



**Fundação Educacional do Município de Assis  
Instituto Municipal de Ensino Superior de Assis  
Campus "José Santilli Sobrinho"**

**CARLOS EDUARDO DI LORETO DI RAIMO**

**SISTEMA DE CONTROLE DE ESTOQUE**

**REVENDA AVON**

**ASSIS - SP**

**2013**

**CARLOS EDUARDO DI LORETO DI RAIMO**

**SISTEMA DE CONTROLE DE ESTOQUE**

**REVENDA AVON**

Monografia apresentada no Curso de Graduação da Instituição Municipal de Ensino Superior de Assis – IMESA e Fundação Educacional do Município de Assis – FEMA, com objetivo de concluir o curso de Análise e Desenvolvimento de sistema.

**Orientador:** Prof. Dr. Alex Sandro Romeo de Souza Poletto

**Área da Concentração:** Informática

**ASSIS – SP**

**2013**

## **FICHA CATALOGRÁFICA**

DI RAIMO, Carlos Eduardo Di Loreto

Sistema de Controle de Estoque – Revenda Avon, Carlos Eduardo Di Loreto Di Raimo, Fundação Educacional do Município de Assis, Assis, 2013.

Número de páginas 41p.

Orientador: Prof. Dr. Alex Sandro Romeo de Souza Poletto

Trabalho de Conclusão de Curso – Instituto Municipal de Ensino Superior de Assis (IMESA).

1. Sistema de Controle de Estoque 2. Sistema de Controle de Revenda.

CDD: 001.61

Biblioteca da FEMA

**SISTEMA DE CONTROLE DE ESTOQUE**  
**REVENDA AVON**

**CARLOS EDUARDO DI LORETO DI RAIMO**

Trabalho de Conclusão de Curso apresentado ao Instituto Municipal de Ensino Superior de Assis como requisito do Curso de Graduação analisado pela seguinte comissão examinadora.

Orientador: Prof. Dr. Alex Sandro Romeo De Souza Poletto

Analisador: Prof. Esp. Guilherme de Cleve Farto

**Assis**  
**2013**

## DEDICATÓRIA

Dedico este trabalho a Deus, a minha namorada que me apoiou nas horas mais difíceis, a minha família por ter me incentivado no decorrer deste curso e pelo apoio de todos que me deram força para não desistir de meu propósito.

## **AGRADECIMENTOS**

Agradeço a Deus, pois sem ele eu nada conquistaria.

A minha namorada pela força e pelo apoio nesses três anos de curso.

Ao meu Orientador e Mestre Alex Poletto, pela atenção e dedicação que teve ao decorrer do trabalho.

A meus familiares, que sempre estiveram ao meu lado, me brindando de todo apoio necessário para prosseguir e conquistar os meus sonhos.

E aos professores do curso de Análise e Desenvolvimento de Sistema que me ensinarão muito durante o decorrer do curso.

A estes o meus mais sinceros Agradecimentos.

*“Não sei se a vida é curta ou longa para nós, mas sei que nada do que vivemos tem sentido, se não tocarmos o coração das pessoas.*

*Muitas vezes basta ser: colo que acolhe, braço que envolve, palavra que conforta, silêncio que respeita, alegria que contagia, lágrima que corre, olhar que acaricia, desejo que sacia, amor que promove.*

*E isso não é coisa de outro mundo, é o que dá sentido à vida. É o que faz com que ela não seja nem curta, nem longa demais, mas que seja intensa, verdadeira, pura enquanto durar. Feliz aquele que transfere o que sabe e aprende o que ensina.”*

*Cora Coralina*

## **RESUMO**

Os revendedores Avon necessitam de um sistema de controle de estoque, pois não podem contar com um programa para auxiliá-los. Com isto surgiu a ideia de desenvolver um sistema que os ajudaria a controlar todo o estoque existente. O sistema possibilitará ao usuário um controle minucioso das vendas efetuadas, diminuindo assim as perdas e auxiliando no controle de seus lucros.

1. Palavras-chave: Sistema de Controle de Estoque, Revenda de sistema de controle.

## **ABSTRACT**

Dealers Avon require a system of inventory control, because it has a program that assists. Ai came up with the idea of developing a program in the system that would help them control the entire inventory. The system will allow the user to control their sales thus reducing losses and aiding in control of your profits.

1. Keywords: Inventory Control System, Control System Reseller.

## LISTA DE ILUSTRAÇÕES

Figura 1: Diagrama de caso de Uso Geral.....	23
Figura 2: Caso de Uso: Usecase_01 Manter Cliente.....	23
Figura 3: Caso de Uso: Usecase_02 Manter Produto.....	24
Figura 4: Caso de Uso: Usecase_03 Gerar Relatório.....	25
Figura 5: Caso de Uso: Usecase_04 Manter Vendas.....	26
Figura 6: Diagrama de WBS.....	28
Figura 7: Diagrama de Classe.....	29
Figura 8: Diagrama Entidade Relacionada.....	29
Figura 9: Camadas.....	31
Figura 10: Dal.....	32
Figura 11: ExecutarComando.....	32
Figura 12: Consultar.....	33
Figura 13: Visual.....	33
Figura 14: Menu.....	34
Figura 15: Cadastros.....	35
Figura 16: Cadastro Produtos.....	35
Figura 17: Efetuar Venda.....	36

## LISTA DE TABELA

Tabela 1. Documentação de Caso de Uso de Manter Cliente.....	23
Tabela 2. Documentação de Caso de Uso de Manter Produto.....	24
Tabela 3. Documentação de Caso de Uso de Relatório.....	25
Tabela 4. Descrição de Caso de Uso de Venda.....	26
Tabela 5. Orçamento dos Produtos.....	39

## SUMÁRIO

<b>1. INTRODUÇÃO.....</b>	<b>15</b>
1.1 JUSTIFICATIVA.....	15
1.2 OBJETIVOS.....	15
1.3 PÚBLICO ALVO.....	16
<b>2. METODOLOGIA DE DESENVOLVIMENTO.....</b>	<b>17</b>
2.1 METODOLOGIA DE ANÁLISE.....	17
2.2 ASTAH PROSSIONAL.....	18
2.3 BANCO DE DADOS SQL SERVER 2008.....	19
2.4 LINGUAGEM C#.....	19
<b>3. DESCRIÇÃO GERAL DO SISTEMA.....</b>	<b>21</b>
3.1 ANÁLISE DE REQUISITOS.....	21
3.2 LISTA DE EVENTOS.....	21
3.3 DIAGRAMA DE CASO DE USO GERAL.....	22
<b>3.3.1 Caso de Uso Manter Cliente.....</b>	<b>23</b>
<b>3.3.2 Caso de Uso Manter Produto.....</b>	<b>24</b>
<b>3.3.3 Caso de Uso de Relatório.....</b>	<b>25</b>
<b>3.3.4 Caso de Uso de Venda.....</b>	<b>26</b>
3.4 DIAGRAMA DE WBS(WORK BREAKDOWN STRUCTURE).....	27
3.5 DIAGRAMA DE CLASSE.....	28
3.6 DIAGRAMA ENTIDADE RELACIONADA.....	29
<b>4. IMPLEMENTAÇÃO.....</b>	<b>31</b>
4.1 ORGANIZAÇÕES DO PROJETO.....	31
4.2 INTERFACE DO SISTEMA.....	34
<b>5. ORÇAMENTO.....</b>	<b>37</b>
5.1 Estimativa de Custo.....	37
5.2 Custo de Material.....	37

5.3 Orçamentos do Projeto.....	39
<b>6. CONCLUSÃO.....</b>	<b>40</b>
6.1 Trabalhos Futuros.....	40
<b>REFERÊNCIAS.....</b>	<b>41</b>

# 1. INTRODUÇÃO

O sistema de controle de estoque de revenda tem como objetivo facilitar o controle dos produtos para revendedores Avon.

Com o desenvolvimento deste sistema, os revendedores Avon terão mais controle da movimentação dos produtos, dos quais possuem no estoque, bem como dos relatórios de vendas mensais das vendas parceladas do montante total das dividas de cada cliente, dentre outras rotinas.

Atualmente não existe nenhum sistema que seja específico para controlar as vendas realizadas.

O sistema é voltado para revendedores que sempre sentiram dificuldades de controlar o estoque dos produtos comercializados.

## 1.1 JUSTIFICATIVA

A necessidade de desenvolver este *software* vem junto com a dificuldade que os revendedores têm passado ao controlar o seu empreendimento, diminuindo os erros e as percas.

Assim podendo servir melhor seus clientes, com maior dedicação e eficiência.

## 1.2 OBJETIVOS

O trabalho tem como objetivo levantar as necessidades do dia a dia de um revendedor Avon, desde os levantamentos de custos, até o desenvolvimento do *software* que irá gerenciar todo o movimento de produtos, bem como emitira um relatório do total vendido por mês e o total a receber.

O *software* gerenciará toda circulação, registrando por completo o processo de entrada e saída de estoque, cadastro de produtos, o cadastro do cliente, vendas a prazo, vendas parceladas, vencimentos, dentre outros eventos.

### 1.3 PÚBLICOS ALVO

O desenvolvido do referido programa atingirá diretamente aos revendedores e seus clientes, facilitando a busca pelo produto escolhido, bem como a busca da dívida de determinado cliente.

Os comerciantes também serão beneficiados com este software, auxiliando no controle do estoque.

## 2. MÉTODO DE DESENVOLVIMENTO

A análise será feita em UML, uma linguagem visual utilizada para modelar *software baseado no paradigma orientado a objeto, que será o conceito do desenvolvimento deste sistema.*

O termo orientado a objeto significa organizar o mundo real como uma coleção de objetos que incorporam a estrutura de dados a um conjunto de operação que manipulam estes dados.

Para o desenvolvimento do sistema serão utilizados os seguintes recursos.

- (Um) *Notebook*;
- *Windows7 Ultimate*;
- (Uma) impressora;
- *Astah professional*;
- Banco de dados *SQL Server 2008*;
- *Visual Studio 2010 (C#)*;
- *DBDESIGNER*;

### 2.1 MÉTODOLOGIAS DE ANÁLISE

Para análise e o desenvolvimento do sistema como na implementação será utilizado à orientação a objetos.

Segundo Eric Gunnerson a orientação a objeto significa organizar o mundo real como uma coleção de objetos que incorporam as estruturas de dados em um conjunto de operações que manipulam estes dados.

A programação Orientada a Objeto e a técnica mais amplamente empregada para desenvolvimento softwares robusto e reutilizável. Também utilizará a linguagem

UML que auxilia na descoberta das características do sistema e como funcionará o próprio. Gilleanes. T. A. Guedes UML 2 ;Uma abordagem pratica.

Além dessas linguagens será utilizada a SQL para controle de banco de dados como definição de dados, consulta e atualização. Segundo Wayne Freeze a SQL tem facilidade para definir as visões do banco de dados, as segurança e autorização, as restrições de integridade, bem como especificar o controle de transação.

As linguagens escolhidas são as mais adequadas para o desenvolvimento do sistema, facilitando a organização dos dados e dos códigos desenvolvidos.

## 2.2 ASTAH PROFISSIONAL

O *Astah Professional* é uma ferramenta que auxilia na criação dos diagramas de fácil manuseio, simples entendimento e com bons detalhes. Intuitiva, envolvente e eminentemente legível. Do qual seu sistema suporta a linguagem UML.

Segundo entendimentos de Gilleanes T.A. Guedes a UML é:

Uma linguagem visual utilizada para modelar software baseados no paradigma Orientado a Objeto. É uma linguagem de modelagem de proposito geral que pode ser aplicada a todos os domínios de aplicação. Esta linguagem tornou nos últimos anos a linguagem padrão adotada internacionalmente pela engenharia de software. (Guedes, Gilleanes T.A, UML2 Uma Abordagem Prática, 2ª Edição, Ed. Novatec, 2011, P.19).

A linguagem UML e uma linguagem que disponibiliza em gráficos uma ideia de como o sistema ira funcionar e qual as suas funcionalidades.

## 2.3 BANCOS DE DADOS SQL SERVER 2008

O Banco de Dados será a Microsoft *SQL Server 2008*, uma linguagem que possibilita o desenvolvedor ter um controle de armazenamento e manipulação de dados.

A linguagem SQL pode ser considerada um dos principais motivos para o sucesso dos bancos de dados relacionados comerciais. Como ela se tornou um padrão para este tipo de banco de dados, os usuários ficarão menos preocupados com a migração de suas aplicações de outros tipos de sistema de rede e hierarquia. A linguagem SQL é uma linguagem de banco de dados abrangente: tem instrução para definição de dados consulta e atualizações. A SQL também tem facilidade para definir visões sobre o banco de dados, para definir restrições de integridade de banco de dados e também para especificar controle de transação. Ela também possui regras para embutir instruções SQL em uma linguagem de programação de uso geral como Java, Cobol, C/C++. (ELMASRI, Ramez e Navathe, Shamkant B. Sistema de Banco de Dados, 6ª Edição, Ed. Pearson, 2011 P.58).

Com base no Escritor Ramez Elmasri a linguagem SQL é de fácil compreensão, tendo como função dar a definição sobre o conteúdo do banco de dados, também é utilizado para fazer ligamento de tabelas. Sendo assim compatível com vários tipos de linguagem de programação.

## 2.4 C sharp (C#).

A ferramenta escolhida para o desenvolvimento foi o *Visual Studio 2010*, da qual utiliza a Linguagem C#, contando com um framework que ajuda muito na sua implementação. Também aplica se alguns métodos, como, *Datatable*, *DataSet*, *DataSource*, com isso há uma facilidade ao consultar as tabelas, as listas, as classes, entre outros. Segundo Anderson Silva (2002) a ferramenta contará com um sistema de Debug que facilita no entendimento de como funciona o aplicativo e para encontrar determinados erros.

“O C# (*C Sharp*) é uma linguagem de programação de orientações a objeto, criada pela Microsoft, da qual faz parte de sua plataforma .Net. A companhia baseou C# em C++ e Java.” (MACKEY, 2011).

Segundo Anderson Silva.

“C# é uma linguagem orientado a objeto. Em alguns aspectos, ela pode ser vista como uma extinção de outras linguagens. A linguagem C# é comparada com o Java e C++. Ela suporta o conceito de classes e a natureza orientado a objeto das classes incluindo encapsulamento, herança e polimorfismo. O framework suporta aplicações de consoles aplicações de interfaces gráficas de usuário (Windows forms). (Anderson Silva, 2002, p32).

Contudo a programação C# é uma linguagem orientada a objeto que também define seus dados e o comportamento de cada classe, disponibilizando também um modo de programação que possibilita o programador dividir – lá em partes, facilitando a modificação do software e de implementação.

A linguagem *C Sharp* facilita na manipulação dos objetos e suas classes, e também auxilia na organização dos dados podendo dividi-los em camadas.

### 3. DESCRIÇÃO GERAL DO SISTEMA

O Sistema de controle, Estoque Avon, terá como objetivo controlar a entrada e a saída dos produtos, gerando relatórios semanais e mensais, cadastro de usuário, editar usuário, excluir usuário, cadastro de produto, editar produto, excluir produto.

#### 3.1 ANÁLISES DE REQUISITOS

Para o Desenvolvimento do sistema foi feita uma entrevista com a revendedora Darlúcia Pinheiro Rozan.

As necessidades são:

- Controlar o estoque, a entrada e a saída de produtos.
- Um sistema, do qual o usuário efetua o cadastro dos clientes e dos produtos, o cadastro deverá conter o nome do novo cliente, seu endereço, o número telefone ou do celular. No cadastro de produtos deverá conter os dados: nome do produto, quantidade do produto, validade do produto, e tipo do produto.
- A revendedora Darlúcia Pinheiro Rozan, disse sobre a necessidade de um relatório de venda semanal e mensal.
- Disponibilizar também uma calculadora.

#### 3.2 LISTAS DE EVENTOS

A lista de eventos tem como objetivo mostrar um jeito simplificado, das atividades que o usuário terá em mãos.

O usuário efetua o login com sua senha.

1. O usuário manterá o cliente
2. O usuário manterá o produto.
3. O usuário manterá a venda.
4. Gerar relatório de vendas semanal.
5. Gerar relatório de vendas mensais.
6. Geras relatório de pedidos.

### 3.3 DIAGRAMAS CASO DE USO GERAL.

Para a especificação dos eventos será utilizado o diagrama de caso de uso Caso de Uso geral (figura 1). Onde ira mostrar de um jeito simples como funcionará o sistema.

Segundo G. Guedes (2011).

O Diagrama de Caso de Uso procura por meio de uma linguagem simples, possibilitar a compreensão do comportamento externo do sistema (em termos de funcionalidades oferecidas por ele). Esse Diagrama Costuma ser utilizado, no inicio da modelagem do sistema, principalmente nas etapas de levantamento e analise de requisitos. Embora venha a ser consultado e modificado durante todo o processo de engenharia e sirva de base para a modelagem de outros diagramas. (G. Guedes, 2011, 28).

Assim podendo facilitar a compreensão do programador e até mesmo do cliente, sobre as atividades que vai a ver no sistema.

A Figura 1 Caso de uso Geral, demonstrara de um modo simples as funcionalidades que o sistema vai oferecer aos usuários.

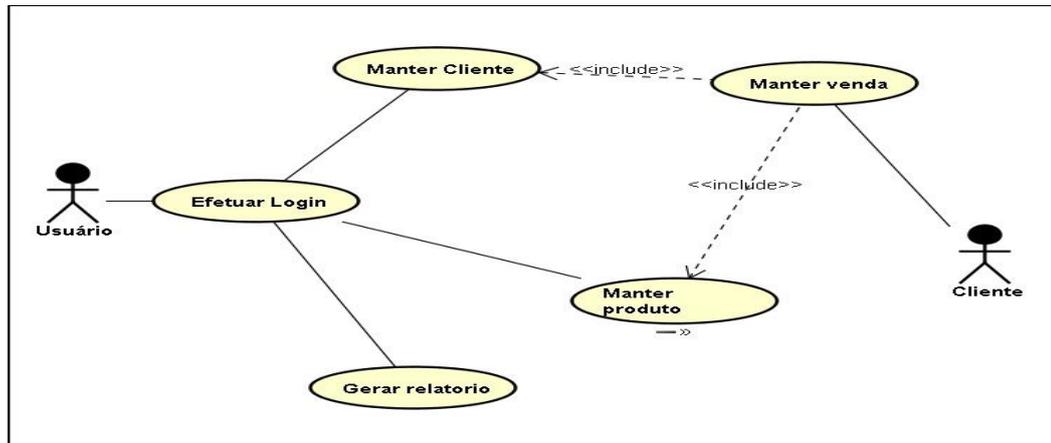


Figura 1 – Ilustra o Diagrama de Caso de Uso Geral.

### 3.3.1 Caso de Uso Manter Cliente

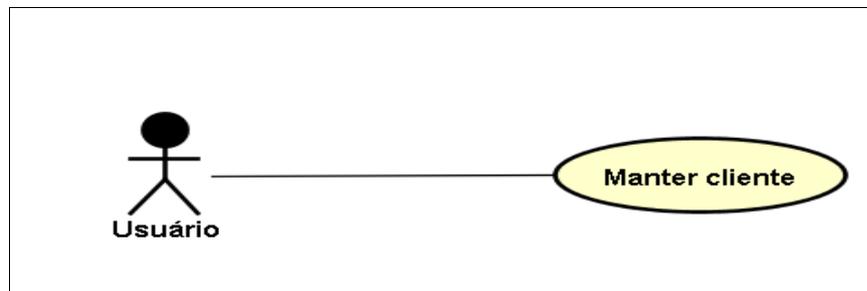


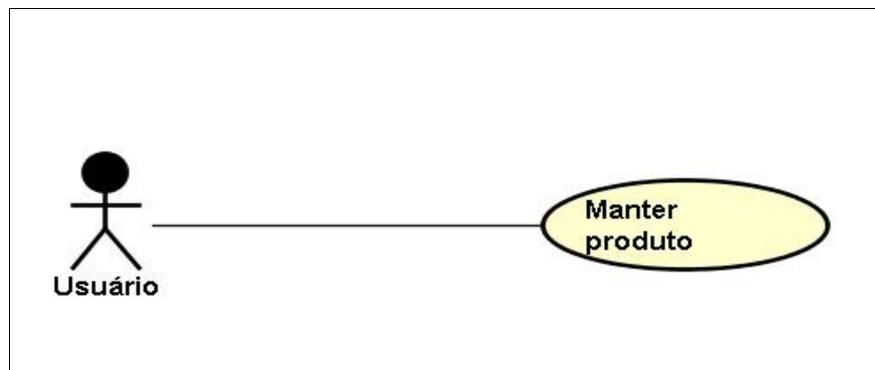
Figura 2 – Ilustra Caso de Uso: UseCase\_01 Manter Cliente.

Nome Do Caso de Uso	Manter Cliente
Caso de uso geral	
Ator principal	Usuário
Ator secundário	
Resumo	Este caso de uso representa o cadastro e a manutenção do cliente, como incluir, alterar, e consultar cliente e a exclusão, o cliente terá que ser excluído definitivamente não poderá ser só desativado.
Pré-condição	O Usuário terá que estar logado no

	sistema.
Pós-condição	
<b>Fluxo principal</b>	
<b>Ação do ator</b>	<b>Ação do cliente</b>
1. Informar os dados do cliente	
	2. Consultar o cliente pelo ID ou CPF.
	3. Se houver o cliente com o CPF informado, mostrar os dados do mesmo.
4. Se necessário inserir ou alterar os dados do cliente.	
	5. Se necessário gravar as alterações.
Restrições/Validações	1. Os campos nome datam de Nascimento, endereço.

**Tabela 1: Documentação Do Caso De Uso Manter Cliente.**

### 3.3.2 Caso de Uso Manter Produto



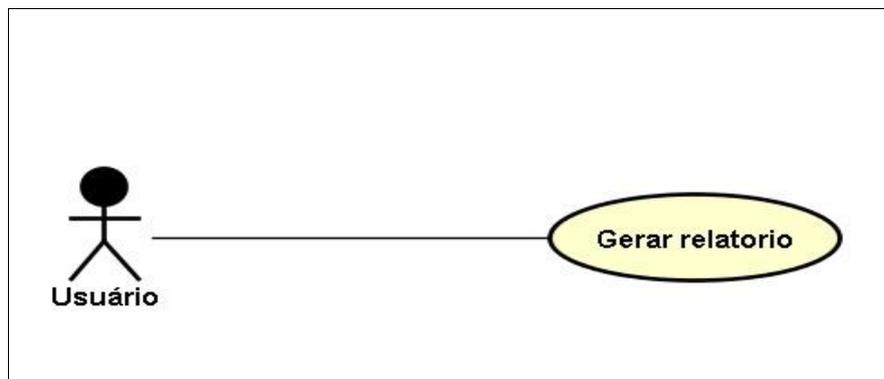
**Figura 3 – Ilustra o Caso de Uso: UseCase-02 Manter Produto.**

<b>Nome caso de uso</b>	<b>Manter Produto</b>
Caso de Uso Geral	
Ator Principal	Usuário
Ator Secundário	
Resumo	Este Caso de Uso representa o cadastro de Produtos, ou seja, permite a inclusão de Produtos, alterar Produto e exclui Produto, mas na exclusão o produto realmente será excluído e não só se tornará inativo.
Pré-condição	O usuário deverá estar logado no

	sistema.
Pós-condição	
<b>Fluxo principal</b>	
1. Informar os dados do Produto	
	2. Consultar o Produto pelo nome
	3. Se houver o produto cadastrado mostrar os detalhes.
4. Se necessário incluir Produto ou alterá-lo.	
	5. Se necessário salvar as alterações.
Restrição/Validação	1. Os Campos de Nome Produto data de vencimento, e quantidade e preço será obrigatório.

**Tabela 2: Documentação Caso de Uso Manter Produto**

### 3.3.3 Caso de Uso de Relatórios



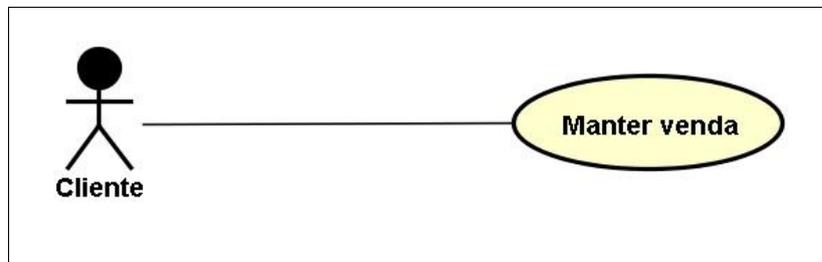
**Figura 4 – Ilustra o Caso de Uso: UseCase\_03 Gerar Relatório.**

<b>Nome do caso de uso</b>	<b>Gerar Relatórios</b>
Caso de Uso Geral	
Ator Principal	Usuário
Ator secundário	
Resumo	Este caso de uso representa a geração de relatórios em modo geral, relatório de venda menção, relatórios de vendas semanais e relatório de produtos em falta.
Pré-condição	O usuário terá que estar logado no sistema.

Pós-condição	
<b>Fluxo principal</b>	
1. Informar a data de início e fim que deseja que o relatório seja gerado.	

**Tabela 3 - Documentação Caso de Uso Relatórios.**

### 3.3.4 Caso De Uso Venda



**Figura 5 – Ilustra o Caso de Uso: UseCase\_04 Manter Vendas.**

<b>Nome do caso de uso</b>	<b>Manter Vendas</b>
Caso de uso geral	
Ator Principal	Cliente
Ator Secundário	Usuário
Resumo	O cliente escolherá os produtos que deseja comprar. Depois que sua lista estiver pronta terá duas opções de pagamento, sendo uma opção efetuar o pagamento à prazo ou à vista
Pré-condição	O usuário deverá estar logado para efetuar a venda e o cliente deverá estar cadastrado no sistema senão estiver será interrompido a venda para que seja efetuado um novo cadastro. Após retornará a venda.
Pós-condições	
<b>Fluxo principal</b>	
Ação do ator	Ação do sistema
1. O cliente escolhe o produto	
	2. O sistema verifica se a o produto

	no estoque
	3. Se houver o produto no estoque incluir o produto na lista de compras do cliente.
4. Terminado a compra o cliente escolhera o modo de pagamento	
	5. O sistema concluirá o pedido e será gerado o pagamento conforme o cliente escolher.

**Tabela 4: Descrição caso de uso Venda.**

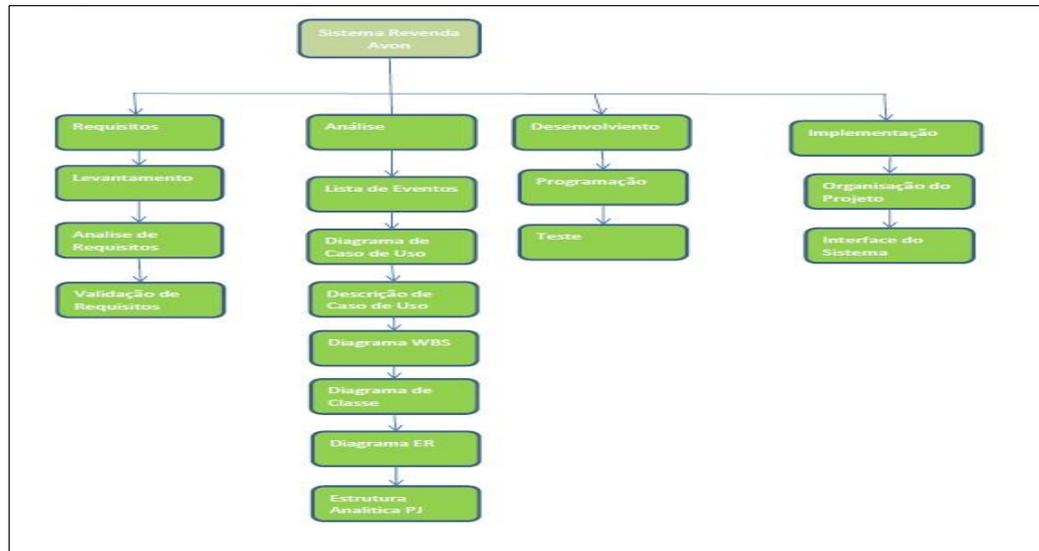
Contudo os diagramas de caso de uso foram utilizados para mostrar de um modo flexível as funcionalidades do sistema e utilizado também para ajudar no levantamento de requisitos.

### 3.4 DIAGRAMA DE WBS(WORK BREAKDOWN STRUCTURE)

*Work Breckdown Struture* (WBS), está metodologia tem a finalidade de demonstrar como é a formatação, bem como de demonstrar a formatação, a forma e a estrutura do desenvolvimento das etapas em que o sistema é elaborada. Trata-se de uma técnica de divisão do projeto em pastas menores, permitindo um controle mais eficiente e de fácil acompanhamento [1].

Sua estrutura e desenvolvida em um formato de arvore hierárquica onde temos as atividades necessita para percepção podendo acompanhar a revolução do estado do projeto pois ao controlar uma etapa de uma atividade teremos a tarefa correspondente finalizada.([1], Disponível em <> <http://WWW.UML.org/>>).

O diagrama WBS auxilia no desenvolvimento do sistema, mostrando passo a passo em sua devida ordem o que deve ser feito no desenvolvimento do sistema.



**Figura 6 – Ilustra o Diagrama de WBS.**

### 3.5 DIAGRAMAS DE CLASSE.

O diagrama de Classe Figura 7 descreve como o sistema e estruturado suas classes e seus devidos atributos e ligações existentes entre as classes.

Segundo G. Guedes (2011).

O Diagrama de Classe é o mais importante e mais utilizado da UML. Seu principal enfoque é a permitir a visualização das classes que compõem o sistema com seus respectivos atributos e métodos, bem como em demonstrar como as classes se relacionam completamente e transmitem informações entre si. Este diagrama representa uma visão estática de como as classes estão organizadas, preocupando-se em como definir a estrutura lógica das mesmas.

A figura 7 demonstrará as classes utilizadas no sistema e seus atributos e demonstra também como as classes do diagrama se relacionam.

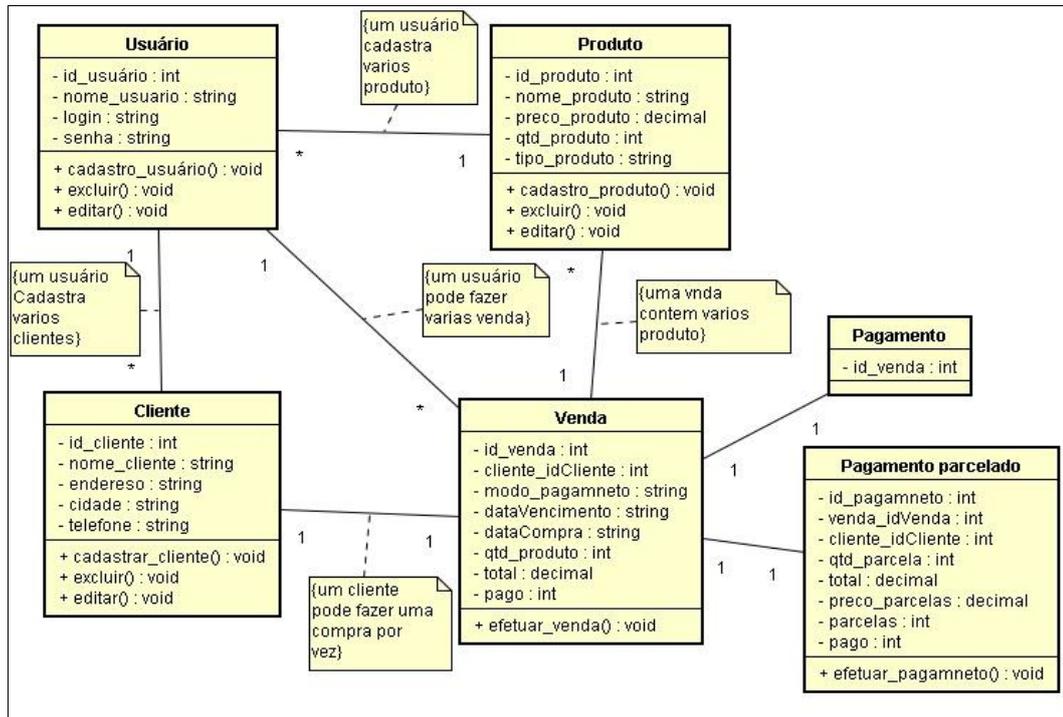


Figura 7 – Ilustra Diagrama de Classe.

### 3.6 DIAGRAMAS ENTIDADE RELACIONADA.

A figura 8 demonstra de um modo visual como será a construção o banco de dados e suas devidas ligações.

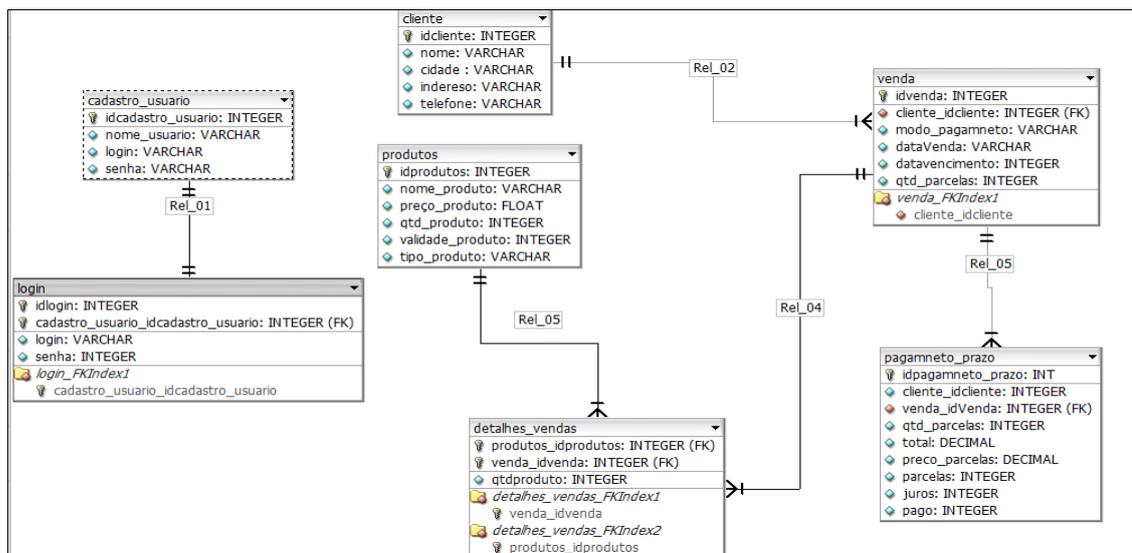


Figura 8 – Ilustra o Diagrama Entidade Relacionada.

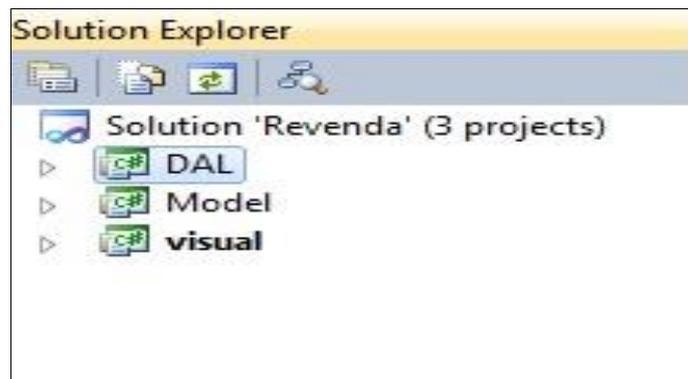
O diagrama de Entidade Relacionada Facilitara na construção do banco de dados mostrando com mais clareza a necessidade de cada tabla e como será as ligações entre as tabelas.

## 4. IMPLEMENTAÇÃO

Para a Implementação do Sistema foi utilizada a plataforma. NET com linguagem c#.

### 4.1 Organizações do Projeto.

Para a Organização do projeto, o Sistema foi desenvolvido em quatro camadas (Figura 9) Camada BLL Significa *Business Logic Layer*, e a camada onde e feito às regras de negócios. DAL significa *Data Access Layer*, camada que e responsável a acesso a dados, Modal, Visual.



**Figura 9 – Camadas.**

Camada DAL (Figura 10): tem a função de criar os métodos que fazem ligação com o banco de dados. Nela foi criada uma conexão onde foram feitos alguns métodos que podem ser aproveitados por todos os estabelecidos, como por exemplo, os métodos executarComando(Figura 11) que executa comandos como Insert e Update, retornando false se o comando falhar e verdadeiro se for executado corretamente e o método consultar(Figura 12) que executa todos os comandos select que retornam uma dataTable, podendo ser usado por todos os métodos de consulta, não importando a tabela que será consultada.

```

DAL.DAL_cliente
- conn

namespace DAL
{
    public class DAL_cliente
    {
        Conexao conn = new Conexao();
        string tabelaRetorno = "cliente";

        public bool incluir(Cliente clien)
        {
            try
            {
                string sql = "INSERT INTO cliente(nome, endereso, cidade, telefone)" +
                    "VALUES ( @Nome, @Endereso, @Cidade, @Telefone)";

                List<SqlParameter> objparam = new List<SqlParameter>();
                objparam.Add(new SqlParameter("@nome", clien.Nome));
                objparam.Add(new SqlParameter("@Endereso", clien.Endereso));
                objparam.Add(new SqlParameter("@Cidade", clien.Cidade));
                objparam.Add(new SqlParameter("@Telefone", clien.Telefone));

                return conn.executarComando(sql, objparam);

            }
            catch
            {
                return false;
            }
        }
    }
}

```

Figura 10 – DAL.

```

public bool executarComando(string p_strSql, List<SqlParameter> p_obParams)
{
    bool blnResult = false;
    if (!this.conectar())
    {
        return false;
    }

    SqlCommand objCmd = new SqlCommand(p_strSql, objConn);

    foreach (SqlParameter param in p_obParams)
    {
        objCmd.Parameters.Add(param);
    }

    try
    {
        blnResult = (objCmd.ExecuteNonQuery() > 0 ? true : false);
    }
    catch
    {
        blnResult = false;
    }

    this.desconectar();

    return blnResult;
}

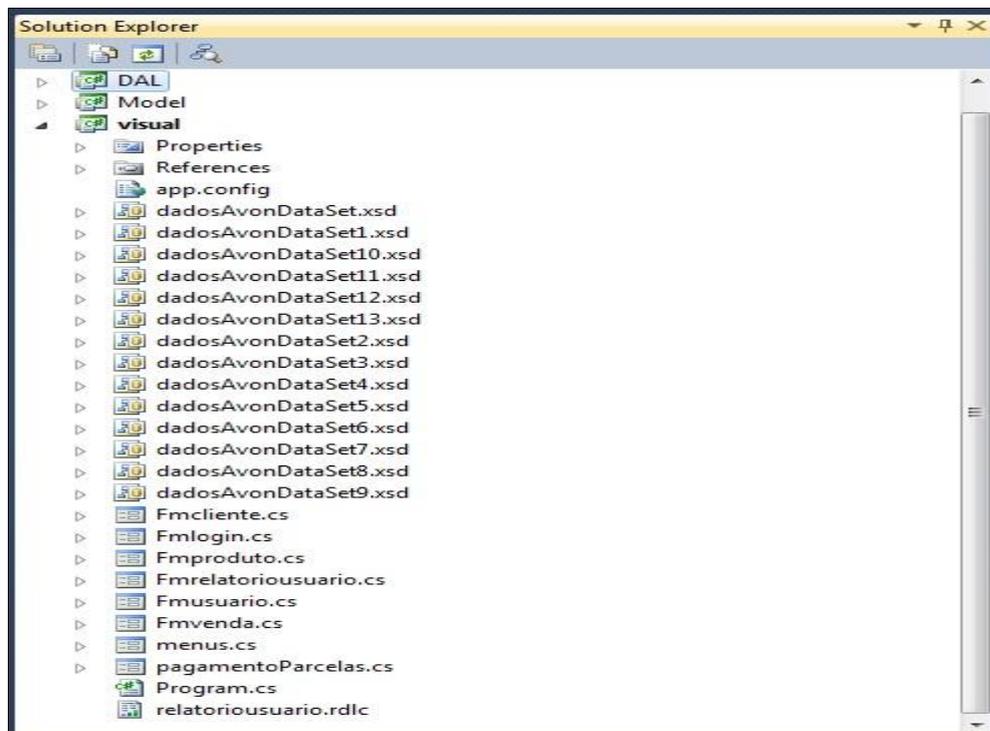
```

Figura 11 – executarComando.

```
public DataTable Consultar(string p_strSql, string p_strNmTabelaRetorno)
{
    conectar();
    SqlCommand objCmd = new SqlCommand(p_strSql, objConn);
    SqlDataAdapter objAdp = new SqlDataAdapter(objCmd);
    DataSet ds = new DataSet();
    objAdp.Fill(ds, p_strNmTabelaRetorno);
    desconectar();
    return ds.Tables[p_strNmTabelaRetorno];
}
}
```

**Figura 12 – consultar.**

Camada Visual (Figura 13) e a camada onde estão às interfaces do sistema, que tem como referência a camada DAL e a camada Modal.

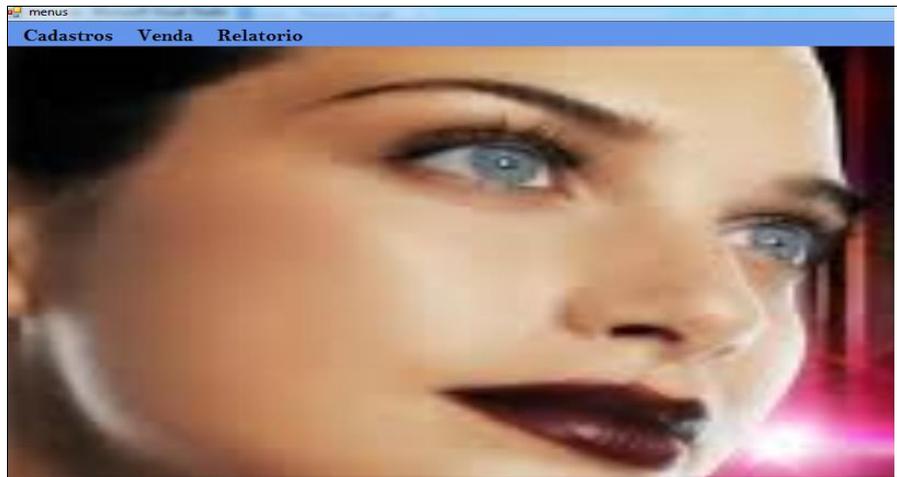


**Figura 13 – visual.**

A imagem acima ilustra a camada Visual onde esta armazenada a as interfaces do sistema e os *DataSet* que são utilizados pelas *GridView*.

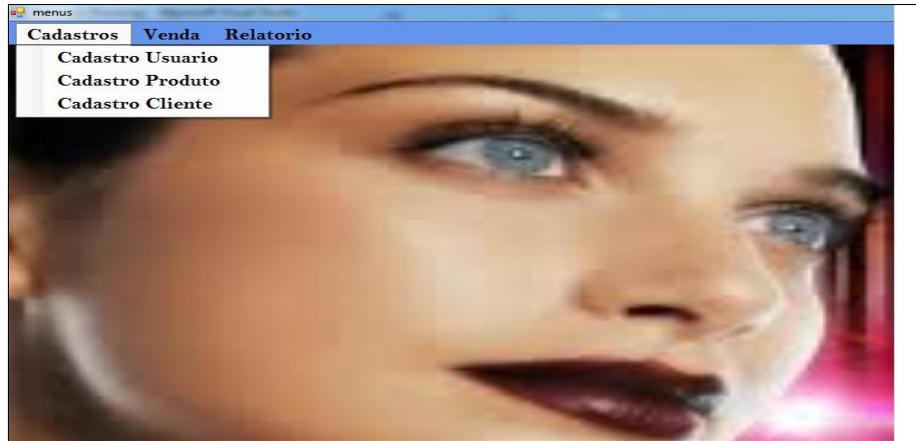
## 4.2 Interface do Sistema

Ao acessar o Sistema vai ser liberar o formulário do menu(Figura 14), com três opções cadastro, clicando sobre ele abrirá mais três opções, quais sejam o cadastro de usuário, o cadastro de produto, cadastro de cliente. No menu terá também a opção de venda, de pagamento a prazo, e a opção de gerar relatórios.



**Figura 14 – menu.**

Ao clicar em cadastro irá abrir uma Aba com varias opções de cadastro.



**Figura 15 – cadastros.**

O cadastro do produto, onde o usuário pode dar entrada aos produtos no estoque, podendo excluí-los. Para excluir o usuário terá que clicar duas vezes em cima do produto que deseja eliminar, abrirá uma janela de advertência perguntando se deseja excluir realmente o produto.

 A screenshot of a software application window titled 'produto' with the subtitle 'Cadastro Produto'. The form contains several input fields: 'Nome:', 'Validade:', 'Preço:', 'Tipo:', and 'Quantidade:'. Below the fields are three buttons: 'Salvar', 'Alterar', and 'Buscar'. At the bottom, there is a table with the following data:
 

Produto	Preço	Quantidade	Validade	Tipo
hidratante SOU	20,50	2	16/10/2014	cosméticos
kalak Pulao	80,00	4	12/02/2014	perfumaria
Sabonete	30,00	4	12/03/2014	cosméticos

**Figura 16 – cadastro produto.**

Ao clicar na opção venda abrirá duas opções, uma delas é a de efetuar venda (Figura 17), o qual o usuário irá dar entrada no nome do cliente que fará a compra.

Ao salvar os dados da compra do cliente, será liberado para escolha dos produtos e a escolha de quantas parcelas será dividida a compra.

The screenshot shows a software interface for recording sales. The window is titled "Form1" and "Venda". It contains several input fields and a table.

**Venda Section:**

- Cliente:
- Data Venda:
- Data Vencimento:
- Modo Pagamento:
- Buttons: Salvar dados

**Produtos Section:**

- Produtos:
- Quantidade Parcelas:
- Preço Unitário: ...
- Quantidade:
- Estoque: ...
- Buttons: Incluir, Alterar
- Total da Venda: R\$ 0

**Sales History Table:**

Cliente	Data da Venda	Data de Vencimento	modo de Pagamento	Vend
Lidia Di Loreto	16/10/2013	16/11/2013	Prazo	
Giovani	16/10/2013	16/11/2013	Prazo	
daLucia	12/10/2013	12/11/2013	Prazo	
Giovani	01/01/2013	01/02/2013	Prazo	

**Product Selection Table:**

Produto	Tipo	Preço Unitário	Quantidade

Figura 17 – Efetuar Vendas.

## 5. Orçamento

- 01 Analista – Programador.
- 01 Notebook Asus core i3, 6 Gigas de memoria RAM, HD de 500 Giga.
- 01 Multifuncional HP.
- Visual Studio 2010.
- Sql Server 2008, programa já incluído com visual Studio 2010.

### 5.1 Estimativa de custos.

Custo Analista – Programador

Custo Diário: R\$ 50,00(Quarenta e cinco Reais).

Total de Dias: 90.

Custo Total:  $(90 * 50,00) = R\$ 4.500,00$ (Quatro mil e quinhentos Reais).

### 5.2 Custo de Materiais

Notebook: R\$ 1.250,00(mil e duzentos e cinquenta reais).

Depreciação 3 anos:  $R\$ 1.250,00 / 36$  (meses) = R\$ 34,72 (trinta e quatro reais e setenta e dois centavos) por mês.

Custo de um dia:  $R\$ 34,72 / 30$  (dias) = R\$ 1,15 (um real e quinze centavos) por dia.

Custo de 167 dias:  $R\$ 1,15 * 167$  (dias) = R\$ 192,05 (cento e noventa e dois reais e cinco centavos).

Multifuncional: R\$ 300,00 (trezentos reais).

Depreciação 3 anos : R\$ 300,00 / 36 (meses) = R\$ 8,35 (oito reais e trinta e cinco centavos) por mês .

Custo de um dia: R\$ 8,35 / 30 (dias) = R\$ 0,28 (vinte e oito centavos)por dia.

Custo de 167 dias: R\$ 0,28 \* 167 (dias) = R\$ 46,76 (quarenta e seis reais e setenta e seis centavos).

Visual Studio 2010: R\$ 890,00 (oitocentos e noventa reais).

Depreciação 3 anos:R\$ 890,00 / 36 (meses) = R\$ 24,75 (vinte e quatro reais e setenta e cinco centavos)por mês.

Custo de um dia: R\$ 24,75 / 30 ( dias) = R\$ 0,83 ( oitenta e três centavos)por dia.

Custo de 167 dias:R\$ 0,83 \* 167 (dias) = R\$ 138,61 (cento e trinta e oito reais e sessenta e um centavos).

Sistema operacional Microsoft Windows 7 Ultimate : R\$ 700,00 (setecentos reais).

Depreciação 2 anos:R\$ 700,00 / 24 (meses) = R\$ 29,20 (vinte e nove reais e vinte centavos)por mês.

Custo de dia: R\$ 29,20 / 30 (dias) = R\$ 0,98 ( noventa e oito centavos)por dia.

Custo de 167 dias: R\$ 0,89 \* 167 ( dias) = R\$ 148,63 (cento e quarenta e oito e sessenta e três centavos).

### 5.3 Orçamento do Projeto

<b>Recursos Utilizados</b>	<b>Valor</b>
<b>Analista – Programados</b>	<b>R\$ 4.500,00</b>
<b>Notebook</b>	<b>R\$ 1.250,00</b>
<b>Multifuncional</b>	<b>R\$ 300,00</b>
<b>Visual Studio 2010</b>	<b>R\$ 890,00</b>
<b>Windows 7 Ultimate</b>	<b>R\$ 700,00</b>
<b>Total</b>	<b>R\$ 7.640,00</b>

**Tabela 5 – Orçamento dos Produtos.**

## **6. CONCLUSÃO**

O sistema acima apresentado irá auxiliar no controle do fluxo e de estoque, do qual diminuirá às perdas e às compras que não tem muita procura.

Sendo este um auxílio que terá um modo fácil e rápido, de modo que conterà um simples manuseio.

Sistema este que não será especificamente só para revendedores Avon podendo ser utilizado para várias vendas de todos os tipos de produtos que possam conter as mesmas especificações. Não atingindo com isso uma única espécie de clientes, mas sim um grupo diverso destes.

### **6.1 TRABALHOS FUTUROS**

A proposta para trabalhos futuros primeiramente e de continuar com o sistema, dar continuidade nas versões e fazer aprimoramentos como impressões de notas fiscais ligações com máquinas de cartões de crédito.

E um Web Server que possibilitara o comerciante sair nas ruas para vender e ter em mãos um dispositivo móvel e podendo assim ter conexão com o seu banco de dados e podendo assim efetuar a venda.

## REFERÊNCIAS

- BEIGHLEY, Lynn; **SQL: Use a cabeça!** Editora: Alta Books. [www.AtaBooks.com.br](http://www.AtaBooks.com.br).
- FREEZE, Wayne; **SQL: Guia de Referencia do Programador**. Editora: Ciência Moderna; Rio de Janeiro; 1998.
- GUEDES, Guilherme T. A: **UML 2: Uma abordagem pratica / Gilleanes T. A. Guedes**. Editora: Novatec, São Paulo, 2011.
- GUNNERSON, Eric: **Introdução à programação em C#**. Editora: Ciência Moderna LTDA.; Rio de Janeiro: 2001. Linguagem de Programação para microcomputadores [http://www.macoratti.net/oo\\_conc2.htm](http://www.macoratti.net/oo_conc2.htm).
- MACKEY, Alex: **Introdução ao .Net 4.0**. Editora: Alto Books. ; Rio de Janeiro 2011.
- SHARP, John: **Microsoft Visual C#2008: passo a passo**. Tradução Edson Furmankiewiz; Editora: Bookman; Porto Alegre 2008.
- SILVA, ANDERSON. ; **C# .Net Guia do Desenvolvedor: Distribui mento e distribuição aplicação Web e de Desktop**, Editora Campo, Rio de Janeiro, 2002.
- STELLMAN, Andrew: **Use a cabeça! C#**. GREENE, Jennifer; Tradução Eveline Machado; 2º Edição; Editora: Books; Rio de Janeiro; 2011.