

UTILIZAÇÃO DA UML NO DESENVOLVIMENTO DE SISTEMA DE CONTROLE DE VENDAS E ESTOQUE

GILBERTO FRANCISCO PACHECO DOS SANTOS

Discente da AEMS

Faculdades Integradas de Três Lagoas

JACKSON LUIZ ARROSTI

Discente da AEMS

Faculdades Integradas de Três Lagoas

LUCAS FREIRE DE OLIVEIRA RODRIGUES

Discente da AEMS

Faculdades Integradas de Três Lagoas

ALAN PINHEIRO DE SOUZA

Docente da AEMS

Faculdades Integradas de Três Lagoas

Mestre em Informática

Área de Sistemas de Informação

O objetivo do trabalho foi desenvolver um sistema de vendas e controle de estoque. Para a sua construção foram utilizados processos, métodos e ferramentas da engenharia de sistemas, requisitos e projeto. O levantamento de requisitos ocorreu a partir da realização de entrevistas junto ao cliente e as atividades de modelagem foram conduzidas pela adoção da linguagem UML. O processo de engenharia promoveu a absorção e documentação de abrangente conteúdo informacional e a geração de modelos precisos e completos a respeito do negócio que culminaram na construção de sistema de *software* visando automatizar as principais tarefas do negócio da empresa, atendendo as características de *software* e sistema solicitadas pelo cliente.

Palavras-Chaves: estoque, modelagem, requisitos, UML, vendas.

1. INTRODUÇÃO

O trabalho propõe-se ao desenvolvimento de um sistema de gestão para uma empresa comercial de modo a facilitar e agilizar a organização das informações para o administrador e demais funcionários do negócio. Atualmente, a empresa possui vários modelos para gerenciar seus dados (planilha eletrônica, fichas preenchida a mão, entre outros), e isso torna difícil a sua administração. Um sistema informatizado agilizaria as atividades da empresa na administração e organização dos dados.

Acredita-se que um sistema informatizado é uma forma mais viável para diminuir o tempo com operações de administração dos dados. Além disso, uma ferramenta computacional aumentará a segurança e a vida útil dos dados, e diminuirá a chances de perdas, falta de dados para análise ou riscos de exposição de informações confidenciais. Além disso, são menos suscetíveis às intempéries do ambiente em relação a outros meios de armazenamento. A automatização das tarefas permite a obtenção de informações de maneira mais rápida e auxilia as tomadas de decisões. A vantagem de se utilizar um sistema informatizado vai além do ambiente empresarial, pois além dos benefícios internos, o cliente também é beneficiado pelo seu uso no dia-a-dia.

Entre os desafios do trabalho, há o esforço extra com relação aos dados em meios externos de informação (planilhas e papéis) a serem transferidos para o sistema, além do trabalho manual para digitalização destas informações. Outra preocupação é a possibilidade de ocorrências de incompatibilidades de informações, por exemplo: vendas registradas sem informações de clientes ou orçamentos sem informações de vendas ou serviços relacionados. Após a implantação do sistema espera-se facilitar o trabalho de organização dos dados da empresa, realizar movimentação de vendas, relatórios financeiros e controle de estoque.

O trabalho está organizado em quatro seções, são elas: Introdução, Linguagem de Modelagem UML, Contexto de Aplicação e Conclusão. Cada seção tem suas particularidades essenciais para o projeto. Na primeira seção foi apresentada uma visão geral do projeto e a estrutura de construção do artigo. Na segunda seção destaca uma revisão bibliográfica sobre a linguagem de modelagem UML. A terceira seção apresenta a modelagem aplicada ao contexto do projeto,

destacando alguns dos diagramas elaborados ao longo das etapas de análise e projeto. A quarta seção destaca as contribuições e trabalhos futuros do projeto.

2. LINGUAGEM DE MODELAGEM UML

A UML (*Unified Modeling Language*) é uma linguagem de modelagem de sistemas com objetivo de auxiliar engenheiros a definir requisitos, comportamentos, estruturas lógicas e até mesmo necessidades físicas do sistema a ser implantado (BOOCH; RUMBAUGH; JACOBSON, 2000). A Figura 1 destaca arquitetura ou conjunto de decisões significativas acerca da organização do sistema de *software*, seleção dos elementos estruturais e suas interfaces, comportamentos, entre outros itens importantes de projeto.



Figura 1 – Modelagem da arquitetura de um sistema.

Os modelos foram úteis para elaboração da estrutura do *software*, além de permitir visualização, especificação, construção e documentação dos artefatos de projeto. Os diagramas contribuíram substancialmente para desenvolver um *software* que satisfaça aos propósitos pretendidos pelo cliente e com qualidade duradoura (BOOCH; RUMBAUGH; JACOBSON, 2000; FURLAN, 1998). Além disso, a modelagem UML trouxe como benefícios a possibilidade de desenvolver um sistema de forma mais rápida e eficiente, com o mínimo de desperdício e retrabalhos de *software* (MATOS, 2002).

3. CONTEXTO DE APLICAÇÃO

A empresa atua dois anos no mercado aplicando decorações de persianas, papel de parede, pisos laminados e forros com tecido, em ambientes domésticos e comerciais e também vende produtos relacionados. Atualmente, a empresa realiza todos os movimentos diários por intermédio do uso de escrituração em papel e não possui sistemas de informação computacionais para apoiar as tarefas do negócio.

3.1. Requisitos do Usuário e Sistema

Com base em estudos acadêmicos e do funcionamento do ambiente de trabalho, desenvolveu-se nesta fase do trabalho o levantamento de requisitos e sua classificação e organização para o entendimento inicial dos problemas que deveriam ser resolvidos e funcionalidades que o sistema desenvolvido deverá oferecer aos seus usuários. Aplica-se aqui a análise dos requisitos funcionais e não funcionais do usuário. Segundo a definição de Sommerville (2007):

“Os requisitos de usuários para um sistema devem descrever os requisitos funcionais e não funcionais de modo compreensível pelos usuários do sistema que não tem conhecimentos técnicos detalhados. Eles devem especificar somente o comportamento externo do sistema evitando tanto possível as características do projeto de sistema” (SOMMERVILLE, 2007).

Os requisitos funcionais descrevem as funcionalidades triviais que o sistema deve oferecer, para os requisitos de usuários eles são expressos informalmente, quando pertencem aos requisitos funcionais do sistema descrevem, especialmente, seu processamento, entrada e saída. Já os requisitos não funcionais correspondem às necessidades do sistema para seu funcionamento, como tratamento de erros, confiabilidade, segurança e entre outras características técnicas. As Tabelas 1 e 2 mostram, respectivamente, os requisitos funcionais e não funcionais relacionados à operação de cadastro de venda.

Tabela 1 – Requisito funcional de cadastro de movimento de vendas.

REQUISITOS FUNCIONAIS
Uma venda gerada contém os produtos comprados e relacionado a um cliente.

Tabela 2 – Requisitos não funcionais de movimento de vendas.

REQUISITOS NÃO FUNCIONAIS
1. Para cada venda realizada deve-se efetuar baixa no estoque. Para cada produto devolvido deve-se repor o estoque.
2. Cada venda poderá ser parcelada e com valor de entrada. Cada parcela será calculada automaticamente para agilizar o cadastro de venda assim como outros dados que podem ser previsto pelo sistema.

A proposta apontada pelos requisitos do sistema tem o objetivo de especificar as características de funcionamento essenciais para os desenvolvedores, segundo a definição de Sommerville (2007):

“Os requisitos de sistema são descrições mais detalhadas dos requisitos do usuário. Eles podem servir como base para um contrato destinado a implementação do sistema e, portanto, deve ser uma especificação completa e consistente de todo o sistema. Eles são utilizados pelos engenheiros de *software* como ponto de partida para o projeto de sistema” (SOMMERVILLE, 2007).

A Tabela 3 exemplifica os requisitos do sistema relacionado ao processo de pesquisa de dados. Outras tabelas similares foram elaboradas para controle de estado de formulário e processamento de dados de entrada.

Tabela 3 - Requisitos do sistema relacionados ao processo de pesquisa de dados.

Requisitos de Sistema para pesquisa de dados	
Função	Pesquisar dados na base de dados.
Descrição	Realizar pesquisa durante a inserção de informações ou consulta dos campos dos formulários.
Problema	Cada pesquisa realizada na base de dados deverá ser feita no momento da inserção dos caracteres nos campos do formulário.
Resolução	Usar campos para pesquisa no banco de dados no formulário
Entradas	Conjunto de caracteres do campo do formulário.
Saída	Resultado da pesquisa em uma tabela do formulário.
Evento	Ao inserir cada caractere no campo de formulário.
Pré-condição	Campo de caracteres não vazios.
Pós-condição	Caso haja dados encontrados disponibilizá-lo através de uma interface para o usuário.

Usando os modelos detalhados e de contexto da UML analisou-se o comportamento externo do sistema e suas funcionalidades que são esquematizadas por seus diagramas A Tabela 4 aponta os eventos modelados no presente projeto:

Tabela 4 – Lista de eventos do projeto.

<ul style="list-style-type: none"> • Cadastro de pessoa física • Cadastro de pessoa jurídica • Cadastro de produto • Movimento de vendas 	<ul style="list-style-type: none"> • Relatório de pessoas • Relatório de vendas • Relatório de produtos
--	--

3.2. Modelagem de Análise e Projeto

A modelagem de sistemas é usada para representar o domínio do sistema, arquitetando a estrutura primordial para início da análise do problema da construção do *software*. Um dos primeiros modelos elaborados foi o diagrama de classes, onde

as classes representam as entidades a serem operadas pelo sistema e a sua interação durante o seu funcionamento (DEBONI, 2003). A Figura 2 ilustra o diagrama de classes modelado no contexto do projeto.

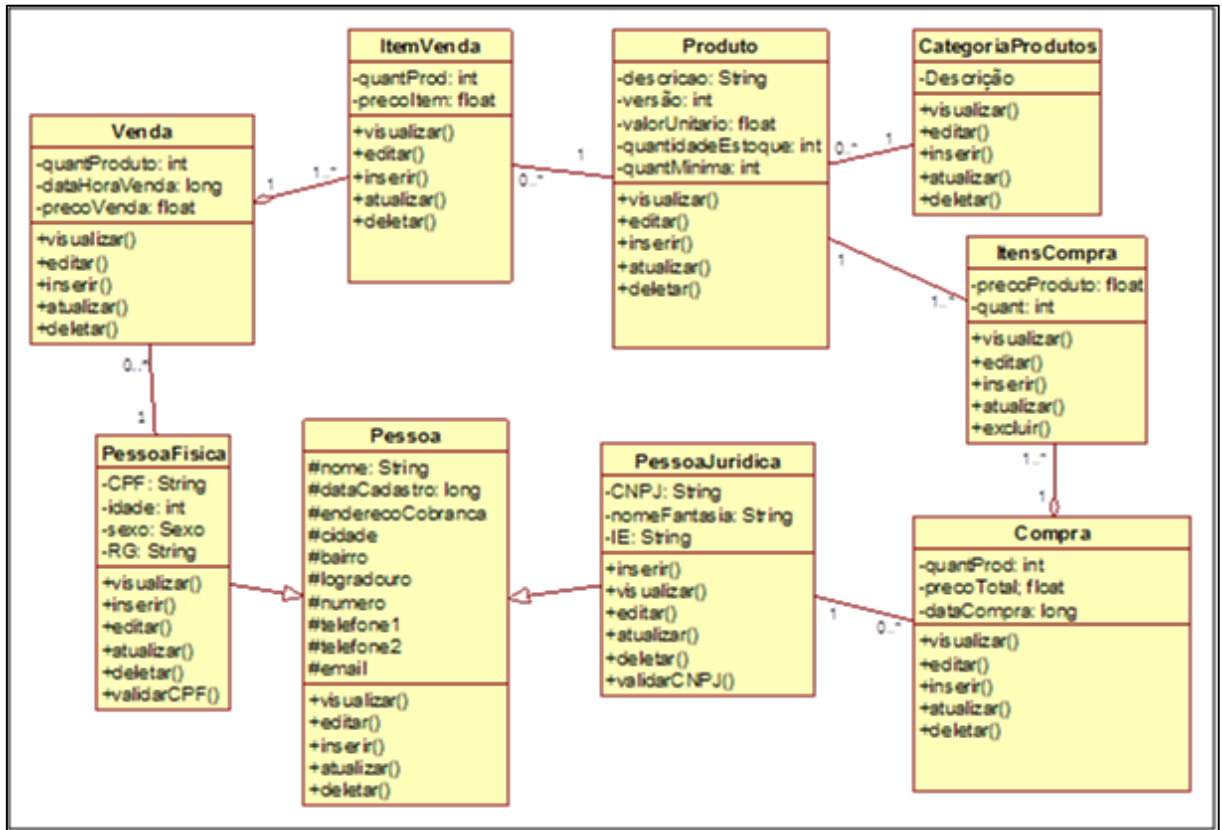


Figura 2 – Diagrama de classes.

Além disso, foram elaborados modelos de casos de uso para cada evento da Tabela 4, destacando seus roteiros normais e alternativos de execução, atores envolvidos e pacotes de dados manipulados e acessados durante a interação do usuário com sistema. A Figura 3 apresenta o diagrama de caso de uso geral do sistema com seus cadastros, controles e geração de relatórios, nesse esquema foi elaborada uma generalização do cadastro de pessoa para oferecer funcionalidades comuns para cadastro de pessoa física e cadastro de pessoa jurídica. Além disso, existem também funcionalidades de extensões para outras operações, por exemplo, ao realizar uma venda é necessário verificar se o cliente está cadastrado no sistema, caso não estiver é preciso cadastrá-lo.

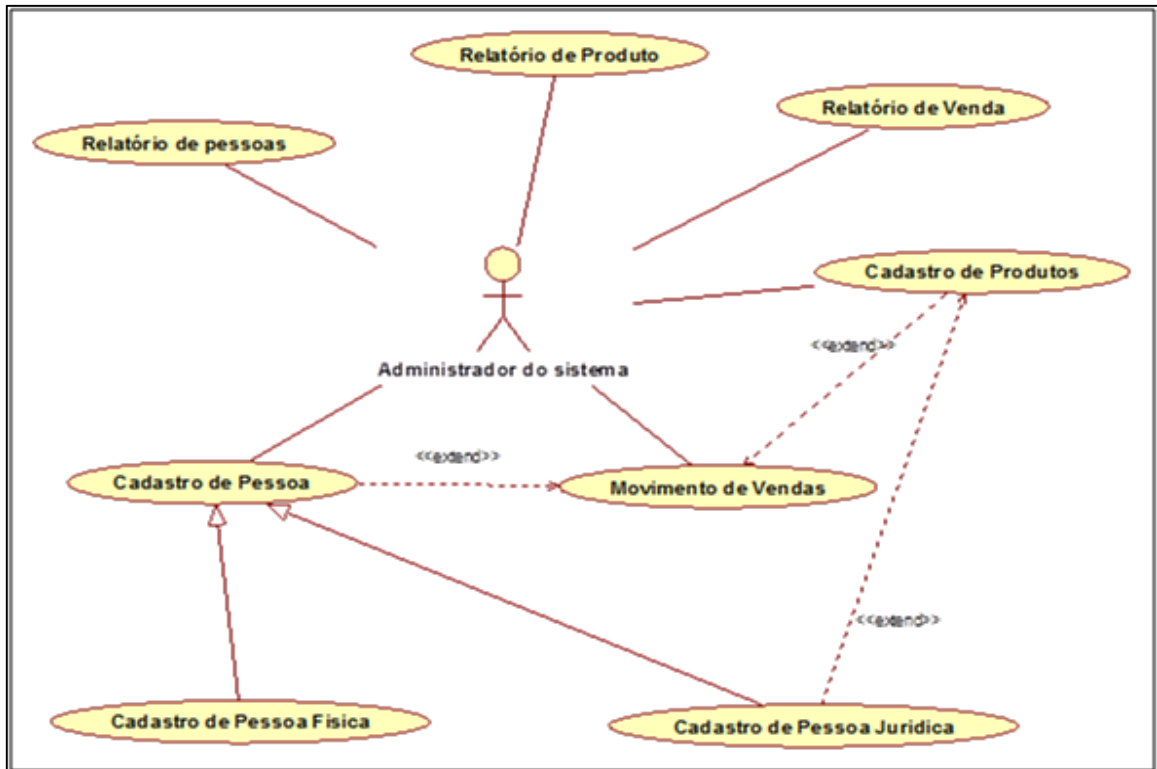


Figura 3 – Diagrama de caso de uso geral.

Além dos modelos destacados anteriormente, vários outros diagramas foram elaborados na etapa de projeto do processo de engenharia do sistema, entre eles: seqüências, estados, atividades e implantação. A Figura 4 apresenta diagrama de componentes utilizado para moldar as unidades de código do sistema, onde cada componente sugere códigos com funcionalidade em comum, de acordo com a visão de (MATOS, 2002).

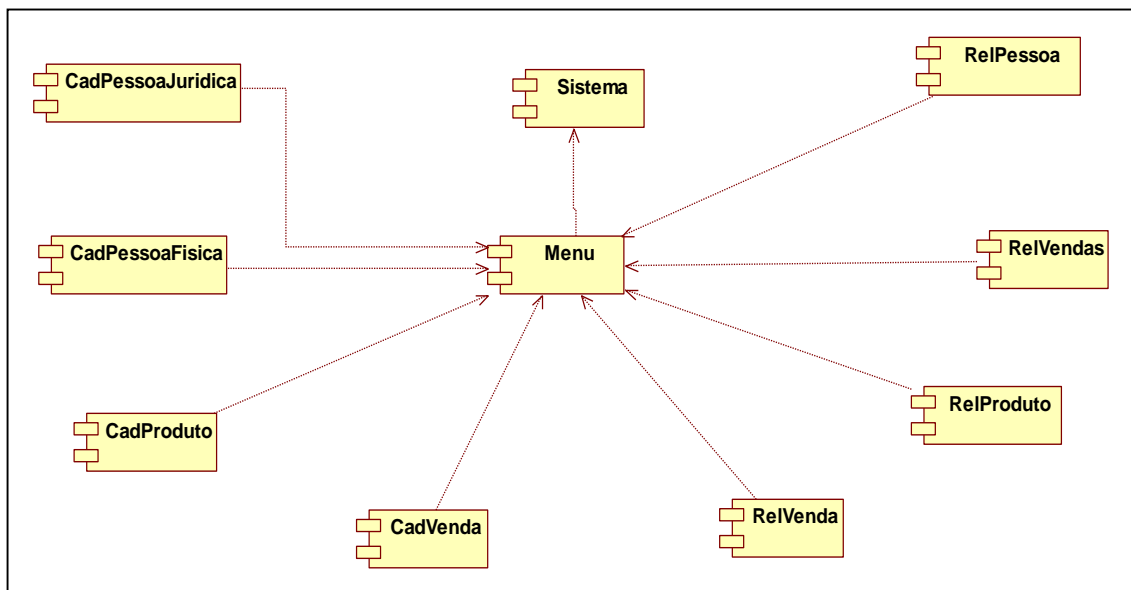


Figura 4 – Diagrama de componente do sistema

3.3. Sistema de Controle de Vendas e Estoque

A modelagem dos diagramas de análise e projeto forneceu subsídios para as etapas seguintes de codificação e teste do sistema. Dessa forma, ocorreu a construção de sistema operacional e gerencial de controle de vendas e estoque com as características descritas nas telas descritas na sequência.

Com exceção a tela principal, as telas do sistema são divididas em três seções: cadastro, consulta e detalhes para edição. A Figura 5 mostra a tela principal que possui botões para acessar funcionalidades de clientes, vendas, fornecedores, compras e estoque. A Figura 6 destaca o formulário de vendas do sistema quando acessado em seu modo de consulta.



Figura 5 – Tela principal do sistema.

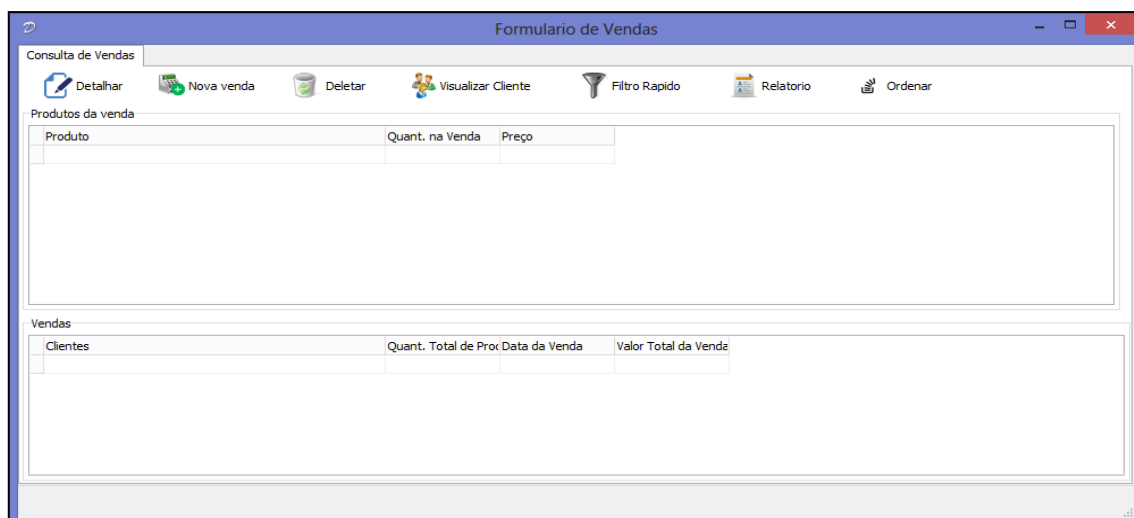


Figura 6 – Formulário de vendas no modo consulta.

As Figuras 7 e 8 continuam a mostrar a funcionalidade de vendas do sistema, com destaques para seu modo de detalhes e edição e modo de vendas aberto, respectivamente. No modo detalhes e edição de vendas é possível visualizar detalhes da venda e editá-los, nesse momento, a guia detalhes e edição de vendas fica visível junto com os controles de edição.

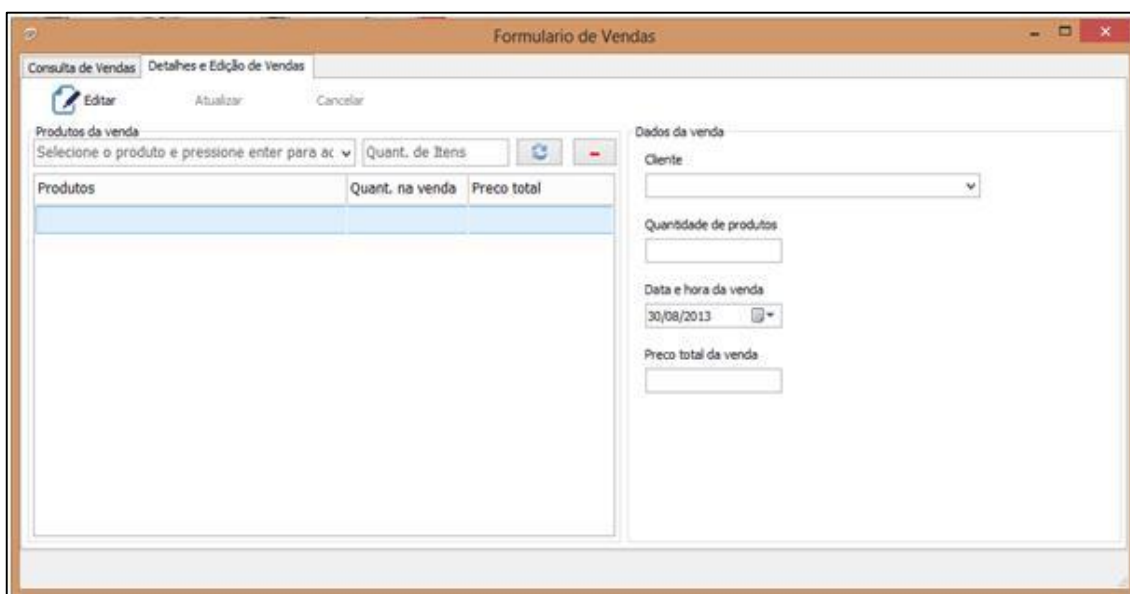


Figura 7 – Formulário de vendas no modo detalhes e edição.

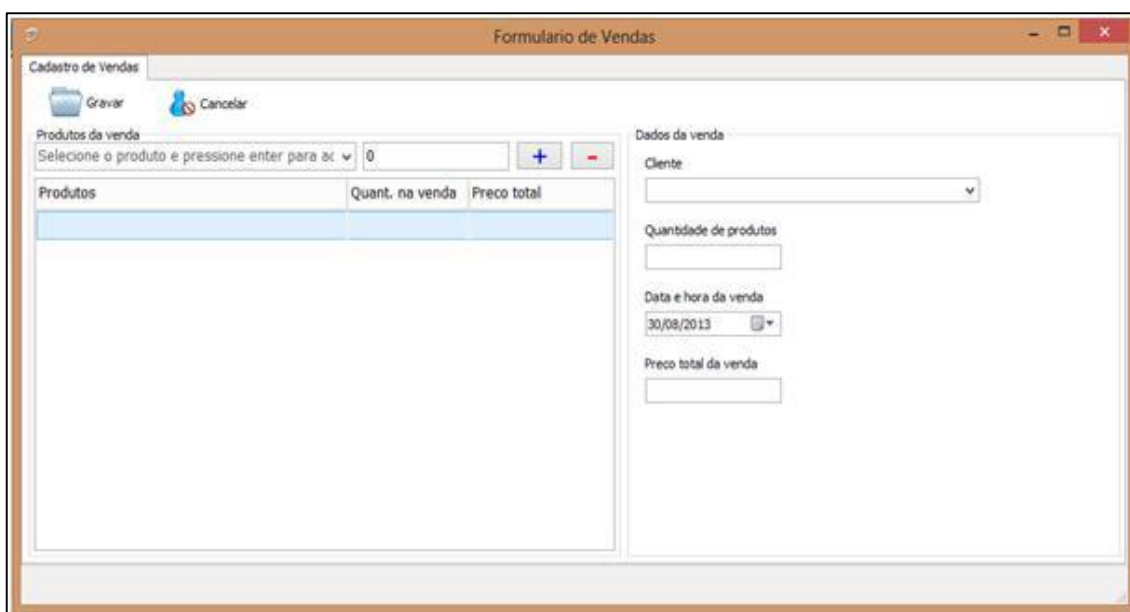


Figura 8 – Formulário de vendas no modo de cadastro.

No modo de cadastro, as vendas são inseridas juntas com as informações dos produtos da venda. Nesse momento, a guia cadastro de vendas torna-se visível e outras guias ficam ocultas. Por outro lado, quando um cadastro é gravado ou cancelado, a guia cadastro fica oculta e a guia consulta de vendas torna-se visível.

4. CONCLUSÃO

Neste trabalho aplicaram-se processos e técnicas de análise e projeto para o desenvolvimento do sistema. Por intermédio da avaliação de requisitos separaram-se as funcionalidades que o sistema proposto deveria fazer abordando as diferentes características solicitadas pelo cliente para o sistema.

Durante o desenvolvimento do sistema houve impossibilidade para implementar certos requisitos, por exemplo, o cliente desejava obter uma lista telefônica eletrônica no sistema para reconhecer possíveis clientes futuros. Esse recurso não foi possível acomodar no projeto. Além disso, outras entidades do sistema foram excluídas do escopo do projeto para não estender demasiadamente o cronograma de desenvolvimento do trabalho.

Como trabalhos futuros, existe a necessidade do cliente para que sejam adicionados alguns requisitos de modo a agregar mais funcionalidades ao projeto, especialmente, características de orçamentos e serviços. Esses requisitos destinam-se, respectivamente, fazer balanços sem compromissos e registrar todos os serviços prestados pela empresa para os seus clientes no sistema eletrônico.

REFERÊNCIAS BIBLIOGRÁFICAS

BOOCH, G.; RUMBAUGH, J.; JACOBSON, I. **UML: Guia do Usuário**. 12ª Ed., Rio de Janeiro: Campus, 2000.

DEBONI, J. **Modelagem Orientada a Objetos com UML**. São Paulo: Futura, 2003.

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

MATOS, A. **UML: Prático e Descomplicado**. São Paulo: Érica, 2002.

SOMMERVILLE, I. **Engenharia de Software**. 8ª Ed., São Paulo: Addison Wesley Brasil, 2007.