML” foi utilizado para a modelagem do relacionamento entre as classes que formam os objetos que compõem a estrutura da aplicação web. Como também o “ArgoUML” foi utilizado para modelar o relacionamento dos usuários (atores) com os requisitos funcionais (funcionalidades) da aplicação web.

3.1.3 Desenvolvimento

Para o desenvolvimento do site foi utilizado o framework do Netbeans para programar com HTML5, JavaScript e PHP. Foi desenvolvido os layouts do site utilizando as folhas de estilo CSS trabalhadas no mesmo ambiente. Utilizou-se também o Netbeans para programar com Bootstrap para desenvolver o aplicativo, visto que o Android estava gerando vários conflitos entre as linguagens e as versões.

Foi utilizado o Xampp como servidor de banco de dados utilizando os comandos da linguagem do MySQL.

4 RESULTADOS

Os resultados estão previstos em apresentarem o que está determinado no protótipo desenvolvido abaixo. Seguindo as modelagens do banco de dados (Diagramas Entidade Relacionamento) e dos requisitos funcionais (Diagramas de Estudo de Casos).

Os “wireframes” apresentados demonstram as partes mais básicas do projeto em que:

- A página inicial propicia ao usuário a possibilidade ou de se logar e efetuar a suas compras ou de se cadastrar para vir a fazer parte do banco de dados da empresa.

- As páginas de cadastro do cliente e do administrador para que os mesmos se cadastrem.
- A página de compras, que apresenta ao usuário o menu de produtos e a possibilidade de comprá-los.

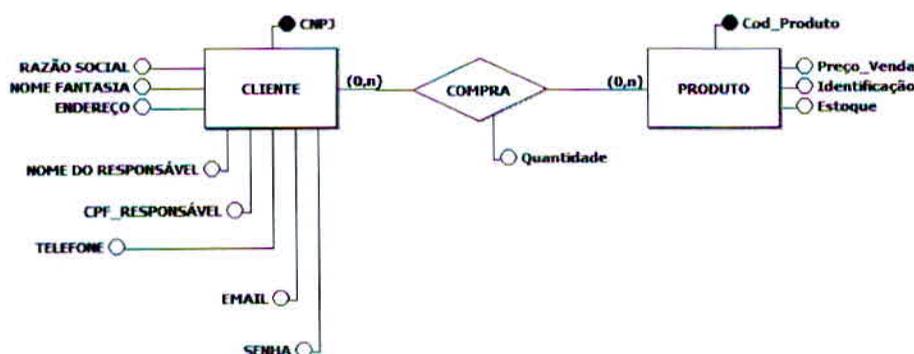
- A página de abastecimento do estoque que possibilita ao administrador editar, estoque, produto, verificar a participação de cada produto no mix da empresa e a geração de relatórios mensais do movimento do estoque.

As tabelas mostram alguns dos resultados mais básicos, como pesquisa do estoque e a porcentagem de participação de cada produto no lucro da empresa.

4.1 Diagrama Entidade Relacionamento - Conceitual

O diagrama entidade relacionamento abaixo mostra as principais entidades da aplicação se relacionando na ação principal do sistema que é a negociação de compra. Com suas devidas chaves primárias e todos os registros que compõem as tuplas.

Figura 07 – Diagrama Entidade Relacionamento - Conceitual

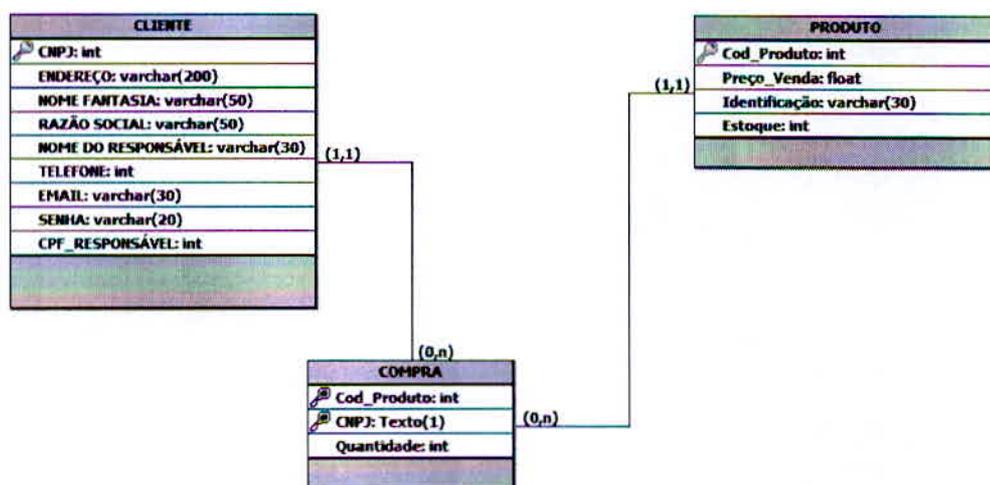


Fonte: o autor

4.2 Diagrama Entidade Relacionamento - Lógico

O diagrama abaixo foi gerado a partir do conceitual e apresenta, além de todos os possíveis registros, as devidas cardinalidades entre as entidades geradas, como também as chaves primárias e estrangeiras de cada uma delas.

Figura 08 – Diagrama Entidade Relacionamento – Lógico.



Fonte: o autor

4.3 Especificação do Caso de Uso – Cadastrar Cliente

CASO DE USO – CADASTRAR CLIENTE

DADOS

Identificação:	Projeto:	Versão:
Caso de uso Cadastrar Cliente	Aplicação Web Compra Café	Ap.01
Cliente:		
Cocatrel – Cooperativa dos Cafeicultores da Zona de Três Pontas		
Descrição Sucinta:		
Este caso de uso permite o auto cadastramento do cliente no aplicativo.		
Atores:		
<ul style="list-style-type: none"> ○ Usuário cliente. 		
Pré-condições:	Pós-condições:	
<ul style="list-style-type: none"> ○ Usuário acessa a homepage 		

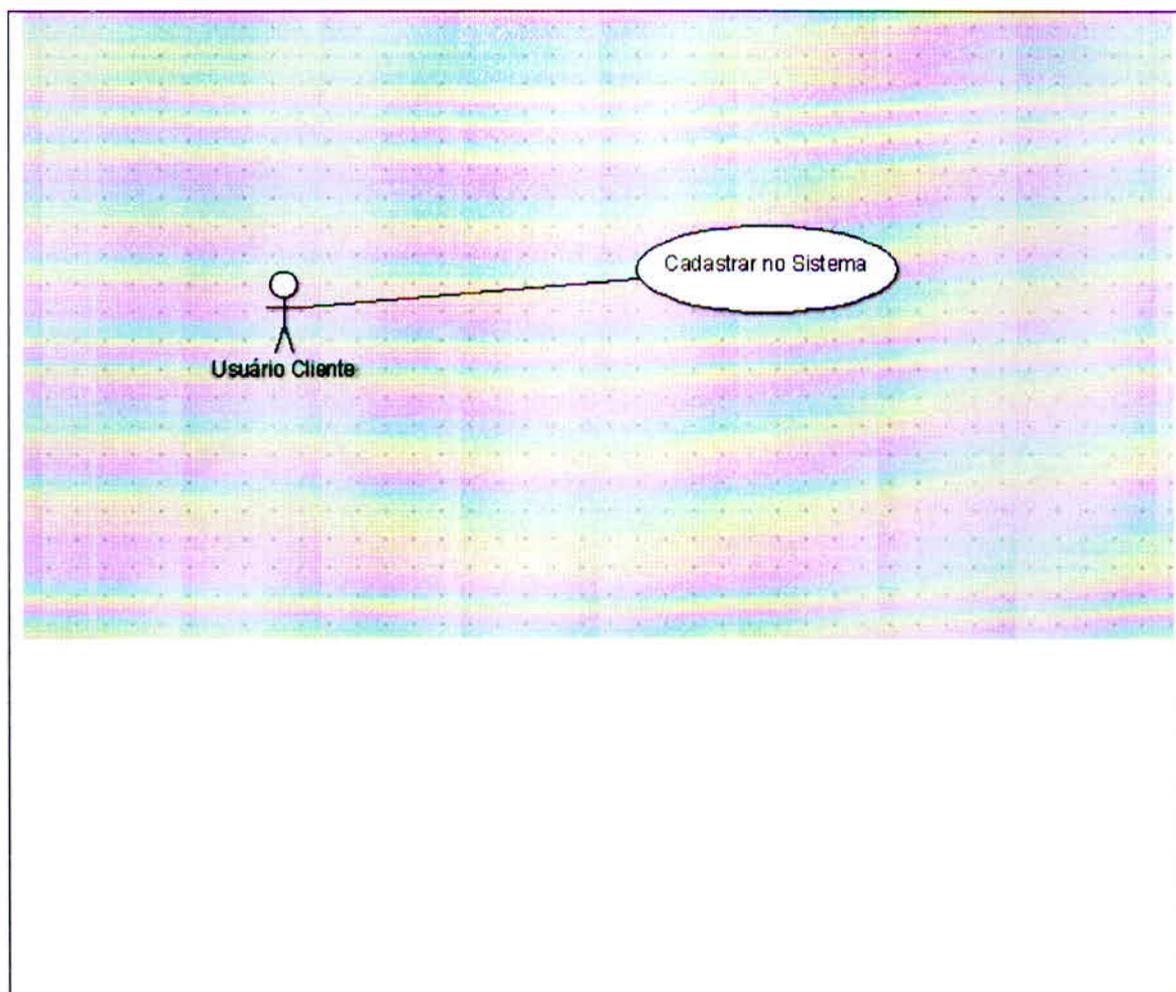
	<ul style="list-style-type: none"> ○ Haverá um usuário cliente cadastrado
Casos de uso Associados:	
<ul style="list-style-type: none"> ○ Não tem 	

FLUXO BÁSICO	
(F1)	Este caso de uso se inicia quando o usuário seleciona o link <u>Cadastrar</u>
(F2)	O sistema disponibiliza os campos do cadastro a serem preenchidos pelo cliente.
(F3)	<p>O novo usuário cliente informa os seguintes dados ao sistema:</p> <ul style="list-style-type: none"> ○ CNPJ (Obrigatório) ○ Razão Social (Obrigatório) ○ Nome Fantasia ○ Endereço completo (Obrigatório) ○ Telefone (Obrigatório) ○ Nome do responsável pelas compras (Obrigatório) ○ CPF do responsável (Obrigatório) ○ E-mail (Obrigatório) ○ Senha (Obrigatório)
(F4)	O usuário inclui os dados pressionando o botão <u>Salvar</u>
(F5)	O sistema valida os dados
(F6)	O sistema inclui o novo cliente e envia mensagem informando que o novo usuário cliente foi gravado e incluído com sucesso.
(F7)	O usuário continua o processamento, clicando no botão <u>OK</u>
(F8)	O sistema encerra o caso de uso, disponibilizando ao novo usuário a possibilidade de comprar os produtos ou sair do sistema.

FLUXO DE EXCEÇÕES	
(E1)	<p><i>CNPJ – Obrigatório</i></p> <p>Se o usuário não informar o CNPJ, o sistema exibirá uma mensagem de alerta informando a necessidade de preenchimento dos campos com asterisco (*).</p>

(E2)	<i>Razão Social – Obrigatório</i> Se o usuário não informar a Razão Social, o sistema exibirá uma mensagem de alerta informando a necessidade de preenchimento dos campos com asterisco (*).
(E3)	<i>Endereço completo – Obrigatório</i> Se o usuário não informar o Endereço completo, o sistema exibirá uma mensagem de alerta informando a necessidade de preenchimento dos campos com asterisco (*).
(E4)	<i>Telefone – Obrigatório</i> Se o usuário não informar o Telefone, o sistema exibirá uma mensagem de alerta informando a necessidade de preenchimento dos campos com asterisco (*).
(E5)	<i>Nome do responsável – Obrigatório</i> Se o usuário não informar o nome do responsável, o sistema exibirá uma mensagem de alerta informando a necessidade de preenchimento dos campos com asterisco (*).
(E6)	<i>CPF do responsável – Obrigatório</i> Se o usuário não informar o CPF do responsável, o sistema exibirá uma mensagem de alerta informando a necessidade de preenchimento dos campos com asterisco (*).
(E7)	<i>E-mail – Obrigatório</i> Se o usuário não informar o e-mail, o sistema exibirá uma mensagem de alerta informando a necessidade de preenchimento dos campos com asterisco (*).
(E8)	<i>Senha – Obrigatório</i> Se o usuário não informar a senha, o sistema exibirá uma mensagem de alerta informando a necessidade de preenchimento dos campos com asterisco (*).

DIAGRAMA



4.4 Especificação do Caso de uso – Cadastrar usuário administrador

CASO DE USO – CADASTRAR USUÁRIO ADMINISTRADOR

DADOS		
Identificação:	Projeto:	Versão:
Caso de uso cadastrar usuário administrador	Aplicação Web Compra Café	Ap. 01
Cliente:		
Cocatrel – Cooperativa dos Cafeicultores da Zona de Três Pontas		
Descrição sucinta:		
Este caso de uso permite o cadastramento do usuário administrador.		
Atores:		

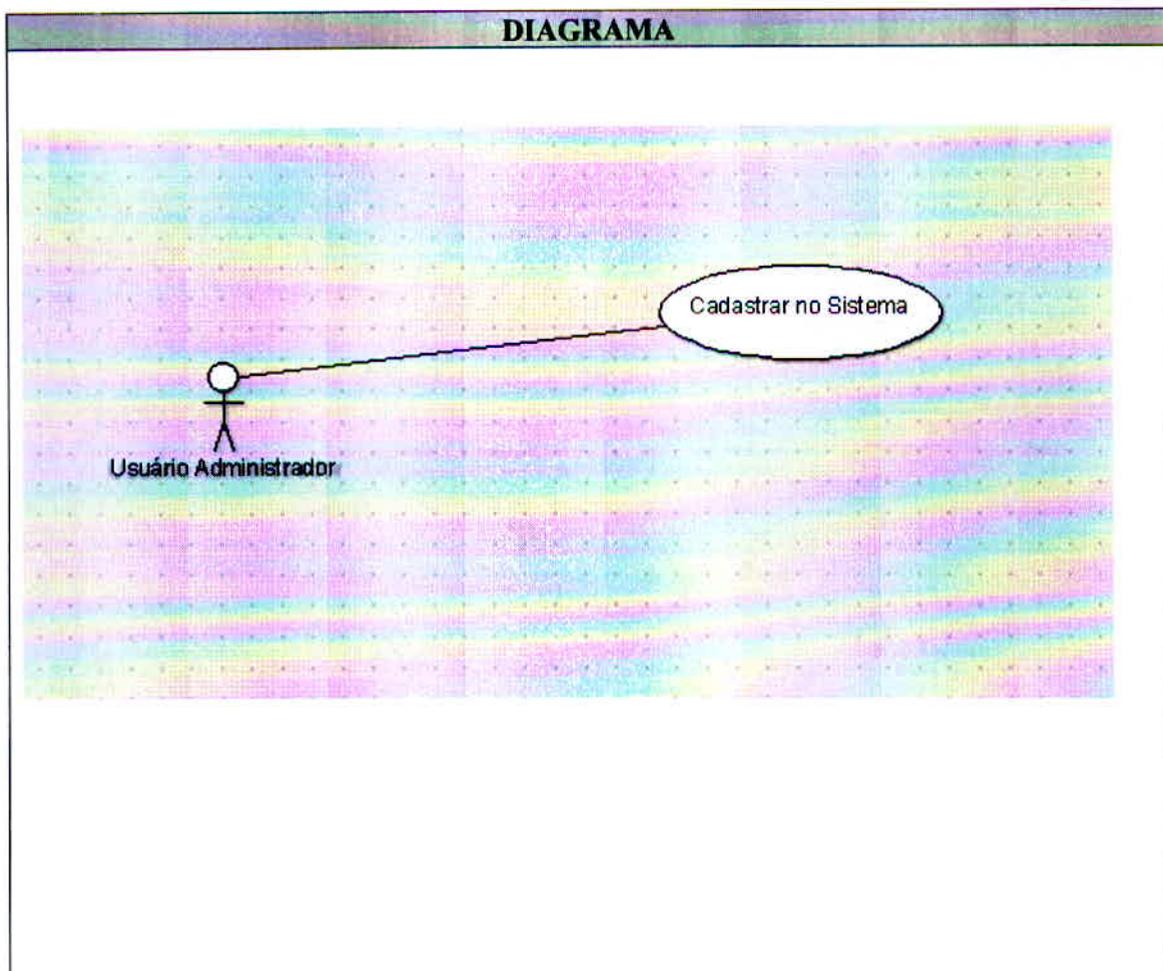
<ul style="list-style-type: none"> ○ Usuário administrador do sistema 	
Pré-condições: <ul style="list-style-type: none"> ○ Usuário administrador logado no sistema 	Pós-condições: <ul style="list-style-type: none"> ○ Haverá um usuário administrador cadastrado no sistema
Casos de uso associados: <ul style="list-style-type: none"> ○ Não tem 	

FLUXO BÁSICO	
(F1)	Este caso de uso se inicia quando o usuário administrador, logado no sistema, seleciona o botão Cadastrar .
(F2)	O sistema disponibiliza os campos do cadastro a serem preenchidos pelo usuário administrador
(F3)	O novo usuário administrador informa os seguintes dados obrigatórios ao sistema: <ul style="list-style-type: none"> ○ Identificador (Pré-estabelecido) ○ Nome completo ○ Cpf ○ Cargo ○ Departamento ○ E-mail ○ Senha
(F4)	O usuário clica no botão “Confirmar”.
(F5)	O sistema confirma o cadastramento enviando a mensagem de “cadastramento realizado com sucesso”.
(F6)	O usuário continua o processamento, clicando no botão <u>OK</u>
(F7)	O caso de uso finaliza com o sistema disponibilizando ao usuário a tela inicial do sistema com as opções de: cadastrar produto, abastecer o estoque, verificar o estoque, verificar relatório de vendas, verificar o agendamento das entregas.

FLUXO DE EXCEÇÕES	
(E1)	<i>Todos os campos obrigatórios</i> Todos os campos são obrigatórios e ao não preenchimento exibirá uma mensagem de

alerta informando a necessidade de preenchimento dos campos com asterisco (*)

DIAGRAMA



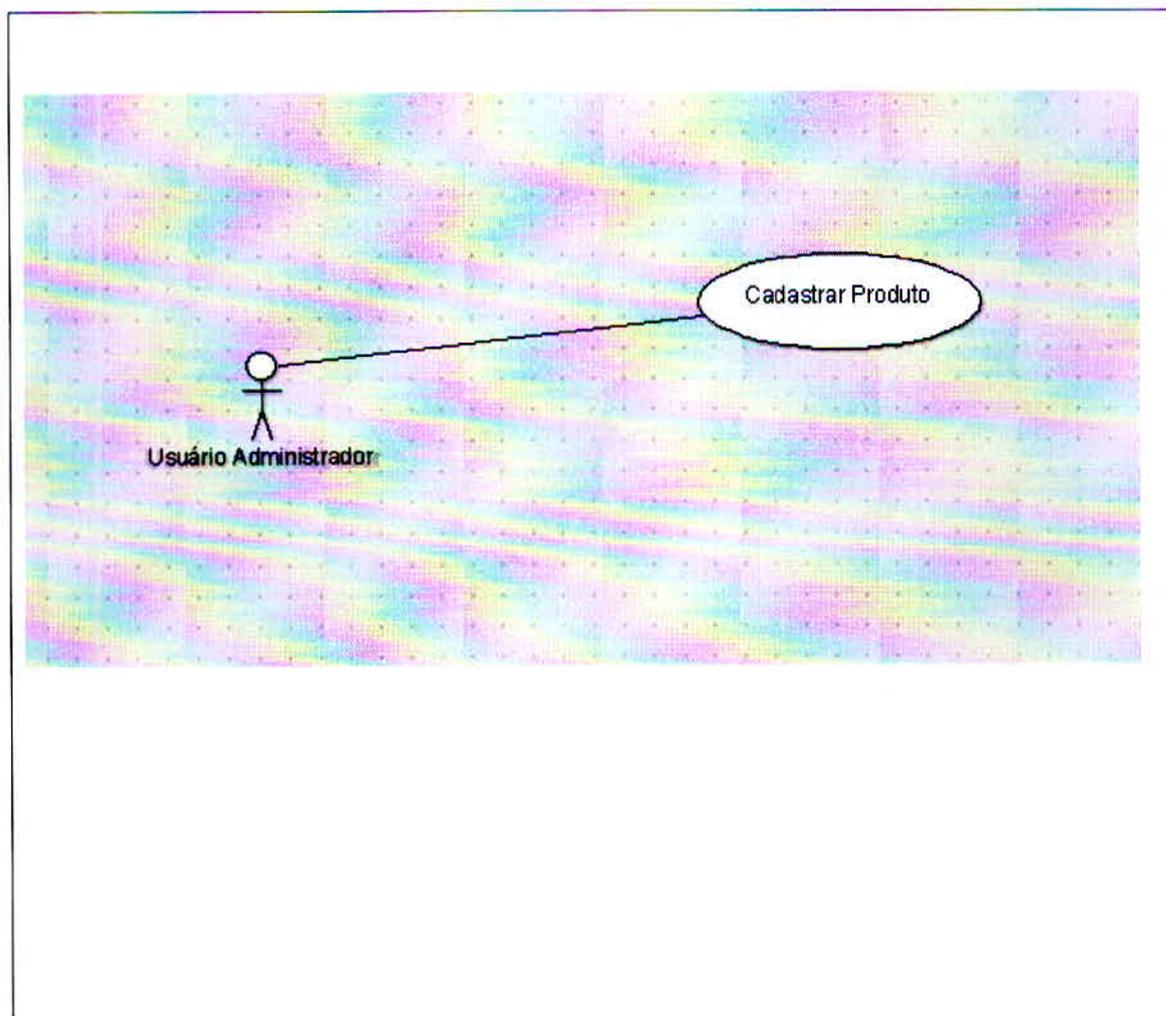
4.5 Especificação do Caso de uso – Cadastrar Produtos

CASO DE USO – CADASTRAR PRODUTOS		
DADOS		
Identificação: Caso de uso cadastrar produtos	Projeto: Aplicativo Compra Café	Versão: Ap.01
Cliente: Cocatrel – Cooperativa dos Cafeicultores da Zona de Três Pontas		
Descrição sucinta: Este caso de uso permite o cadastramento de produtos no sistema.		

Atores:	
<ul style="list-style-type: none"> ○ Usuário administrador do sistema 	
Pré-condições:	Pós-condições:
<ul style="list-style-type: none"> ○ Usuário administrador cadastrado no sistema. ○ Usuário administrador logado no sistema. 	<ul style="list-style-type: none"> ○ Haverá um registro de produto cadastrado.
Casos de uso associados:	
<ul style="list-style-type: none"> ○ Cadastrar usuário administrador 	
FLUXO BÁSICO	
(F1)	Este caso de uso se inicia quando o usuário administrador seleciona a opção de “Cadastrar Produtos”.
(F2)	O sistema disponibiliza os campos para o cadastramento do produto.
(F3)	O usuário administrador preenche os seguintes campos para o cadastramento: <ul style="list-style-type: none"> ○ Código do produto ○ Identificação ○ Valor ○ Preço de venda ○ Quantidade estocada
(F4)	O usuário confirma o cadastro clicando no botão “ <u>Salvar</u> ”
(F5)	O sistema informa que o produto foi cadastrado com sucesso.
(F6)	O usuário administrador continua o processamento clicando no botão <u>OK</u>
(F7)	O caso de uso finaliza com o sistema disponibilizando ao usuário administrador a tela inicial do sistema para finalidades administrativas.

FLUXO DE EXCEÇÕES	
(E1)	<p><i>Todos os campos obrigatórios</i></p> <p>Todos os campos são obrigatórios e ao não preenchimento exibirá uma mensagem de alerta informando a necessidade de preenchimento dos campos com asterisco (*)</p>

DIAGRAMA



4.6 Especificação do Caso de uso – Abastecer o estoque

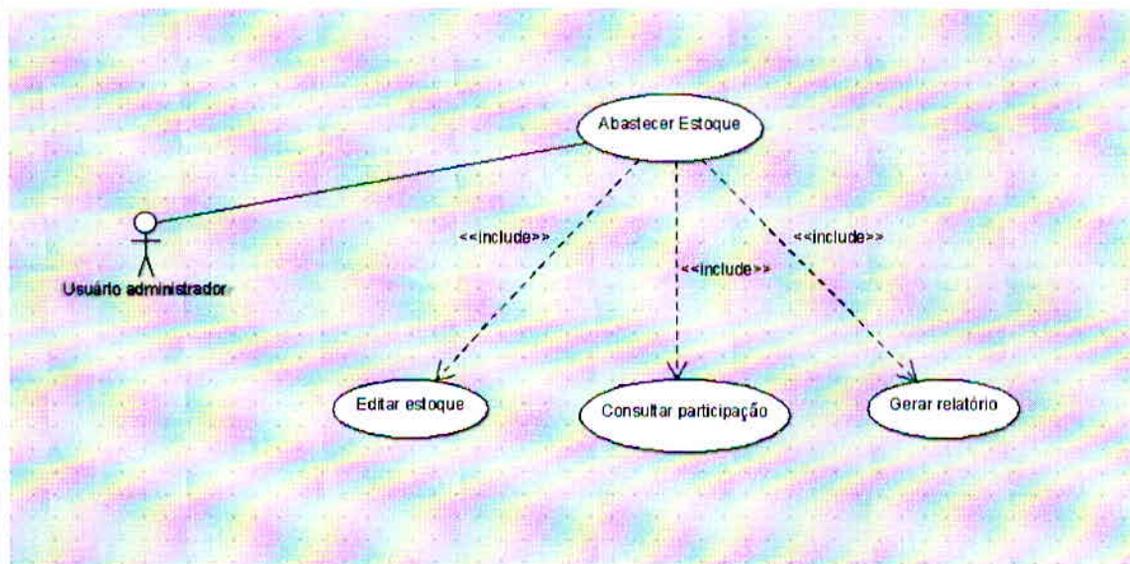
CASO DE USO – ABASTECER O ESTOQUE		
DADOS		
Identificação: Caso de uso Abastecer o estoque	Projeto: Aplicação Web Compra Café	Versão: Ap.01
Cliente: Cocatrel – Cooperativa dos Cafeicultores da Zona de Três Pontas		
Descrição sucinta: Este caso de uso possibilita ao usuário visualizar e editar informações do produto e da quantidade do produto no estoque.		

Atores:	
<ul style="list-style-type: none"> ○ Usuário administrador da aplicação web 	
Pré-condições:	Pós-condições:
<ul style="list-style-type: none"> ○ Usuário administrador cadastrado e logado no sistema. ○ Exista algum produto cadastrado 	<ul style="list-style-type: none"> ○ Haverá produto e estoque alterados.
Casos de uso associados:	
<ul style="list-style-type: none"> ○ Cadastrar Usuário Administrador ○ Cadastrar Produto 	
FLUXO BÁSICO	
(F1)	Este caso de uso se inicia quando o usuário seleciona a opção do botão <u>Estoque</u> .
(F2)	<p>O sistema apresenta uma listagem com todos os produtos cadastrados. Com tuplas compostas pelos seguintes campos:</p> <ul style="list-style-type: none"> ○ Código do produto ○ Identificação ○ Valor ○ Preço de venda ○ Quantidade acrescentada ao estoque (QAE) ○ Estoque atual ○ Quantidade vendida ○ Faturamento com o produto ○ Lucro ○ Lucro Total ○ Opções (A1) (A2) (A3)
(F3)	O usuário pesquisa o produto pelo nome
(F4)	O sistema exibe os dados do produto solicitado
(F5)	O caso de uso finaliza disponibilizando as opções de “Editar”, “Marketing” e “Relatório M”

FLUXOS ALTERNATIVOS

(A1)	<p>Editar:</p> <p>A opção “Editar” possibilita ao usuário editar todos os dados relacionados ao produto que são apresentados na listagem. Sendo que, o “Estoque”, a “Quantidade Vendida”, o “Faturamento com o produto” e o “Lucro”; são calculados mensalmente pelo programa e não permitem acesso de edição ao usuário.</p>
(A2)	<p>Marketing:</p> <p>A opção “Marketing” possibilita a apresentação de um relatório em que se pode verificar a porcentagem de participação do produto no total do faturamento e do lucro da empresa a qualquer instante.</p>
(A3)	<p>Relatório M:</p> <p>A opção “Relatório M” apresenta ao usuário um relatório completo dos produtos comercializados pela empresa com todos os dados alimentados pelo usuário e pelo sistema durante o mês. Sendo que, após gerar o relatório as informações de “Estoque”, “Quantidade Vendida”, “Faturamento com o produto” e o “Lucro” são zerados para essa tabela e armazenados na tabela anual.</p>

DIAGRAMA



4.7 Especificação do Caso de uso – Comprar produtos

CASO DE USO - COMPRAR PRODUTOS		
DADOS		
Identificação: Caso de uso comprar produtos	Projeto: Aplicação Web Compra Café	Versão: Ap.01
Cliente: Cocatrel – Cooperativa dos Cafeicultores da Zona de Três Pontas		
Descrição sucinta: Este caso de uso permite que o cliente compre um produto.		
Atores: <ul style="list-style-type: none"> ○ Usuário cliente 		
Pré-condições: <ul style="list-style-type: none"> ○ Cliente cadastrado no sistema ○ Usuário cliente logado no sistema ○ Produto cadastrado no sistema ○ Produto abastecido no sistema 	Pós-condições: <ul style="list-style-type: none"> ○ Haverá um registro de compra no Sistema 	
Casos de uso associados: <ul style="list-style-type: none"> ○ Cadastrar cliente ○ Cadastrar produto ○ Abastecer estoque 		
FLUXO BÁSICO		
(F1)	Este caso de uso se inicia quando o usuário seleciona a opção “Comprar” na página principal da Aplicação Web.	
(F2)	O usuário cliente informa: <ul style="list-style-type: none"> ○ A identificação do produto ○ O preço ○ Condições de pagamento. 	
(F3)	O sistema informa a data da entrega e confirma o endereço.	

(F4)	O sistema disponibiliza a confirmação do pedido “Concluir a compra” ou a edição do endereço “Editar Endereço” (A1).
(F5)	O sistema apresenta uma mensagem de agradecimento pela confirmação do pedido
(F6)	O processamento continua clicando em OK.
(F7)	O pedido em tela é disponibilizado com a condição de impressão.
(F8)	O caso de uso finaliza com sistema disponibilizando as opções de nova compra ou de sair do sistema.

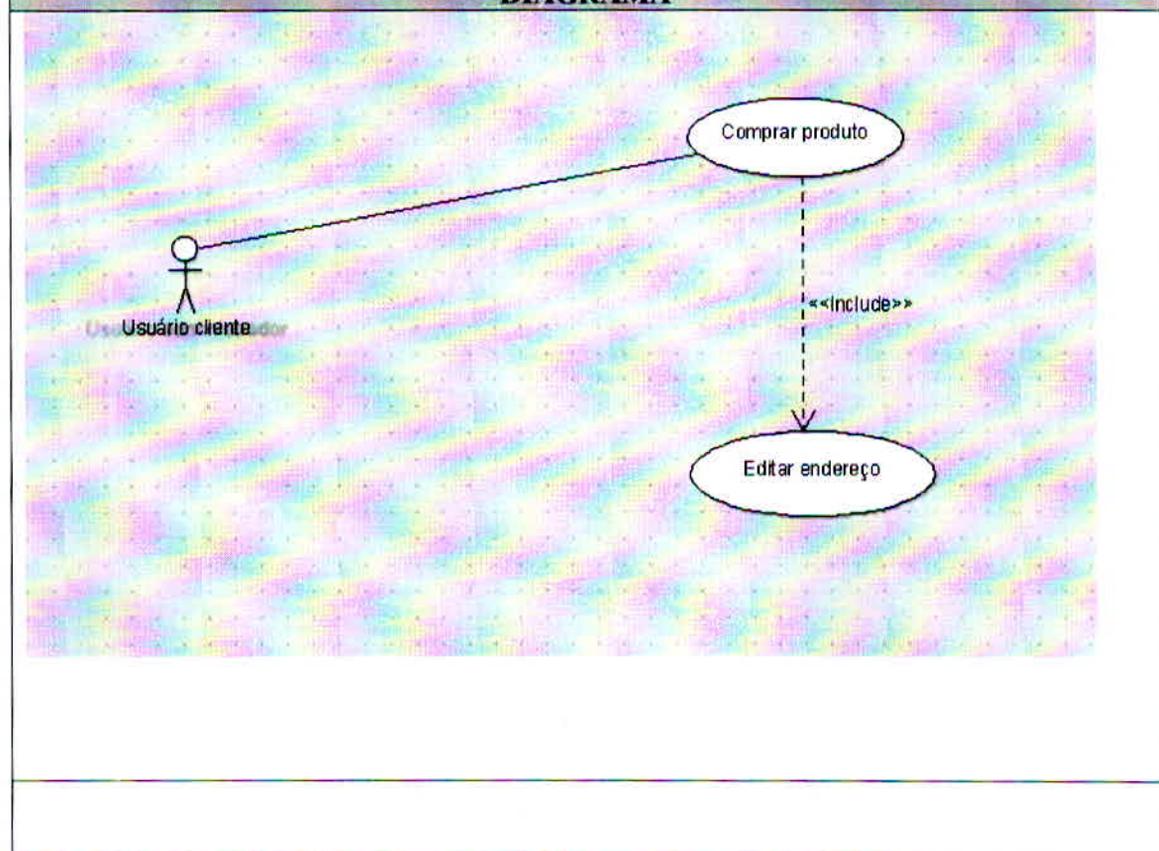
FLUXO DE EXCEÇÕES

(E1)	<i>Todos os campos – Obrigatórios</i> Todos os campos são obrigatórios e ao não preenchimento exibirá uma mensagem de alerta informando a necessidade de preenchimento dos campos com asterisco (*)
------	--

FLUXOS ALTERNATIVOS

(A1)	<i>Editar endereço</i> A opção “Editar endereço” permite a alteração do endereço, caso for necessário.
------	---

DIAGRAMA



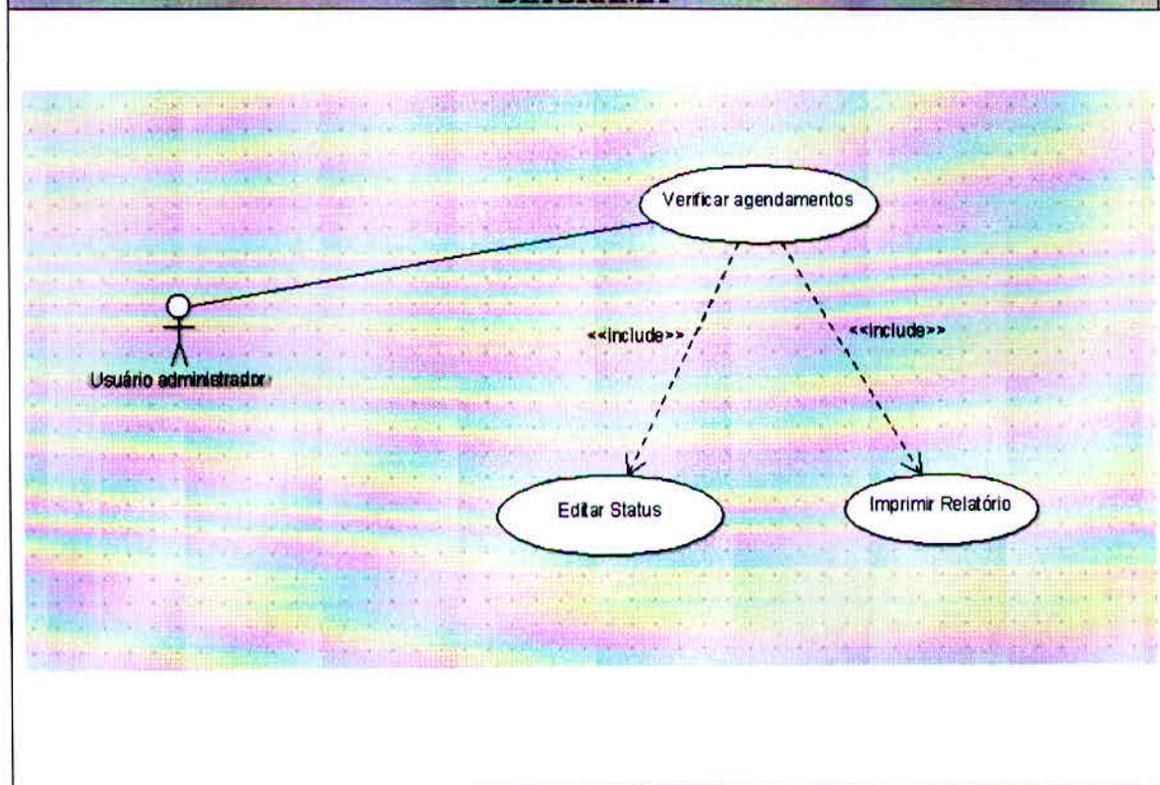
4.8 Especificação do Caso de Uso – Verificar agendamento de entregas

CASO DE USO - VERIFICAR AGENDAMENTO		
DADOS		
Identificação: Caso de uso Verificar agendamento	Projeto: Aplicação web Compra Café	Versão: Ap.01
Cliente: Cocatrel – Cooperativa dos Cafeicultores da Zona de Três Pontas		
Descrição sucinta: Este caso de uso permite a verificação dos agendamentos das entregas aos clientes.		
Atores: <ul style="list-style-type: none"> ○ Usuário administrador do sistema 		
Pré-condições: <ul style="list-style-type: none"> ○ Usuário administrador logado no sistema ○ Existência de pelo menos uma venda solicitada ao sistema. 	Pós-condições: <ul style="list-style-type: none"> ○ Um relatório será gerado 	
Casos de uso associados: <ul style="list-style-type: none"> ○ Cadastrar usuário administrador ○ Cadastrar usuário cliente ○ Comprar produto 		

FLUXO BÁSICO	
(F1)	Este caso de uso se inicia quando o usuário seleciona o botão “Agendamentos”, que se encontra na tela principal do ambiente do administrador.
(F2)	O sistema disponibiliza campos de data inicial e final para determinar o limite da pesquisa.
(F3)	O usuário informa as datas da pesquisa e clica em “OK”.
(F4)	O sistema apresenta um relatório dos agendamentos de entrega e a situação atual (status). E a possibilidade de “Editar Status”(A1) e “Imprimir Relatório”(A2)
(F5)	O caso de uso termina com a escolha do botão “Voltar” e o sistema retorna à tela inicial do ambiente administrativo.

FLUXOS ALTERNATIVOS

(A1)	<i>Editar Status</i> A opção “Editar Status” disponibiliza a possibilidade de alterar o status da entrega entre “Entregue” e “Pendente”.
(A2)	<i>Imprimir Relatório</i> A opção “Imprimir Relatório” possibilita a impressão dos agendamentos.

DIAGRAMA

4.9 Página inicial da Aplicação Web – Compra Café

Esse wireframe da página inicial apresentado abaixo na figura 9, propiciará ao usuário a possibilidade de entrar no sistema. Para que esse evento ocorra o usuário tem a possibilidade de se cadastrar, ou caso já seja cadastrado clicar em “Comprar” para usufruir das promoções exclusivas que se encontram disponíveis aos usuários dessa aplicação.

Caso o cadastramento seja de um usuário do setor administrativo, o mesmo será previamente informado de um login e senha iniciais para que o sistema aceite o cadastramento inicial ou clicar em no botão “ADM” para utilizar do sistema.

Figura 09 – Página inicial da Aplicação Web – Compra Café



4.10 Formulário – Cadastro de clientes

Com o formulário do cadastro de clientes o usuário informará todos os dados necessários para alimentar o banco de dados da aplicação web.

Figura 10 – Formulário para o cadastro de clientes

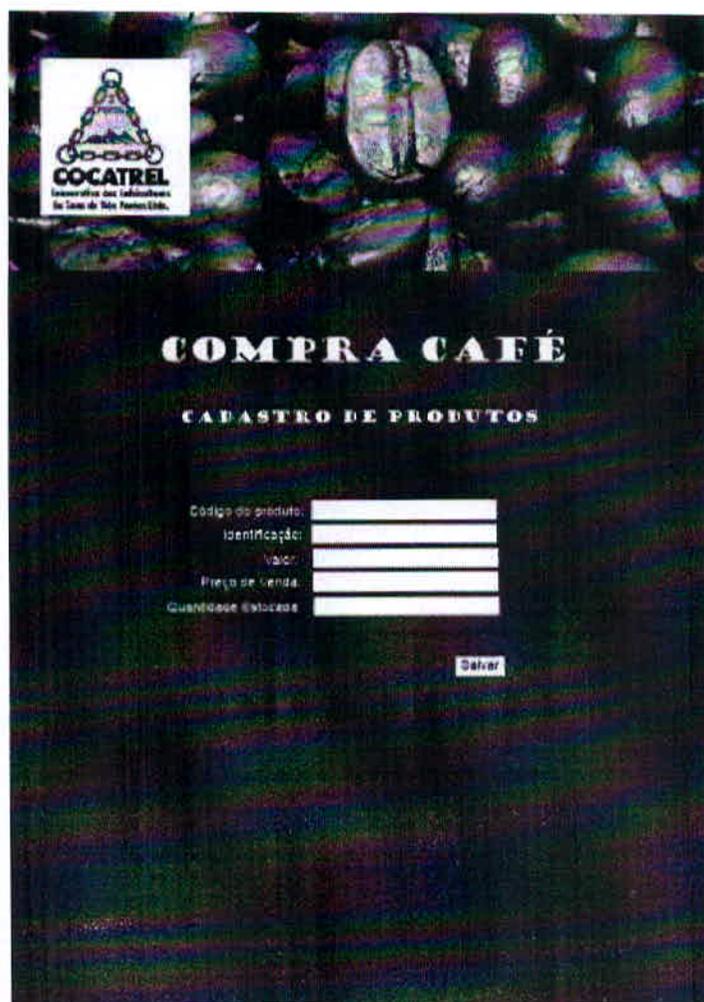
The image shows a web form for client registration. At the top left is the logo for 'COCATREL' (Cooperativa dos Cultivadores do Caramuru Ltda.), featuring a stylized coffee plant. The background of the form is a dark image of coffee beans. The main heading is 'COMPRACAFÉ' in large, bold, white letters, with the subtitle 'CADASTRO DE CLIENTES' below it. The form contains several input fields: 'CNPJ', 'Razão Social', 'Nome Fantasia', 'Estrada', 'Rua', 'Número', 'Complemento', 'Bairro', 'Cidade', 'UF', 'CEP', 'Telefone', 'Ramal', 'Complador', 'CPF do Complador', 'Email', and 'Senha'. There is also a 'Confirma Senha' field. A 'Salvar' button is located at the bottom right of the form.

Fonte: o autor

4.11 Formulário – Cadastro de Produtos

Com esse formulário inicial de cadastro de produtos o usuário administrador cadastrará os produtos que irão fazer parte do mix de produtos a serem negociados pela empresa.

Figura 11 – Formulário para cadastro de produtos.



Fonte: o autor.

4.12 Página inicial do Ambiente de Compras

Na página inicial do ambiente de compras o usuário encontrará os produtos ofertados com as devidas informações adicionais, o preço e o pedido mínimo pré-estabelecido para se iniciar a compra.

Figura 12 – Página inicial do ambiente de compras.

COCATREL
Associação dos Trabalhadores
do Sudoeste de São Paulo

COMPRA CAFÉ
outros produtos

Café COCATREL Gourmet Suave Torrado Extremo Tipo Exposição
Arábica - 100% Arábica Sul de Minas - Embalagem 500 g
Preço: R\$ 30,00 / kg - Para compras acima de 100 quilos

Café COCATREL Gourmet Suave Torrado em Grãos para moedores e máquinas de Café Expresso - 100% Arábica Sul de Minas - Embalagem 1 kg
Preço: R\$ 29,50 / kg - Para compras acima de 100 quilos

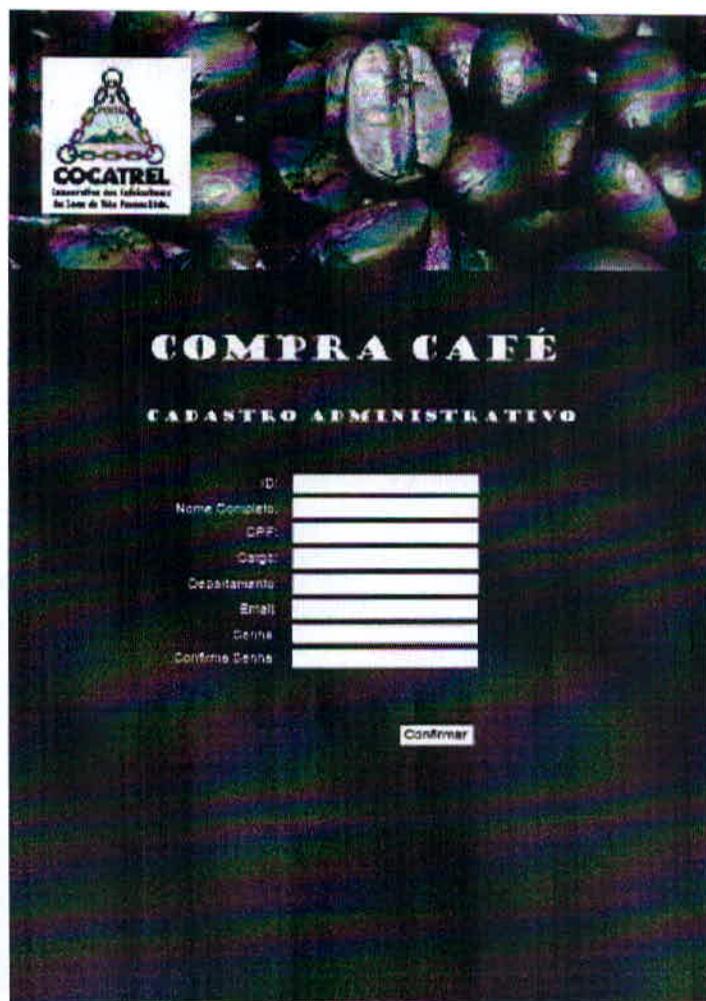
Café COCATREL Gourmet Suave Torrado e Moído
- 100% Arábica Sul de Minas - Embalagem 500g
Preço: R\$ 20,00 / kg - Para compras acima de 100 quilos

Fonte: o autor.

4.13 Formulário do Cadastro Administrativo

Com esse formulário o usuário administrador será capaz de se cadastrar na aplicação web, alimentando o banco de dados do sistema.

Figura 13 – Formulário para o cadastro administrativo.



The image shows a web form titled 'COMPRAS CAFÉ' with the subtitle 'CADASTRO ADMINISTRATIVO'. The form is set against a background of coffee beans. In the top left corner, there is a logo for 'COCATREL' (Associação dos Trabalhadores do Lado de São Francisco) featuring a stylized figure. The form fields are as follows:

ID:	<input type="text"/>
Nome Completo:	<input type="text"/>
CPF:	<input type="text"/>
Cargo:	<input type="text"/>
Departamento:	<input type="text"/>
Email:	<input type="text"/>
Senha:	<input type="password"/>
Confirma Senha:	<input type="password"/>

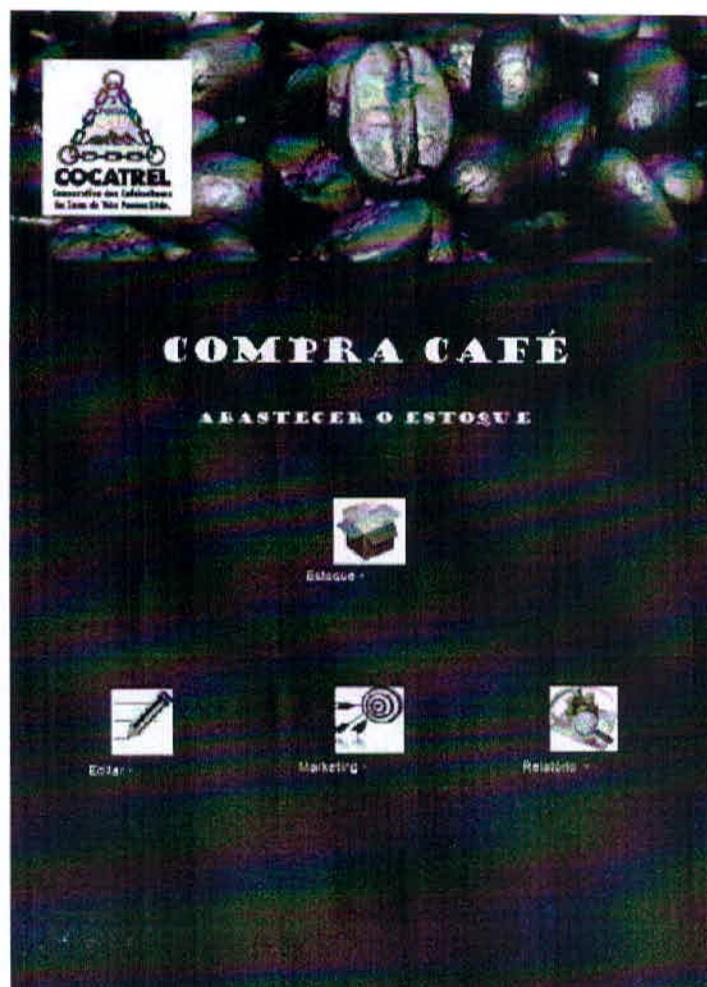
At the bottom center of the form is a 'Confirmar' button.

Fonte: o autor.

4.14 Página Abastecer o Estoque

Na página abastecer o estoque o usuário administrador terá condições de editar e verificar o estoque, como também de verificar a situação de participação de cada produto na lucratividade da empresa e de emitir um relatório de todo o andamento das negociações.

Figura 14 – Página para abastecer o estoque.



Fonte: o autor.

4.15 Apresentação do Relatório do Estoque

Essa tabela abaixo exemplifica a apresentação básica do Relatório do Estoque. Sendo que, a identificação do produto e o acréscimo do estoque podem ser editados, o restante dos registros são corrigidos de forma automatizada conforme a atividade do sistema.

Tabela 01 – Tabela / Relatório do Estoque.

Código do Produto	Identificação	Valor	Preço de Venda	Acréscimo do estoque	Estoque Atual	Quantidade Vendida	Faturamento	Lucro	Lucro Total
110	CAFÉ GOURMET EXTRAFINO	R\$ 20,00	R\$ 30,00	2.500.000	5.250.000	2.368.485	R\$ 71.054.580,00	R\$ 23.684.860,00	R\$ 53.762.262,00
111	CAFÉ GOURMET MOIDO	R\$ 15,00	R\$ 20,00	5.500.000	10.450.000	3.238.378	R\$ 104.767.560,00	R\$ 26.193.890,00	
112	CAFÉ GOURMET EM GRÃOS	R\$ 18,00	R\$ 26,00	500.000	1.750.000	485.689	R\$ 12.627.914,00	R\$ 3.885.512,00	

Fonte: o autor.

4.16 Apresentação do Relatório Marketing

A tabela abaixo apresenta o relatório Marketing em que a apresentação da participação no lucro de cada produto propicia ao administrador a uma melhor análise da representatividade de cada produto na situação rentável empresarial, auxiliando ao administrador na tomada de decisões quanto a estipulação de preço de venda, ou em alguma ação de mercado que possa ser acrescentada a algum produto que não esteja com resultados satisfatórios.

Tabela 02 – Tabela / Relatório Marketing.

Código do Produto	Valor	Preço de Venda	Quantidade Vendida	Faturamento	Lucro	Lucro Total	Participação no Lucro
110	R\$ 20,00	R\$ 30,00	2.368.486	R\$ 71.054.580,00	R\$ 23.684.860,00	R\$ 53.762.262,00	44,05%
111	R\$ 15,00	R\$ 20,00	5.238.378	R\$ 104.767.560,00	R\$ 26.191.890,00		48,72%
112	R\$ 18,00	R\$ 26,00	485.689	R\$ 12.627.914,00	R\$ 3.885.512,00		7,23%

Fonte: o autor.

4.17 Apresentação do Relatório do Planejamento das Entregas

A tabela do Relatório do Planejamento das Entregas apresenta ao administrador uma visão macro da situação das entregas, possibilitando ao mesmo visualizar em qual data um identificado cliente receberá um determinado produto que foi comprado em uma data especificada. Como também apresenta a quantidade comprada o valor total comprado de cada produto e o valor total da entrega.

Tabela 03 – Tabela / Relatório Planejamento das entregas.

COD CLIENTE	COD_PRODUTO	QUANTIDADE COMPRADA	VALOR TOTAL POR PRODUTO	VALOR TOTAL DA ENTREGA	DATA DA COMPRA	DATA DA ENTREGA
1	110	100	R\$ 3.000,00	R\$ 11.600,00	06/10/2014	09/10/2014
1	111	300	R\$ 6.000,00			
1	112	100	R\$ 2.600,00			
5	111	100	R\$ 2.000,00	R\$ 2.000,00	07/10/2014	10/10/2014
2	112	100	R\$ 2.600,00	R\$ 2.600,00	08/10/2014	11/10/2014

Fonte: o autor.

5 CONCLUSÃO

Com esse trabalho foi possível desenvolver um protótipo apresentável de uma aplicação web para fins comerciais. Que abordou a possibilidade de obtenção de resultados que possam vir a favorecer usuários de ambos lados de uma negociação disponibilizada somente a uma clientela selecionada que faz parte da carteira de clientes da Cooperativa de produtores de café da zona de Três Pontas.

Embora que seja um protótipo, a finalidade de tornar possível a interação entre clientes selecionados e um fornecedor, tendo por objetivo principal a agilidade na obtenção de resultados comerciais é um projeto, que certa forma, mostra uma tendência comum e fortemente utilizada, mas que para se conseguir bons resultados para uma clientela previamente selecionada, possa ainda ter que se desenvolver melhores ferramentas interativas para identificação prévia e acesso limitado aos clientes que possam vir a ser beneficiados pela utilização da aplicação.

Percebe-se ainda que, o desempenho trabalho trouxe importante contribuição acadêmica para o autor, para que o mesmo pudesse solidificar melhor os conceitos e conhecimentos adquiridos no curso de Bacharel em Sistemas de Informação.

REFERÊNCIAS

- AGUILAR, Luis Joyanes. **Fundamentos de Programação – Algoritmos, estruturas de dados e objetos**: 3ª edição – São Paulo: McGraw-Hill Interamericana do Brasil Ltda, 2008.
- ALVES, Andrea Cristina Oliveira. **Guia de Estudo – Banco de Dados**: Varginha/MG: GEAD-UNIS/MG, 2012.
- BRUNO, Odemir M., ESTROZI, Leandro F., NETO, João E.S. Batista. **Programando para a internet com PHP**: Rio de Janeiro – Brasport, 2010.
- CAMPOS, André. **Sistema de Segurança da Informação – Controlando os Riscos**: 2ª Edição – Florianópolis: Editora Visual Books, 2007.
- DATE, C. J. **Introdução a Sistemas de Banco de Dados**: 8ª edição – São Paulo :Elsevier Editora Ltda.
- FONSECA, Leticia Rodrigues da. **Guia de Estudo – Comércio Eletrônico**: Varginha/MG: GEaD – UNIS/MG, 2010.
- FOWLER, Martin. **UML Essencial – Um breve guia para a linguagem-padrão de modelagem de objetos**: 3ª edição – Porto Alegre: Bookman, 2005.
- GOMES, Ana Laura. **Xhtml/CSS – Criação de Páginas Web**: São Paulo: Editora Senac, 2010.
- HEUSER, Carlos Alberto. **Projeto de Banco de Dados**: 4ª edição – Porto Alegre: UFRGS, 1998.
- JOBSTRAIBIZER, Flávia. **Criação de banco de dados com MySQL**: São Paulo:Digerati Books, 2010.
- LAKATOS, Eva Maria; MARCONI, Marina de Andrade . **Fundamentos de Metodologia Científica**:5ª edição – São Paulo: Atlas, 2003.
- LYRA, Mauricio Rocha. **Segurança e Auditoria em Sistemas de Informação**: Rio de Janeiro: Editora Ciência Moderna Ltda,2008.
- LOWE, David B., SELLERS, Brian Henderson. **Characteristics of Web Development Processes**: present at: SSGRR – 2001: Infrastruture for E-Business, E-Education, and E-Science: L’Aquila, Italy, 2001.
- PRADO, Alan Souza. SILVA, Rodrigo Comes da – **Guia de Estudo – Desenvolvimento de Aplicações para Internet**: Varginha/MG: GeaD – UNIS, 2009.
- SILVA, Rodrigo Gomes da. **Introdução à Engenharia de Requisitos – Conceitos, fundamentos e ferramentas – Referências do Babok 2.0 e RUP 7.1.1 – Abordagem teórica e prática**: Varginha/MG: UNIS, 2014.

SINTES, Tony. **Aprenda Programação Orientada a Objetos em 21 dias**: São Paulo Pearson Education do Brasil – Pearson Education do Brasil, 2002.

SOUZA, Rafael Rodrigues de. **Guia de Estudo – Engenharia de Software**: Varginha – MG: GEad – UNIS, 2008.

SOMMERVILLE, Ian. **Engenharia de software**: São Paulo: Addison Wesley, 2003.

REIS, Paulo Pinto. **Artigo – O que são sistemas WEB (web-based)?**:Centro de Competência TIC – Entre Mar e Serra: Colocado na internet em 16/01/2008, no endereço:

<<http://www.ccems.pt/Artigos/tabid/215/ctl/Details/mid/994/ItemID/6/language/pt-PT/Default.aspx>>Acesso em: 30 maio 2014.

PIURCOSKY, Fabrício Pelloso. **Guia de Estudo – Segurança e Auditoria de Sistemas**: Varginha: GEaD – UNIS, 2012.

PLAIS, Antonio. **Artigo – Definindo Regras de Negócio: O que são elas, realmente?**:CENTUS – Consultoria e Negócios: postado na internet em 13/12/2012 e atualizado em 27/07/2013, no endereço URL:

<<http://www.centus.com.br/base-de-conhecimento/artigos/definindoregrasdenegocio>>

Acesso em 02 jun. de 2014.