

INSTITUTO FEDERAL DE EDUCAÇÃO, CIÊNCIA E TECNOLOGIA DO SUDESTE
DE MINAS GERAIS
CAMPUS AVANÇADO BOM SUCESSO

WANDERSON HENRIQUE CASTANHEIRA

DESENVOLVIMENTO DE UM SISTEMA DESKTOP PARA O CONTROLE E
GERENCIAMENTO DE VENDAS DA EMPRESA NEW ART COLOR

BOM SUCESSO

2021

WANDERSON HENRIQUE CASTANHEIRA

DESENVOLVIMENTO DE UM SISTEMA DESKTOP PARA O CONTROLE E
GERENCIAMENTO DE VENDAS DA EMPRESA NEW ART COLOR

Trabalho de Conclusão apresentado ao
Câmpus Avançado Bom Sucesso, do
Instituto Federal de Educação Ciência e
Tecnologia do Sudeste de Minas Gerais,
como parte das exigências do Curso de
Graduação de Tecnologia em Análise e
Desenvolvimento de Sistemas para
obtenção do título de Tecnólogo.

Orientador: Prof. Pedro Henrique de
Oliveira e Silva

Coorientador: Prof. Victor Schmidt Comitti

BOM SUCESSO

2021

Dados internacionais de catalogação na publicação (CIP)
Bibliotecária responsável Maria de Lourdes Cardoso CRB-6/3242

C346d Castanheira, Wanderson Henrique, 1998 -

Desenvolvimento de um sistema desktop para o controle e gerenciamento de vendas da empresa New Art Color / Wanderson Henrique Castanheira. -- 2021.

47 f. : il. ; 30 cm.

Orientador: Pedro Henrique de Oliveira e Silva
Coorientador: Victor Schimdt Comitti

Monografia (Tecnólogo em Análise e Desenvolvimento de Sistemas) - Instituto Federal do Sudeste de Minas Gerais, Campus Avançado Bom Sucesso, Bom Sucesso, 2021.

1. Software - Desenvolvimento. 2. Software de aplicação. 3. UML (Computação). 4. Vendas - Administração. I. Silva, Pedro Henrique de Oliveira e. II. Comitti, Victor Schimdt. III. Instituto Federal do Sudeste de Minas Gerais, Campus Avançado Bom Sucesso. IV. Título.

CDD: 005.36



MINISTÉRIO DA EDUCAÇÃO
INSTITUTO FEDERAL DE EDUCAÇÃO, CIÊNCIA E TECNOLOGIA DO SUDESTE DE MINAS GERAIS

COMPROVANTE DE APROVAÇÃO DO PROJETO POR BANCA EXAMINADORA Nº 9 / 2021 - BSCNA
(11.01.10.01.01.02)

Nº do Protocolo: NÃO PROTOCOLADO

Juiz de Fora-MG, 07 de Abril de 2021

TERMO DE APROVAÇÃO

Wanderson Henrique Castanheira

DESENVOLVIMENTO DE UM SISTEMA DESKTOP PARA O CONTROLE
E
GERENCIAMENTO DE VENDAS DA EMPRESA NEW ART COLOR

Este Trabalho de Conclusão de Curso foi julgado e aprovado como requisito parcial para a obtenção do grau de Tecnólogo em Análise e Desenvolvimento de Sistemas do Instituto Federal de Educação, Ciência e Tecnologia do Sudeste de Minas Gerais - Campus Avançado Bom Sucesso.

(Assinado digitalmente em 07/04/2021 20:27)

HEBER FERNANDES AMARAL
PROFESSOR ENS BASICO TECN TECNOLOGICO
Matrícula: 1147455

(Assinado digitalmente em 07/07/2021 19:24)

PEDRO HENRIQUE DE OLIVEIRA E SILVA
PROFESSOR ENS BASICO TECN TECNOLOGICO
Matrícula: 1758559

(Assinado digitalmente em 07/07/2021 19:48)

VICTOR SCHMIDT COMITTI
PROFESSOR ENS BASICO TECN TECNOLOGICO
Matrícula: 3082930

Para verificar a autenticidade deste documento entre em

Este trabalho é todo dedicado aos meus pais, pois é graças a eles e seus esforços que hoje posso concluir meu curso.

AGRADECIMENTOS

Primeiramente quero começar agradecendo a Deus por ter me concedido essa graça de realizar este projeto, E secundamente quero agradecer meu professor orientador Pedro Henrique de Oliveira e juntamente meu professor coorientador Victor Schmidt Comitti pelas suas colaborações e paciência, por todo tempo e dedicação e orientações, eles tiveram um papel muito importante para a realização deste trabalho, obrigado pelos seus ensinamentos e pela confiança depositada em mim. Quero agradecer também aos meus pais, colegas e amigos por todo apoio que me deram durante a realização do curso Superior de Tecnologia em Desenvolvimento e Análise de Sistemas, foram muitas lutas e dificuldades, momentos em que pensávamos deixar tudo de lado, mais com a benção de Deus e a força de meus amigos e professores foi permitido vencer todos os obstáculos para chegar até aqui. obrigado pelos concelho, pela força pois se não fossem vocês não teria conseguido chegar até aqui.

"Sonhos determinam o que você quer.
Ação determina o que você conquista."
Aldo Novak.

RESUMO

Pequenas empresas, frequentemente, fazem todo o controle de estoque e de caixa manualmente, o que leva a decisões gerenciais e de planejamento ruins. Neste trabalho apresentamos o desenvolvimento de uma ferramenta de Tecnologia da Informação criada para a empresa New Art Color de Bom Sucesso - MG. Esta ferramenta, que foi chamada de SysArtVendas, é capaz de registrar as entradas e saídas de todos os produtos da loja, tornando-se portanto um instrumento essencial para a tomada de boas decisões por parte da gerência. Com isso, espera-se que a New Art Color ganhe uma vantagem competitiva num mercado cada vez mais desafiador.

Palavras-chave: Desenvolvimento de software; RUP e UML.

ABSTRACT

Small companies often do all inventory and cash control manually, which leads to poor management and planning decisions. In this work we present the development of an Information Technology tool created for the company New Art Color of Bom Sucesso-MG. This tool, which was called SysArtVendas, is capable of registering the inputs and outputs of all products in the store, thus becoming an essential tool for making good decisions by management. With that, it is expected that New Art Color will gain a competitive advantage in an increasingly challenging market.

Key-words: Software development; RUP and UML.

LISTA DE FIGURAS

FIGURA 1 - Ciclo de vida do Processo Unificado Rational	21
FIGURA 2 - Modelo mental MVC	25
FIGURA 3 - Modelo MVC	26
FIGURA 4 - Arquitetura do sistema	27
FIGURA 5 - Modelo conceitual inicial do sistema	27
FIGURA 6 - Diagrama de casos de usos	30
FIGURA 7 - Diagrama de classes representação de camada MODEL.....	31
FIGURA 8 - Diagrama de classes representação da camada DAO.....	32
FIGURA 9 - Diagrama de classes representação da camada VIEW.....	32
FIGURA 10 - Diagrama lógico de banco de dados.....	33
FIGURA 11 - Exemplo do código implementado camada DAO.....	34
FIGURA 12 - Tela login do sistema.....	35
FIGURA 13 - Tela inicial do sistema.....	36
FIGURA 14 - Tela gerenciamento de clientes.....	36
FIGURA 15 - Tela gerenciamento de produtos.....	37
FIGURA 16 - Tela dados da empresa.....	37
FIGURA 17 - Tela gerenciamento dos tipos de produtos.....	38
FIGURA 18 - Tela entrada de produtos.....	38
FIGURA 19 - Tela gerenciamento de usuários.....	39
FIGURA 20. Tela gerenciamento de usuários - Definição das Permissões do Sistema.	39
FIGURA 21 - Tela de vendas de produtos.....	40

FIGURA 22: Tela de vendas de produtos - Selecionar os produtos	41
FIGURA 23: Trecho do código implementado na camada MODEL	45
FIGURA 24: Trecho do código implementado na camada DAO	46
FIGURA 25: Trecho do código implementado na camada CONTROLLER	46
FIGURA 26: Trecho do código implementado na camada VIEW	47

LISTA DE QUADROS

QUADRO 1 - Requisitos Funcionais e Não Funcionais do sistema.....	22
QUADRO 2 - Risco identificados	28

LISTA DE SIGLAS

BD - Banco de Dados

CNPJ - Cadastro Nacional de Pessoa Jurídica

CPF - Cadastro de pessoas física

CRUD *Create, Retrieve, Update e Delete*

MVC - *Model View Controller*

PARC - *Palo Alto Research Laboratory Xerox*

RUP - *Rational Unified Process*

SGBD - Sistema de gerenciamento de banco de dados

TI - Tecnologia da Informação

UML - Linguagem de modelagem unificada

XML - *Extensible Markup Language*

SUMÁRIO

1. INTRODUÇÃO	14
1.1 OBJETIVOS	15
1.1.1 Objetivo Geral	15
1.1.2 Objetivos Específicos	15
1.2 ORGANIZAÇÃO DO TRABALHO	16
2. METODOLOGIA	17
3. DESENVOLVIMENTO DO SYSARTVENDAS	18
3.1. SOBRE O SISTEMA	18
3.2. METODOLOGIA DE DESENVOLVIMENTO	18
3.3. RUP – RATIONAL UNIFIED PROCESS	19
3.4. FASES REALIZADAS DURANTE O DESENVOLVIMENTO DO SISTEMA	22
3.4.1. Concepção	22
3.4.1.1. Coleta e especificação dos requisitos	22
3.4.2. Elaboração	25
3.4.3. Construção	28
3.4.3.1. Projeto e Análise	28
3.4.3.2. Codificação	34
4. RESULTADOS E DISCUSSÕES	35
5. CONCLUSÃO E TRABALHOS FUTUROS	42
REFERÊNCIAS	43
APÊNDICE A - Exemplo fragmentos dos códigos desenvolvidos	45

1. INTRODUÇÃO

O avanço da tecnologia de informação (TI) nos últimos anos vem permitindo às empresas executar operações que antes eram inimagináveis, oferecendo recursos tecnológicos e computacionais para a geração de informações, propondo mudanças nos processos, novas estruturas e estratégias de negócios para as empresas.

A TI é uma ferramenta versátil capaz de auxiliar todo tipo de empresas a melhorar seus processos administrativos, sua tomada de decisão gerencial e a colaboração entre os grupos de trabalhos internos. Os ganhos de produtividade decorrentes da tecnologia se traduzem em uma posição competitiva mais sólida dentro do mercado e, por esse motivo, O'Brien (2013) argumenta que um sistema de informação é indispensável para o sucesso de qualquer companhia.

Tendo como perspectiva os problemas que a falta de tecnologia da informação pode trazer às grandes e pequenas empresas, Porter (1989) argumenta que as principais vantagens de um bom sistema de informação são estratégicas. De acordo com o autor, uma organização sempre busca vantagens competitivas em determinado setor. Isso significa que toda companhia está, a todo momento, tentando superar seus concorrentes diretos em custo, qualidade e velocidade. Neste contexto, é essencial que as empresas sejam capazes de processar informação de maneira eficiente, segura e confiável.

Diante deste cenário, o presente trabalho se propõe a desenvolver um sistema para o controle e gerenciamento das vendas realizadas pela empresa New Art Color - uma fabricante de textura e grafiato que tem entre seus principais objetivos o desenvolvimento de produtos utilizados na construção civil, principalmente como linha de acabamento, para oferecerem características decorativas, opções de contraste visual, além de controle para umidade, entre outras utilidades. A empresa está situada na cidade de Bom Sucesso, localizada no sudeste do estado de Minas Gerais. Atualmente a empresa possui uma base de clientes formada e atende profissionais da engenharia e arquitetura, empreiteiros,

empresas de gestão de obras, aplicadores e clientes finais da cidade de Bom Sucesso e região.

No momento, a empresa passa por diversas dificuldades por falta de um sistema de controle e gerenciamento das suas vendas, seu atendimento ao público é feito via balcão e por meio de anotações manuscritas, fazendo com que seu processo de venda se torne desorganizado e lento, ocasionando perdas em suas vendas, ocasionando um grande prejuízo para a empresa. Com o intuito de agilizar o processo de vendas, além de facilitar a apresentação dos produtos, agilizando as negociações, bem como evitando a perda de vendas, torna-se indispensável a utilização de um sistema de informação para o controle e gerenciamento das vendas realizadas pela empresa.

1.1 OBJETIVOS

1.1.1 Objetivo Geral

O objetivo geral deste trabalho é apresentar todas as etapas realizadas durante o desenvolvimento do sistema para o controle e gerenciamento das vendas (SysArtVenda) realizadas pela empresa New Art Color, bem como apresentar o sistema desenvolvido.

1.1.2 Objetivos Específicos

- Desenvolver um software para o controle e gerenciamento de vendas;
- Demonstrar as boas práticas para o desenvolvimento de aplicações, entre elas a utilização da metodologia de desenvolvimento de software RUP (*RATIONAL UNIFIED PROCESS*);
- Demonstrar a utilização de técnicas de modelagem para resolver problemas na área de desenvolvimento de software;
- Demonstrar a utilização do padrão de projeto arquitetural MVC (*Model, View e Controller*).

1.2 ORGANIZAÇÃO DO TRABALHO

Este trabalho está subdividido em 5 capítulos. No Capítulo 2 é apresentado a metodologia utilizada para o desenvolvimento deste trabalho. Já no Capítulo 3 é especificado o projeto do software desenvolvido contendo os requisitos funcionais e não funcionais do sistema, os casos de uso, o diagrama conceitual do sistema, o diagrama de classes e diagrama lógico de banco de dados, além de apresentar o conjunto de atividades, métodos, práticas e tecnologias utilizadas para o desenvolvimento do sistema proposto. No Capítulo 4 é apresentado os resultados e as discussões acerca do sistema. Por fim, no Capítulo 5 é apresentada a conclusão com as contribuições e trabalhos futuros.

2. METODOLOGIA

O presente trabalho foi realizado através de uma pesquisa bibliográfica, entrevista informal e observação direta, com o objetivo de coletar as informações principais para o desenvolvimento do sistema proposto bem como efetuar uma análise das mesmas. A entrevista foi realizada com os funcionários e proprietários da empresa New Art Color, a fim de se obter os requisitos necessários para a construção do sistema e os principais problemas organizacionais enfrentados pela empresa.

Com o intuito de utilizar técnicas que ajudarão na especificação, desenvolvimento e na validação do sistema proposto, atendendo principalmente às necessidades da empresa, foi utilizada a metodologia de desenvolvimento de software *Rational Unified Process* (RUP), promovendo uma solução disciplinada, consistindo na organização das tarefas e responsabilidades de todos envolvidos durante o desenvolvimento do sistema. A metodologia utilizada consistiu em uma estrutura de trabalho de processos com o objetivo de orientar a concepção do produto e assim projetá-lo e documentá-lo baseado na *Unified Modeling Language* (UML). A UML possibilitou a criação dos diagramas que possibilitaram um melhor entendimento do sistema e a desenvolvê-lo de forma mais funcional e completa. No desenvolvimento do projeto, foram utilizados 4 diagramas UML, sendo eles: o diagrama conceitual inicial, o diagrama de caso de uso, o diagrama de classe e o diagrama lógico de banco de dados. Depois de realizada toda a modelagem do sistema, foi criado o Banco de Dados, onde todas as informações seriam devidamente armazenadas de forma segura e íntegra. O Sistema de gerenciamento de banco de dados (SGBD) escolhido para o desenvolvimento do sistema foi o MySQL. Além da escolha do MYSQL foi definido a utilização do padrão de projeto arquitetural MVC (*Model, View e Controller*) para a codificação do sistema, este padrão de projeto teve como principal característica ajudar no desenvolvimento de um sistema seguro, com código limpo e simples.

3. DESENVOLVIMENTO DO SYSARTVENDAS

Este Capítulo tem por finalidade apresentar de forma geral as funcionalidades do sistema para o controle e gerenciamento das vendas da empresa New Art Color, bem como fundamentar todo o processo, metodologia e tecnologias utilizadas durante o desenvolvimento do sistema, definindo os conceitos concernentes ao tema proposto.

3.1. SOBRE O SISTEMA

O SysArtVendas, consiste em um sistema responsável por todo o processo de controle e gerenciamento das vendas realizadas pela empresa New Art Color, o sistema tem como objetivo prover de uma maneira fácil e rápida o gerenciamento e controle de todas as vendas realizadas dentro da empresa, possibilitando um atendimento melhor a seus clientes, garantindo que seus pedidos sejam realizados com sucesso e rapidez. O sistema contará com um cadastro e edição dos produtos fabricados pela empresa; cadastro e gerenciamento de usuários; entrada de produtos; saída de produtos (venda); cadastro e edição de clientes e emissão de relatórios.

A seguir é apresentado todo o processo de desenvolvimento do sistema, bem como a metodologia utilizada e os artefatos (documentos) gerados.

3.2. METODOLOGIA DE DESENVOLVIMENTO

Para o desenvolvimento de um software se faz necessária a criação de um conjunto de atividades, métodos, práticas e transformações que possam guiar as pessoas no processo de desenvolvimento. Para o desenvolvimento eficaz, segundo Falbo (2005), deve-se considerar claramente as relações entre as atividades, os artefatos produzidos no desenvolvimento, as ferramentas e os procedimentos necessários e a habilidade, o treinamento e a motivação do pessoal envolvido.

O desenvolvimento de um software é decomposto em diversos processos, tais como: processo de desenvolvimento, processo de garantia da qualidade, processo de gerência de projetos, processo de testes, etc. Segundo Falbo (2005), o

desenvolvimento de um software não pode ser definido de forma universal. Para ser eficaz e conduzir à construção de produtos de boa qualidade, um processo deve ser adequado às especificidades do projeto em questão. Deste modo, processos devem ser definidos caso a caso, considerando-se as características da aplicação (domínio do problema, tamanho, complexidade etc), a tecnologia a ser adotada na sua construção (paradigma de desenvolvimento, linguagem de programação, mecanismo de persistência etc), a organização onde o produto será desenvolvido e a equipe de desenvolvimento.

A seguir é apresentada a metodologia utilizada para o desenvolvimento do sistema proposto.

3.3. RUP – RATIONAL UNIFIED PROCESS

Para o desenvolvimento do sistema proposto utilizou-se a metodologia de desenvolvimento de software RUP. Segundo Piske (2003), o RUP é uma metodologia para desenvolvimento de software criada pela Rational Software, IBM, Softeam, Unisys, Nihon, Alcatel e Q-labs. O RUP também pode ser encontrado em forma de um software fornecido pela Rational Software, ou seja como um conjunto de processos.

Ainda de acordo com Piske (2003), o RUP é mais que um software que auxilia no desenvolvimento; é uma metodologia de desenvolvimento com uma estrutura formal e bem definida. Um dos principais pilares do RUP é o conceito de *best practices* (melhores práticas) que são regras/práticas que visam a reduzir os riscos existentes em qualquer software e que tornam o desenvolvimento muito mais eficiente. O RUP define seis best practices:

- **Desenvolvimento iterativo:** desenvolver iterativamente significa desenvolver em ciclos. Cada ciclo contém um objetivo que deve ser alcançado. Esta prática possibilita uma série de vantagens, como a identificação/modificação de requerimentos com mais facilidade; a integração progressiva de elementos ao software, ocasionando uma melhora no descobrimento e endereçamento de riscos;

desenvolvimento iterativo provê aos gerentes maneiras de fazer mudanças táticas aos produtos; entre outros.

- **Gerenciamento de requerimento:** provê, de maneira prática, a produção , organização e comunicação dos requisitos do projeto. O RUP é uma metodologia dirigida a casos de uso, de modo que é possível utilizar os mesmos casos de uso definidos no sistema como base para o resto do processo.
- **Utilizar arquitetura baseada em componentes:** o RUP foca o desenvolvimento na modularização, através do uso de componentes, de modo a criar um sistema flexível, adaptável, intuitivamente entendível e reutilizável.
- **Modelar visualmente:** a modelagem visual não somente permite ter um entendimento maior sobre a concepção e a complexidade do sistema, mas também sobre o seu dimensionamento, além de facilitar a identificação e solução de problemas.
- **Verificação contínua de qualidade:** o RUP não toma a qualidade como responsabilidade de apenas uma pessoa ou grupo, mas como uma responsabilidade de todos os integrantes do projeto. A qualidade é focada na qualidade do produto e na qualidade do processo.
- **Controle de mudança:** por se tratar de um processo de desenvolvimento iterativo, muitas mudanças podem ocorrer durante a execução do projeto, sendo assim controlar as mudanças durante todo o projeto é prática fundamental para manter a qualidade do projeto.

Uma das utilidades do RUP consiste em diminuir os riscos do desenvolvimento tornando-o mais eficiente através das seis melhores práticas a serem executadas durante todo o processo.

Segundo DevMedia (2007) o RUP passa por 4 fases básicas. Estas fases são:

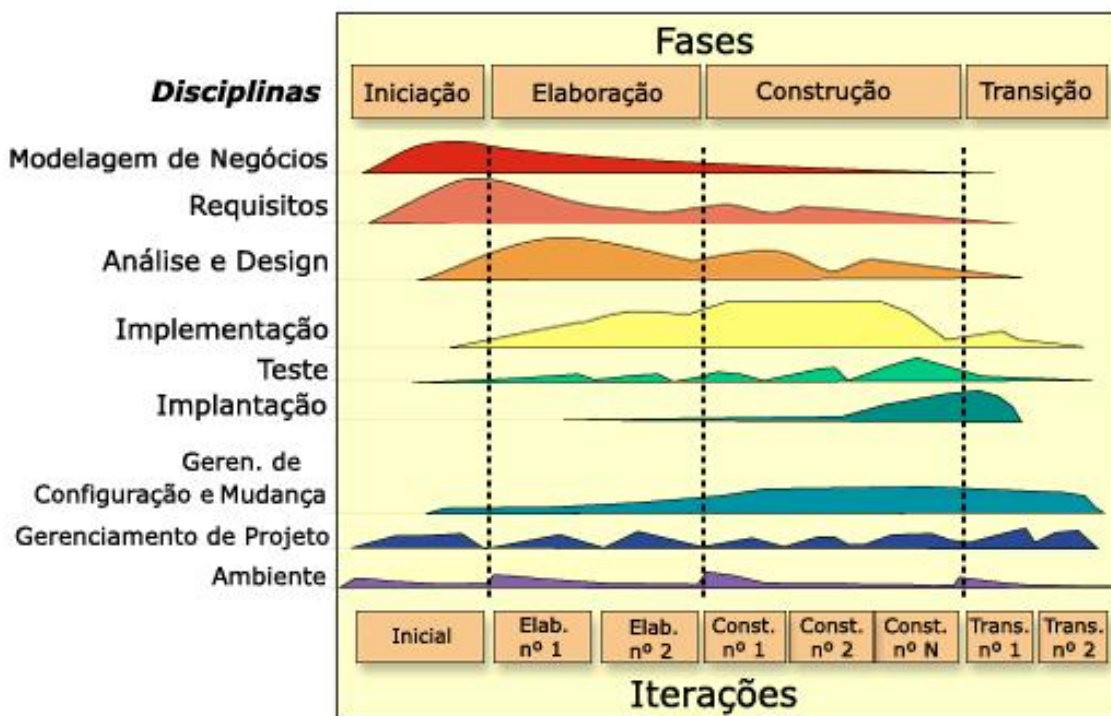
- **Inception (concepção):** entendimento da necessidade e visão do projeto,

- **Elaboration (elaboração):** especificação e abordagem dos pontos de maior risco,
- **Construction (construção):** desenvolvimento principal do sistema,
- **Transition (transição):** ajustes, implantação e transferência de propriedade do sistema.

Cada uma das fases é composta de uma ou mais iterações, o que se assemelha a um modelo em espiral. Estas iterações são em geral curtas (1-2 semanas) e abordam algumas poucas funções do sistema, reduzindo o impacto das mudanças.

Além das fases e iterações, existem os *workflows*. Cada *workflow* é uma sequência de tarefas encadeadas e relacionadas a um aspecto importante do projeto. Os gráficos da Figura 1 apresentam a ênfase de cada *workflow* em cada etapa do projeto.

Figura 1. Ciclo de vida do Processo Unificado Rational.



Fonte: The Rational Unified Process (IBM, 2006)

3.4. FASES REALIZADAS DURANTE O DESENVOLVIMENTO DO SISTEMA

3.4.1. Concepção

Nesta fase foi realizado uma discussão sobre o problema, definição do escopo do projeto, estimativa de recursos necessários para a execução do projeto e a coleta e especificação dos principais requisitos do sistema.

3.4.1.1. Coleta e especificação dos requisitos

Os requisitos de um sistema podem ser considerados como a descrição de um recurso e funcionalidade de um software, os requisitos também transmitem as expectativas do usuário de acordo com suas necessidades.

A especificação é a descrição sistemática e abstrata do que o software deve fazer a partir daquilo que foi analisado anteriormente. Ela apresenta a solução de como os problemas levantados na análise devem ser resolvidos pelo software em desenvolvimento (TONSIG, 2021). A especificação é a forma de comunicação direta entre o analista e a equipe de desenvolvimento do software. Os requisitos de software são, além de funções, objetivos, propriedades e restrições que o sistema deve possuir para satisfazer contratos, padrões ou especificações de acordo com o(s) usuário(s).

Os requisitos são separados em dois núcleos ou seja requisitos funcionais e não funcionais, onde requisitos funcionais são aqueles que mostram as funcionalidades que o sistema deve possuir, e os requisitos não funcionais descrevem o que os requisitos funcionais devem realizar dentro do sistema. O Quadro 1 apresenta os requisitos funcionais e não funcionais do sistema SysArtVendas

QUADRO 1. Requisitos Funcionais e Não Funcionais do sistema SysArtVendas

Código	Requisito Funcional
USUÁRIOS/CADASTROS	
RF001	O sistema irá cadastrar funcionários (usuários).

RF002	O sistema permitirá que todos os usuários façam alterações em seus cadastros.
RF003	O sistema permitirá a exclusão usuários
RF004	O sistema permitirá a pesquisa de usuários cadastrados, pelo nome ou código.
RF005	O cadastro do usuário deverá conter: código, nome, cpf e senha.
RF006	O sistema deverá restringir o acesso para determinados tipos de funcionários, determinando quais funcionalidades eles poderão acessar.
USUÁRIOS/LOGIN	
RF007	O sistema permitirá o acesso dos usuários através do login (cpf) e senha.
PRODUTOS/CADASTROS	
RF008	O sistema permitirá que produtos sejam cadastrados
RF009	O sistema permitirá a alterações dos dados dos produtos cadastrados
RF0010	O sistema permitirá a exclusão de produtos
RF0011	O sistema permitirá a pesquisa de produtos cadastrados pelo nome e código.
RF0012	O cadastro do produto deverá conter código, nome, quantidade, valor, cor, descrição, tipo de produto, código de barras e compras.
EMPRESA/CADASTRO	
RF0013	O sistema permitirá que seja salvo os dados da empresa.
RF0014	O sistema permitirá que sejam alterados os dados da empresa.
RF0015	Os dados da empresa deverão conter código, nome, cnpj, razão social, inscrição estadual, telefone, rua, número, bairro, cep.
TIPO PRODUTO/CADASTROS	
RF0016	O sistema permitirá cadastrar os tipos de produto pelo nome ou código.
RF0017	O sistema permitirá que os tipos de produtos cadastrados sejam alterados.
RF0018	Os tipos de produtos deverão conter o código e nome.

CLIENTES/CADASTROS	
RF0019	O sistema permitirá que sejam cadastrados clientes.
RF0020	O sistema permitirá que sejam alterados os dados dos clientes.
RF0021	O sistema permitirá pesquisar clientes pelo código e nome.
RF0022	O sistema permitirá a exclusão de clientes cadastrados.
RF0023	Os clientes cadastrados devem conter código, nome, cpf, rg, data de nascimento, telefone, e-mail, cnpj e inscrição estadual caso possua.
PRODUTOS/ENTRADA	
RF0024	O sistema deverá permitir que seja cadastrada a entrada de produtos.
RF0025	O sistema permitirá que seja alterada a entrada de produtos.
RF0026	o sistema permitirá que seja excluída a entrada de produtos.
RF0027	A entrada de produtos deverá conter código da entrada, código do produto, quantidade, data, valor, código fornecedor, código do usuário.
PRODUTOS/VENDA	
RF0028	O sistema permitirá registrar a saída de produto.
RF0029	O sistema permitirá alterar a saída de produtos.
RF0030	O sistema permitirá que seja excluída a saída de produtos.
RF0031	Na saída de produto deverá conter código, data, valor, desconto, tipo de pagamento, código usuário e código do cliente.
REQUISITOS GERAIS DO SISTEMA	
RF0032	O sistema deverá permitir o cancelamento das operações.
Código	Requisito não Funcional
RNF001	O sistema deve ser desenvolvido em linguagem Java usando o padrão de projeto MVC (Model, View e Controller).
RNF002	O banco de dados deve ser desenvolvido no MySQL.
RNF003	A interface deve ser agradável e de fácil utilização.
RNF004	O sistema deve consumir poucos recursos do computador.
RNF005	O sistema deve ser ágil e rápido para o melhor desempenho.

RNF006	O sistema deverá corresponder às necessidades da empresa.
--------	---

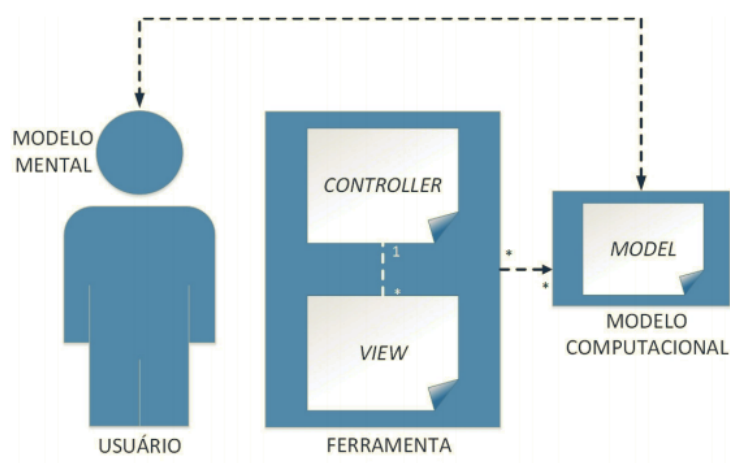
Fonte: Elaborado pelo autor

3.4.2. Elaboração

Para um melhor entendimento do domínio do problema, bem como estabelecer um *framework* para arquitetura do sistema, desenvolvendo um plano de projeto e identificando seus principais riscos, foi utilizado o padrão de projeto arquitetural MVC (*Model View Controller*) juntamente com o padrão de projeto *Data Access Object* (DAO).

Segundo Lemos, *et al.* (2013) o MVC foi criado e introduzido pela primeira vez na linguagem de programação Smalltalk 76 em 1978, por Reenskaug (1978), enquanto ele era um cientista visitante no Palo Alto Research Laboratory Xerox (PARC), como uma solução para o problema geral de dar aos usuários controle sobre suas informações. Conforme cita Reenskaug (1978), a principal característica do MVC é facilitar para os usuários o emprego de uma aplicação, pois existe um vácuo entre o modelo mental e o modelo digital que existe no computador conforme ilustrado na Figura 2. O MVC apresenta para o usuário a visualização e a manipulação dessas informações de maneira evidente. Assim, o usuário poderá ver o mesmo elemento do modelo em diferentes situações.

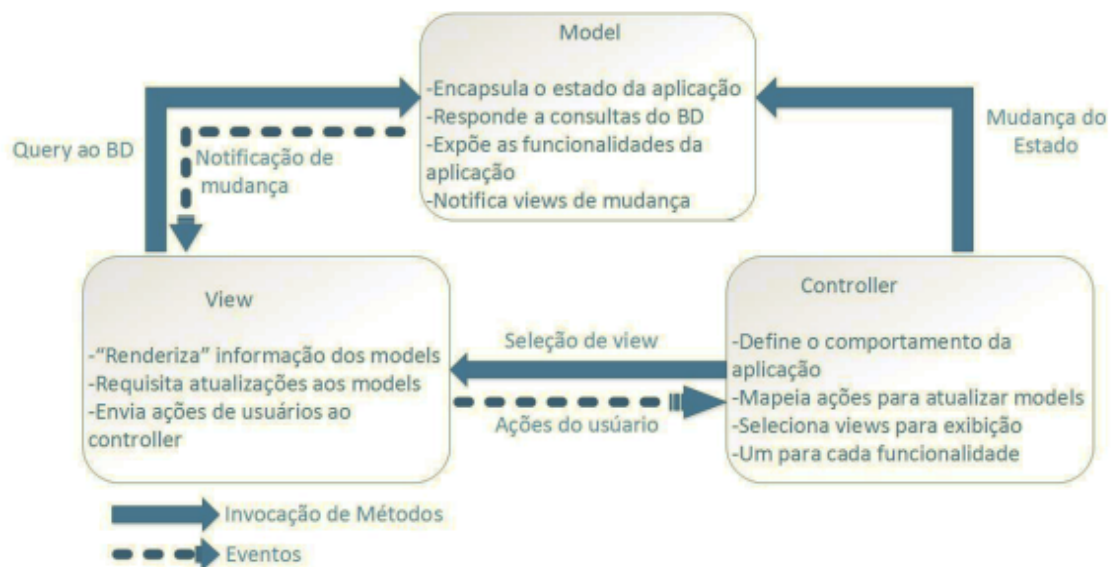
Figura 2. Modelo mental MVC.



Fonte: Lemos, *et al.* (2013)

A arquitetura MVC auxilia os desenvolvedores a construir aplicações separando seus principais componentes, a manipulação e armazenamento dos dados, as funções que irão trabalhar com as entradas dos dados e a visualização do usuário. A arquitetura MVC especifica onde cada tipo de lógica deve estar localizada na aplicação (SANTOS et al., 2010). O componente *Model* é o objeto de aplicação, a *View* é a interface visualizada pelo usuário e o *Controller* trabalha em relação às entradas de uma *View* e como as mesmas reagiram (GAMMA et al, 2000). A Figura 3 representa algumas funções e a comunicação de cada componente.

Figura 3. Modelo MVC.



Fonte: Lemos, et al. (2013) adaptado de Zemel (2009).

Segundo Sardagna e Vahldick (2008) o padrão DAO consiste em abstrair o mecanismo de persistência utilizado na aplicação. A camada de negócios acessa os dados persistidos sem ter conhecimento se os dados estão em um banco de dados relacional ou um arquivo XML. A Figura 4 demonstra a utilização do padrão de projeto MVC e DAO no sistema proposto.

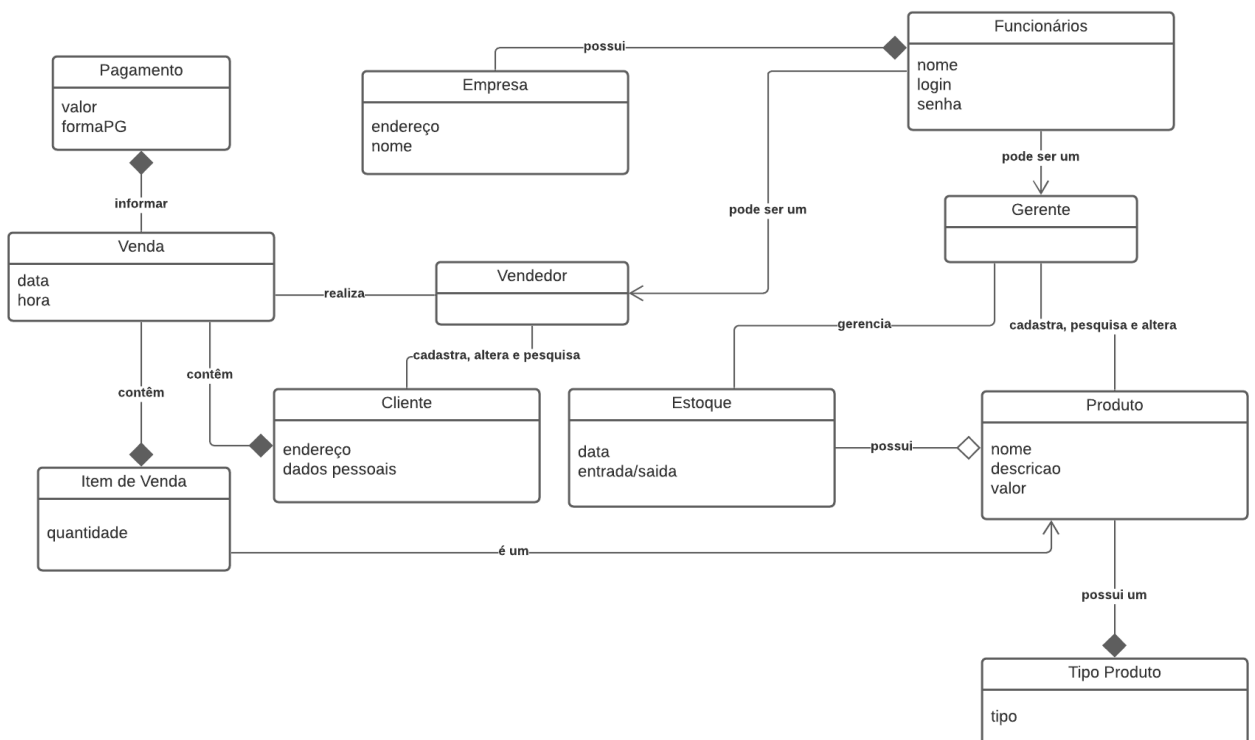
Figura 4. Arquitetura do sistema SysArtVendas



Fonte: Elaborado pelo autor

Após a definição da arquitetura a ser utilizada foi construído o modelo conceitual inicial do sistema (Figura 5), conforme descrição apresentada no item 3.1, bem como identificado os principais riscos que poderiam ocorrer durante a execução do projeto, como pode ser observado no Quadro 2.

Figura 5. Modelo Conceitual inicial do sistema.



Fonte: Elaborado pelo autor.

QUADRO 2. Riscos identificados.

Risco	Natureza	Impacto
Usuários finais resistentes ao sistema	Risco de negócio	moderado
Pressão para reduzir prazos de entrega	Risco de negócio	crítico
Usuários irão modificar os requisitos	Risco de projeto	crítico
Tecnologia não atingirá as expectativas	Risco de projeto	catastrófico
Menos reuso que o planejado	Risco técnico	crítico
Falta de treinamento no uso das ferramentas	Risco de negócio	moderado
Cronograma não realista.	Risco de negócio	crítico
Desenvolvimento das funções erradas.	Risco técnico	catastrófico
Desenvolvimento da interface com o usuário errada.	Risco técnico	catastrófico

Fonte: Elaborado pelo autor.

3.4.3. Construção

Esta fase compreendeu a fase de modelagem e a fase de desenvolvimento em si, aquela em que o sistema foi efetivamente programado.

3.4.3.1. Projeto e Análise

O processo de desenvolvimento de software compreende um conjunto de atividades que engloba métodos, ferramentas e procedimentos, com o objetivo de produzir sistemas que atendam às necessidades e requisitos especificados pelos usuários.

Entre as várias etapas existentes no processo de desenvolvimento de um sistema se encontra a etapa de projeto e análise dos requisitos. Segundo Geraldo Xexéo (2007) é a partir da análise que um sistema é desenvolvido. Ela é, simultaneamente, um acordo entre os desenvolvedores e seus clientes e um mecanismo de comunicação entre os desenvolvedores. A análise define que

serviços devem ser fornecidos pelo sistema a ser implementado e, por consequência, que serviços não estão no escopo do sistema.

Segundo Pressman (1997), todos os métodos de análise devem ser capazes de suportar cinco atividades:

- Representar e entender o domínio da informação;
- Definir as funções que o software deve executar;
- Representar o comportamento do software em função dos eventos externos;
- Separar os modelos de informação, função e comportamento de maneira a apresentar os detalhes de forma hierárquica, e,
- Prover a informação essencial em direção à determinação dos detalhes de implementação.

Com base nesta definição, foi necessário desenvolver uma representação dos requisitos coletados, para tal foi utilizado a Linguagem Unificada de Modelagem (UML).

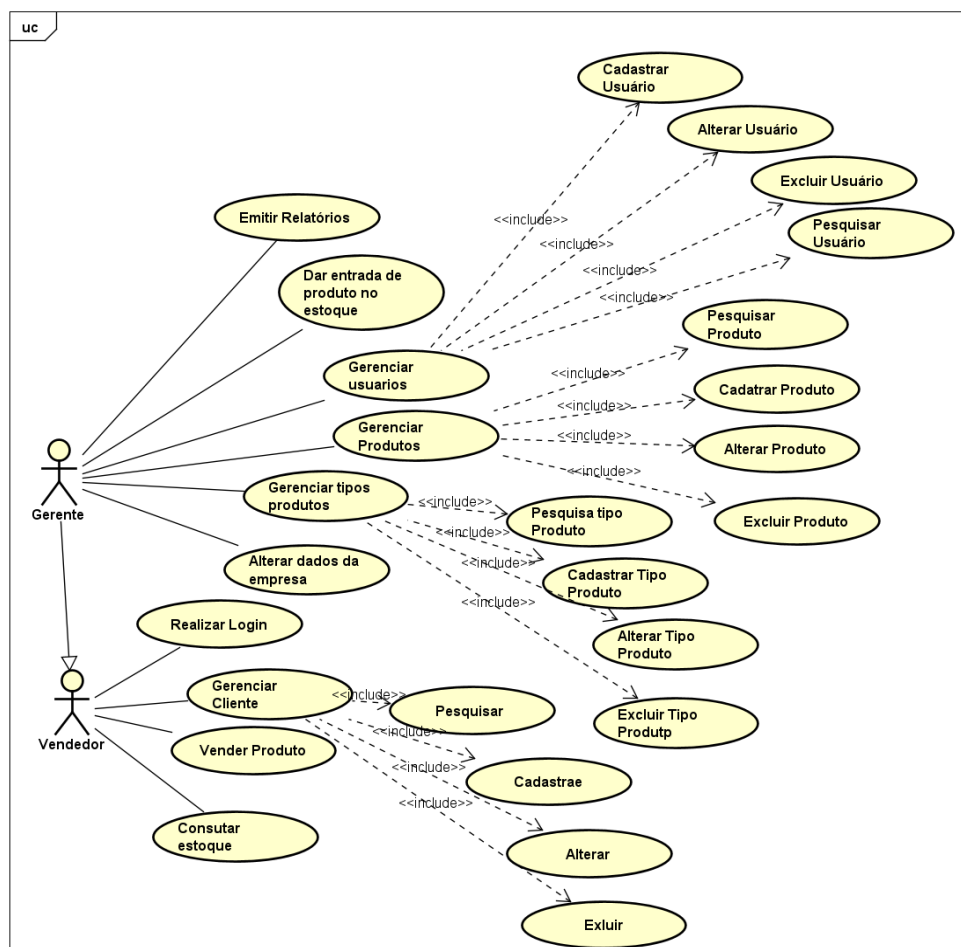
A UML é uma linguagem gráfica utilizada para visualização, especificação, construção e documentação de um projeto de software. Segundo Booch, Rumbaugh e Jacobson (2006), a UML nos proporciona de uma forma padrão para a preparação de planos para uma arquitetura de projetos de sistemas. Conforme apontado por Salmon (2017) a UML oferece um conjunto de notações e diagramas para representar os requisitos licitados observando diferentes aspectos (onde predomina o funcional) sem que haja uma forte interferência entre eles.

Segundo Guedes (2018), a UML é uma linguagem utilizada para modelar software baseado no paradigma de orientação a objeto, aplicada principalmente durante as fases análise de requisitos e projetos de software. Neste projeto, buscou-se utilizar o UML para a análise e projeto dos requisitos especificados no item 3.4.1.1, mais especificamente, foram utilizados para o projeto do sistema os seguintes diagramas: modelo conceitual do sistema, casos de usos, classes e diagrama lógico de banco de dados.

O diagrama conceitual do sistema é um artefato do domínio do problema e não do domínio da solução, é utilizado para especificação da arquitetura do sistema. Conforme apresentado na Figura 5, o diagrama é formado pelos conceitos obtidos a partir da análise textual da definição do problema, atributos e associações.

Os diagramas de casos de usos nos oferecem uma visão de todo o sistema de uma forma geral. Registrando em forma de figuras as funcionalidade do sistema a serem executadas mostrando uma visão ampla para os usuários. A Figura 6 representa o diagrama de caso de uso do sistema proposto. É importante observar que o diagrama de casos de usos proposto não descreve os comportamentos e processos do sistema, apenas apresenta as associações entre os casos de usos e os autores envolvidos.

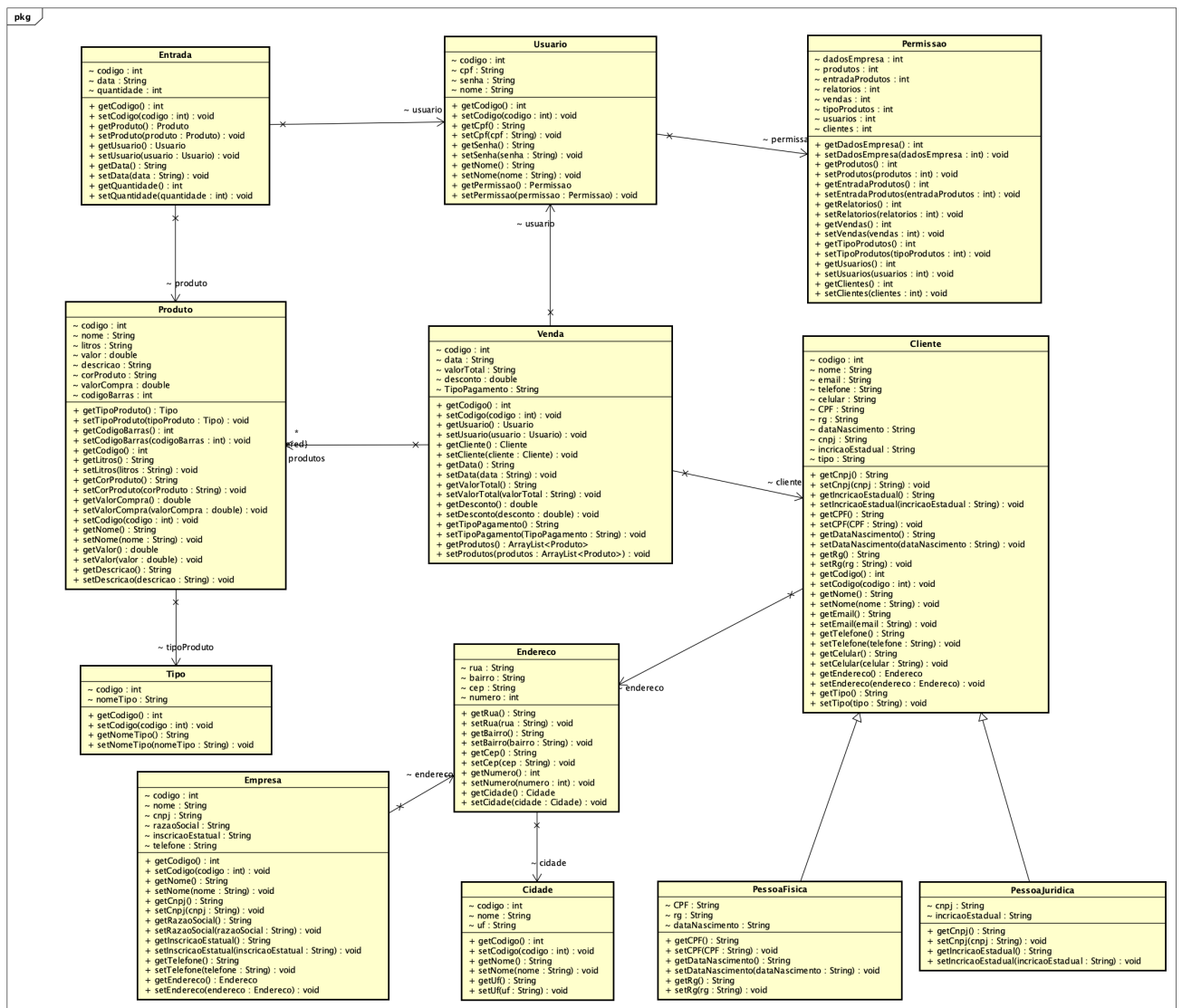
Figura 6. Diagrama de Casos de Usos do sistema.



Fonte: Elaborado pelo autor.

Para representar a parte estática do sistema bem como a estrutura das classes existentes, interfaces e seus relacionamentos propomos a utilização do diagrama de classes, conforme apresentado nas Figuras 7, 8 e 9. Nestes diagramas foi possível representar alguns dos classificadores do sistema, componentes, os diferentes relacionamentos entre estes, seus atributos, operações e restrições.

Figura 7. Diagrama de Classes - Representação da Camada MODEL.



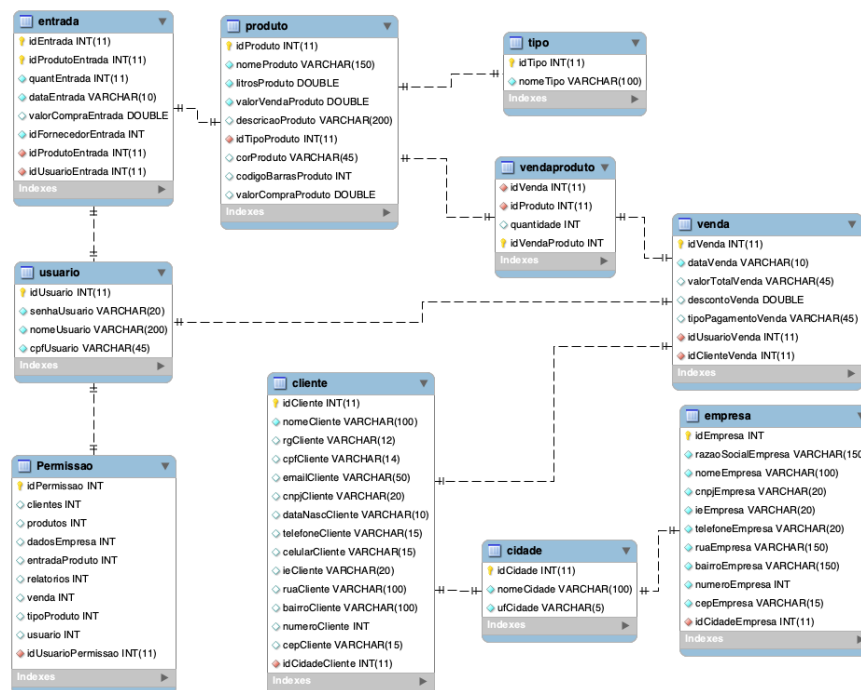
Fonte: Elaborado pelo autor.

armazenados nos sistemas. Esta técnica é realizada nas atividades desenvolvidas pelos analistas de sistemas e programadores.

Já para Elmasri e Navathe (1994), a modelagem de dados é um conjunto de conceitos que se usa para descrever o Banco de Dados (BD) e certas requisições que o banco deverá possuir. Conforme Miranda (2005), a estrutura por modelo de dados permite ao usuário recuperar dados de forma rápida e eficiente.

Para realizar a modelagem de dados vários elementos precisam ser levados em consideração além do conhecimento técnico. Um destes elementos é o conhecimento sobre a realidade que o banco será construído buscando solucionar alguns tipos de problemas. A Figura 10 apresenta o Diagrama Lógico do Banco de dados do sistema proposto, nela podemos observar as tabelas e os relacionamentos entre elas, as limitações do banco de dados, a adequação de padrão e nomenclatura, as chaves primárias e estrangeiras, a normalização e integridade referencial entre as tabelas.

Figura 10. Diagrama Lógico de Banco de Dados do Sistema.



Fonte: Elaborado pelo autor.

3.4.3.2. Codificação

Para a implementação do sistema, foram pesquisadas várias linguagens de programação, através de diversos critérios, tendo sido escolhida a Linguagem de Programação Java integrada ao ambiente de desenvolvimento Netbeans para o desenvolvimento do sistema proposto. A linguagem Java foi escolhida devido ao fato de ser confiável, pela facilidade em aprendê-la, por ser uma linguagem multiplataforma, pela facilidade com que o programador enxerga seu código-fonte, pela facilidade em reaproveitar o código-fonte sendo desnecessário sua re-escrita, bem como por ser uma linguagem orientada à objetos e *multithreading*. A escolha de se utilizar o Netbeans se deu devido à oportunidade de usar uma ferramenta capaz de aumentar a produtividade no desenvolvimento de aplicações. A Figura 11 apresenta um fragmento do código desenvolvido na camada DAO, nela podemos observar a inserção de novos produtos no banco de dados, bem como a entrada de produtos no banco de dados, demais fragmentos dos códigos desenvolvidos são apresentados no no Apêndice A.

Figura 11. Exemplo do código implementado - Camada DAO.

```
public class ProdutoDao extends DaoConexao {

    public void cadastrar(Produto prod) {
        String sql = "INSERT INTO produto (nomeProduto, litrosProduto, valorVendaProduto, descricaoProduto, "
            + "idTipoProduto, corProduto, codigoBarrasProduto, valorCompraProduto) "
            + "VALUES ('"+prod.getNome()+"', '"+prod.getLitros()+"', '"+prod.getValor()+"', "
            + "'"+prod.getDescricao()+"', '"+prod.getTipoProduto().getCodigo()+"', '"+prod.getCorProduto()+"', "
            + "'"+prod.getCodigoBarras()+"', '"+prod.getValorCompra()+"');";
        conectar();
        try {
            comando.executeUpdate(sql);
            JOptionPane.showMessageDialog(null, "Produto cadastrada com sucesso.", "Mensagem do Sistema",
                JOptionPane.INFORMATION_MESSAGE);
        } catch (SQLException ex) {
            JOptionPane.showMessageDialog(null, "Erro ao cadastrar a Produto. \n Erro:" + ex.getMessage(),
                "Mensagem do Sistema", JOptionPane.ERROR_MESSAGE);
        }
        fechar();
    }

    public void entrada(Entrada entrada) {
        String sql = "INSERT INTO entrada (idProdutoEntrada, quantEntrada, dataEntrada, idUsuarioEntrada) "
            + "VALUES ('"+entrada.getProduto().getCodigo()+"', '"+entrada.getQuantidade()+"', "
            + "'"+entrada.getData()+"', '"+entrada.getUsuario().getCodigo()+"');";
        conectar();
        try {
            comando.executeUpdate(sql);
            JOptionPane.showMessageDialog(null, "Entrada do produto realizada com sucesso.",
                "Mensagem do Sistema", JOptionPane.INFORMATION_MESSAGE);
        } catch (SQLException ex) {
            JOptionPane.showMessageDialog(null, "Erro ao dar entrada no Produto. \n Erro:" + ex.getMessage(),
                "Mensagem do Sistema", JOptionPane.ERROR_MESSAGE);
        }
        fechar();
    }
}
```

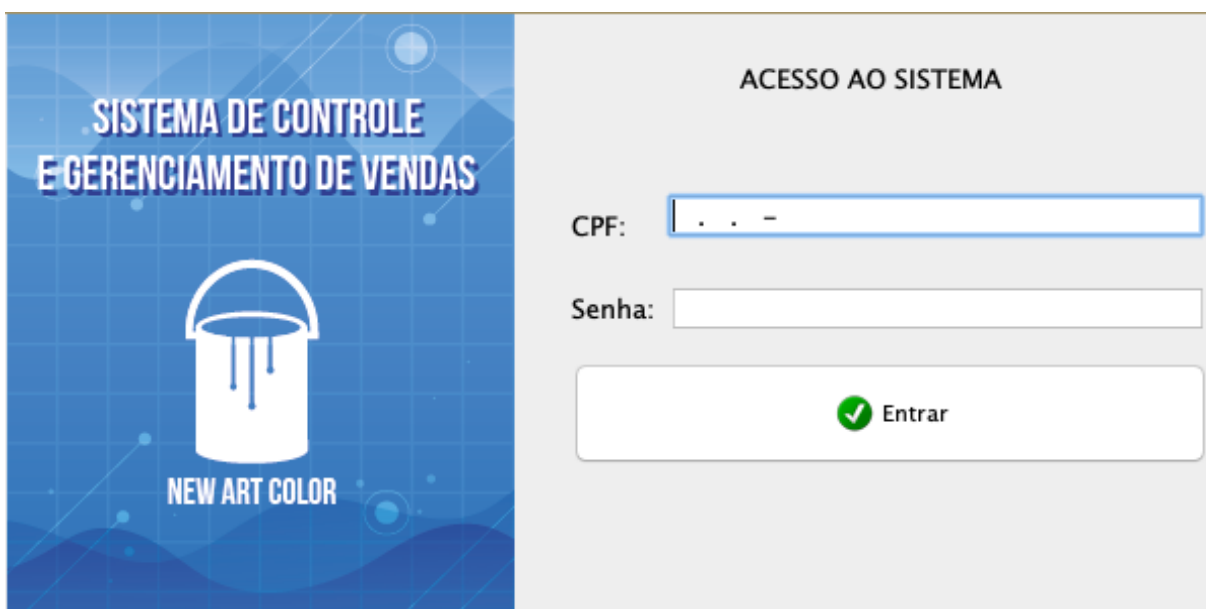
FONTE: Elaborado pelo autor.

4. RESULTADOS E DISCUSSÕES

O SysArtVendas , tem como objetivo prover de uma maneira fácil e rápida o gerenciamento e controle de todas as vendas realizadas dentro da empresa, possibilitando um atendimento melhor a seus clientes, garantindo que seus pedidos sejam realizados com sucesso e rapidez.

A Figura 12, apresenta a Tela de Login do sistema, onde é solicitado o Cadastro de Pessoa Física (CPF) e senha do usuário, direcionando para a Tela Inicial do sistema.

Figura 12. Tela de Login do sistema.

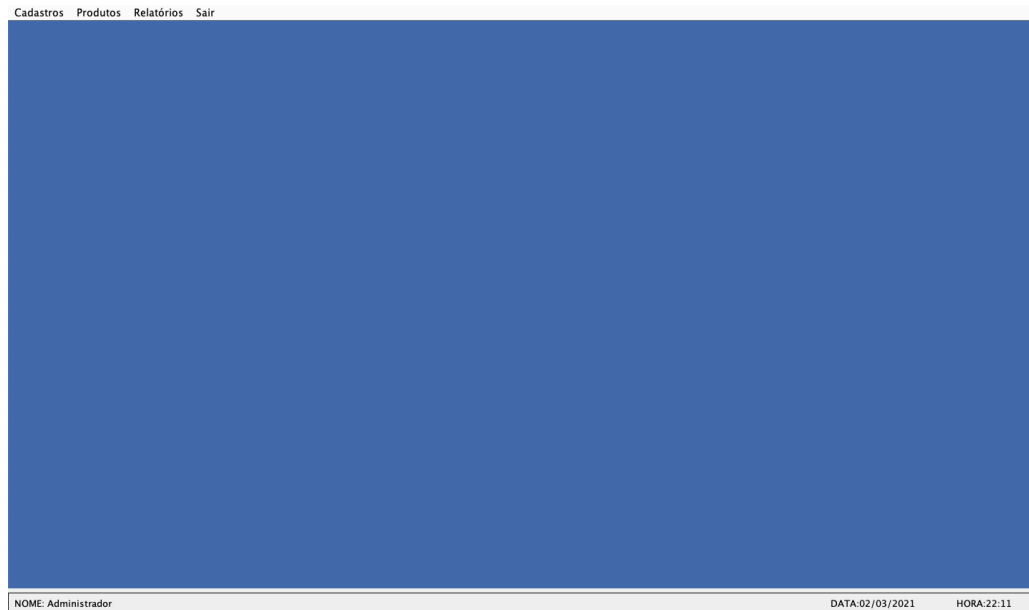


A imagem mostra a interface de login do sistema. À esquerda, há um banner azul com o texto "SISTEMA DE CONTROLE E GERENCIAMENTO DE VENDAS" e "NEW ART COLOR" abaixo de um ícone de uma lata de tinta. À direita, o formulário de login tem o título "ACESSO AO SISTEMA". Ele contém dois campos de entrada: "CPF:" com um formato de máscara ". . -" e "Senha:". Abaixo dos campos, há um botão "Entrar" com um ícone de checkmark verde.

Fonte: Elaborado pelo autor.

A Figura 13 apresenta a tela inicial do programa onde contém o menu contendo todas as funcionalidades existentes no sistema.

Figura 13. Tela de inicial do sistema.



Fonte: Elaborado pelo autor.

A Figura 14 apresenta a tela de gerenciamento de clientes onde o funcionário poderá fazer o cadastro de todos os clientes da empresa, bem como pesquisar por um determinado cliente, alterar os dados e excluir os dados de um cliente.

Figura 14. Tela gerenciamento de clientes.

Fonte: Elaborado pelo autor.

A Figura 15 apresenta a tela de gerenciamento de produtos onde o gerente poderá cadastrar todos os produtos da empresa, bem como pesquisar, excluir e alterar os dados de um determinado produto.

Figura 15. Tela gerenciamento de produtos.

Fonte: Elaborado pelo autor.

A Figura 16 apresenta a tela de cadastro dos dados da empresa, nesta tela o gerente poderá informar os dados principais da empresa, tais como razão social, cadastro nacional da pessoa jurídica (CNPJ), endereço, entre outros.

Figura 16. Tela dados da empresa.

Fonte: Elaborado pelo autor.

A Figura 17 apresenta a tela de gerenciamento dos tipos de produtos onde o gerente poderá cadastrar, alterar, excluir e pesquisar os tipos de produtos que a empresa possui.

Figura 17. Tela gerenciamento dos tipos de produtos.

The screenshot shows a web interface for managing product types. At the top, there is a search bar labeled 'Tipo de produto:' with a placeholder text 'Digite o nome ou o código do tipo de produto' and a 'PESQUISAR' button. Below this, there are two input fields: 'Código:' and 'Tipo:'. At the bottom, there is a row of five buttons: 'NOVO' (with a document icon), 'CADASTRAR' (with a floppy disk icon), 'ALTERAR' (with a pencil icon), 'EXCLUIR' (with a trash can icon), and 'CANCELAR' (with an 'X' icon).

Fonte: :Elaborado pelo autor.

A Figura 18 apresenta a tela de entrada de produtos no estoque onde o gerente poderá dar entrada de produtos no estoque.

Figura 18. Tela entrada de produtos.

The screenshot shows a web interface for entering products into the stock. At the top, there is a search bar labeled 'Tipo de produto:' with a placeholder text 'Digite o nome ou o código do produto' and a 'PESQUISAR' button. Below this, there is a section titled 'Dados do Produto' with three input fields: 'Código:', 'Produto:', and 'Qtd atual no estoque:'. Below that, there is a section titled 'Entrada do Produto' with two input fields: 'Data da entrada:' and 'Quantidade:'. At the bottom, there are two buttons: 'DAR ENTRADA' (with a floppy disk icon) and 'CANCELAR' (with an 'X' icon).

Fonte:: Elaborado pelo autor..

A Figura 19 apresenta a tela de gerenciamento de usuários onde o gerente poderá cadastrar, pesquisar, alterar e excluir os dados dos funcionários da empresa que terão acesso ao sistema. Nesta tela o gerente poderá também definir

as permissões que cada usuário terá no sistema, conforme apresentado na Figura 20.

Figura 19. Tela gerenciamento de usuários.

The screenshot shows a user management interface. On the left is a sidebar with five buttons: 'NOVO' (with a document icon), 'CADASTRAR' (with a floppy disk icon), 'ALTERAR' (with a pencil icon), 'EXCLUIR' (with a trash can icon), and 'CANCELAR' (with a red 'X' icon). The main content area has a search bar at the top with the text 'Pesquisar Usuário' and a placeholder 'Digite o nome ou o código do Usuário', followed by a 'PESQUISAR' button with a magnifying glass icon. Below the search bar are two tabs: 'Dados do Usuário' (active, highlighted in blue) and 'Permissões do Sistema'. The 'Dados do Usuário' tab contains a form with the following fields: 'Código:' with a text input, 'Nome:' with a text input, 'Cpf:' with a masked text input (format: . . -), and 'Senha:' with a text input.

Fonte: Elaborado pelo autor.

Figura 20. Tela gerenciamento de cadastro de usuários - Definição das Permissões do Sistema.

The screenshot shows the same user management interface as Figure 19, but with the 'Permissões do Sistema' tab selected. The sidebar remains the same. The search bar is also present. The 'Permissões do Sistema' tab contains a list of permissions, each with an unchecked checkbox: 'Marcar Todos', 'Clientes', 'Produtos', 'Dados Empresa', 'Entrada de Produto', 'Relatórios', 'Venda de Produto', 'Tipo Produto', and 'Usuário'.

Fonte: Elaborado pelo autor.

A Figuras 21 e 22 apresentam a tela de vendas de produtos onde poderá ser realizadas todas as vendas realizadas pela empresa, nesta o usuário deverá

selecionar o cliente que está realizando a comprar, os itens (produtos), definir a forma de pagamento e aplicar o desconto, quando necessário .

Figura 21: Tela de vendas de produtos - Selecionar o cliente.

Dados da Venda

Data: Horário:

CLIENTE ITENS DA VENDA

Pesquisar:

Código:

Cliente:

RG: CPF:

CNPJ: Inscrição Estadual:

Rua: Bairro:

Numero: UF: Cidade:

Cep:

Sub Total (R\$): Desconto (%): Total (R\$):

Forma de Pagamento: Qtde Vezes:

Menu de Opções

Fonte: Elaborado pelo autor.

Figura 22: Tela de vendas de produtos - Selecionar os produtos.

Dados da Venda

Data: Horário:

CLIENTE ITENS DA VENDA

Pesquisar:

Código: Produto:

Tipo de Produto Cor:

Quantidade:

Código	Produto	Valor Unitário	Quantidade	Valor Total
--------	---------	----------------	------------	-------------

Sub Total (R\$): Desconto (%): Total (R\$):

Forma de Pagamento: Qtde Vezes:

Menu de Opções

Fonte:: Elaborado pelo autor.

5. CONCLUSÃO E TRABALHOS FUTUROS

Em vista dos resultados obtidos pode-se dizer que o sistema trará uma grande melhoria para o desenvolvimento da empresa proporcionando uma maior facilidade para o controle dos produtos que estão armazenados no estoque e sua respectiva venda, podendo minimizar os custos, oferecendo um ambiente intuitivo e simples para o usuário, permitindo uma melhor organização das informações da empresa.

A utilização da linguagem de programação Java mostrou-se bastante assertiva, por a mesma ser uma linguagem de programação orientada a objeto, de fácil entendimento, bem como ajudar a manter o sistema de forma modulável, flexível e extensível. Juntamente com a utilização da linguagem de programação Java podemos destacar a utilização dos padrões de projeto MVC e DAO trazendo diversos benefícios, tais como, o aumento na produtividade durante o desenvolvimento, a uniformidade na estrutura do sistema e a separação das classes de CRUD (Create, Retrieve, Update e Delete) das regras de negócios. Além disso, destaca-se a utilização do processo unificado de desenvolvimento de software RUP, onde obteve-se um maior controle sobre o desenvolvimento dentro dos prazos e níveis de qualidade desejados, tornando todo o processo iterativo e incremental, guiado pelos diagramas UML desenvolvidos, o que trouxe um maior entendimento dos requisitos do sistema.

Como trabalhos futuros serão realizadas a implementação dos relatórios de gestão, emissão de nota fiscal, bem como a realização das etapas de testes e implantação do sistema. Acredita-se que a implementação dos relatórios e a realização das etapas citadas permitirá o alcance de um produto de qualidade a ser implantado no cliente. Nesse estágio haverá necessidade de capacitação dos usuários finais da ferramenta. Esse *feedback* será valioso para realizar correções e acrescentar novas características, se necessário.

REFERÊNCIAS

BOOCH, Grady; RUMBAUGH, James; JACOBSON, Ivar. **UML: guia do usuário**. Elsevier Brasil, 2006.

DEVMEDIA. **RUP - Rational Unified Process**. Disponível em: <https://www.devmedia.com.br/rup-rational-unified-process/4574> Acesso em: 09 mar. 2021.

ELMASRI, R; NAVATHE, S, B. **Fundamentals of database system**. 7. ed. The Benjamim/Cummins, 2016. Disponível em: <https://www.auhd.site/upfiles/elibrary/Azal2020-01-22-12-28-11-76901.pdf> Acesso em: 25 fev. 2021.

FALBO, R. A. **Engenharia de software - Notas de Aula**. Departamento de Informática, Universidade Federal do Espírito Santo, 2014. Disponível em: http://www.inf.ufes.br/~jssalomon/wp-content/uploads/disciplinas/engsoft/Notas_Aula_Engenharia_Software_Falbo_2014.pdf Acesso em: 25 fev. 2021.

GAMMA, Erich *et al.* **Padrões de projeto: soluções reutilizáveis de software orientado a objetos**. Porto Alegre: Bookman, 2000.

GUEDES, G. T. **UML 2: uma abordagem prática**. [S.l.]: Novatec Editora, 2018.

LEMONS, Maxmilian Ferreira de; OLIVEIRA, Patrícia Carvalho de; RUELA, Leandro César; SANTOS, Matheus da Silva; SILVEIRA, Thallis Carvalho; REIS, José Cláudio de Sousa. Aplicabilidade da arquitetura mvc em uma aplicação web (webapps). **Revista Eletrônica Científica de Ciência da Computação**, [S.l.], 2013. Disponível em: <http://revistas.unifenas.br/index.php/RE3C/article/view/54/13> Acesso em: 25 fev. 2021.

MIRANDA, M.L. C. de. **Organização e representação do conhecimento: fundamentos teóricos-metodológicos na busca e recuperação de informação em ambientes virtuais**. 2005. 353 f. Tese (Doutorado em Ciências da Informação) - Universidade Federal do Rio de Janeiro, Rio de Janeiro 2005.

O'BRIEN, J. A.; MARAKAS, G. M. **Administração de sistemas de informação**. 15. ed. Porto Alegre: AMGH/McGraw-Hill/Bookman, 2013.

PISKE, O. T. RUP-Rational Unified Process. Disponível em: http://www.angusyong.org/arquivos/artigos/trabalho_rup.pdf. Acesso em: 25 fev. 2021.

PORTER, M. E. **Vantagem competitiva: criando e sustentando um desempenho superior**. 1 ed. Rio de Janeiro: GEN Atlas, 1989.

PRESSMAN, R. S. **Software Engineering: a practitioner's approach**. 4. ed. New York: McGraw-Hill, Inc., 1997.

REENSKAUG, Trygve M. H. MVC XEROX PARC. Disponível em: <https://folk.universitetetioslo.no/trygver/themes/mvc/mvc-index.html>. Acesso em: 25 fev. 2021.

SALMON, Arianna Zoila Oliveira. **Modelagem e análise de requisitos de sistemas automatizados usando uml e rede de petri**. São Paulo, 2017. Tese (Doutorado) - Escola Politécnica da Universidade de São Paulo, Departamento de Engenharia Mecatrônica e de Sistemas Mecânicos, São Paulo, 2017.

SANTOS, Isaias. **Possibilidades e limitações da arquitetura mvc (model – view – controller) com ferramenta ide (integrated development environment)**. 2010. 56 f. Trabalho de Conclusão de Curso (Graduação em Ciências da Computação) - Universidade José do Rosário Vellano, Alfenas-MG, 2010.

SARDAGNA, Marcelo; VAHLICK, Adilson. **Aplicação do padrão data access object (dao) em projetos desenvolvidos com delphi departamento de sistemas e computação**. Blumenau, SC: Universidade Regional de Blumenau - FURB, 2008.

TONSIG, Sérgio Luiz. **Engenharia de Software: análise e projeto de sistemas**. Disponível em: <http://www.linhadecodigo.com.br/artigo/3224/analise-e-especificacoes-de-requisitos.aspx#ixzz6pEAdKsC6> Acesso em: 25 fev. 2021.

XEXÉO, Geraldo. **Modelagem de Sistemas de Informação: da análise de requisitos ao modelo de interface**. Creative Commons

APÊNDICE A - Exemplo fragmentos dos códigos desenvolvidos

Figura 23: Trecho do código implementado na camada MODEL.

```
public class Produto {  
  
    int codigo;  
    String nome;  
    String litros;  
    String valor;  
    String descricao;  
    String corProduto;  
    Tipo tipoProduto;  
    String valorCompra ;  
    int codigoBarras;  
  
    public Tipo getTipoProduto() {  
        return tipoProduto;  
    }  
  
    public void setTipoProduto(Tipo tipoProduto) {  
        this.tipoProduto = tipoProduto;  
    }  
  
    public int getCodigoBarras() {  
        return codigoBarras;  
    }  
  
    public void setCodigoBarras(int codigoBarras) {  
        this.codigoBarras = codigoBarras;  
    }  
  
    public int getCodigo() {  
        return codigo;  
    }  
  
    public String getLitros() {  
        return litros;  
    }  
}
```

Fonte: Elaborada pelo autor:

Figura 24: Trecho do código implementado na camada DAO.

```

public class UsuarioDao extends DaoConexao {

    public void cadastrar(Usuario usu) {
        String sql = "INSERT INTO usuario (senhaUsuario, nomeUsuario, cpfUsuario, clientesUsuario, produtosUsuario, "
            + "dadosEmpresaUsuario, entradaUsuario, relatoriosUsuario, vendaUsuario, tipoProdutoUsuario, usuarioUsuario) "
            + "VALUES(" + usu.getSenha()+ ", " + usu.getNome()+ ", " + usu.getCpf()+ ", "
            + "" + usu.getPermissao().getClientes()+ ", " + usu.getPermissao().getProdutos()+ ", "
            + "" + usu.getPermissao().getDadosEmpresa()+ ", " + usu.getPermissao().getEntradaProdutos()+ ", "
            + "" + usu.getPermissao().getRelatorios()+ ", " + usu.getPermissao().getVendas()+ ", "
            + "" + usu.getPermissao().getTipoProdutos()+ ", " + usu.getPermissao().getUsuarios()+ ");";

        conectar();
        try {
            comando.executeUpdate(sql);
            JOptionPane.showMessageDialog(null, "Usuário cadastrado com sucesso.", "Mensagem do Sistema",
                JOptionPane.INFORMATION_MESSAGE);
        } catch (SQLException ex) {
            JOptionPane.showMessageDialog(null, "Erro ao cadastrar o usuário. \n Erro:" + ex.getMessage(),
                "Mensagem do Sistema", JOptionPane.ERROR_MESSAGE);
        }
        fechar();
    }

    public void deletar(Usuario usu) {
        String sql = "DELETE FROM usuario WHERE idUsuario = " + usu.getCodigo() + ";";
        conectar();
        try {
            comando.executeUpdate(sql);
            JOptionPane.showMessageDialog(null, "Usuário deletado com sucesso.", "Mensagem do Sistema", JOptionPane.INFORMATION_MESSAGE);
        } catch (SQLException ex) {
            JOptionPane.showMessageDialog(null, "Erro ao deletar o usuário. \n Erro:" + ex.getMessage(), "Mensagem do Sistema",
                JOptionPane.ERROR_MESSAGE);
        }
        fechar();
    }
}

```

Fonte: Elaborada pelo autor.

Figura 25: Trecho do código implementado na camada CONTROLLER.

```

public class ClienteController {
    ClienteDao dao;

    public ClienteController() {
        dao = new ClienteDao();
    }

    public void cadastrar(Cliente cli) {
        dao.cadastrar(cli);
    }

    public void deletar(Cliente cli) {
        dao.deletar(cli);
    }

    public void alterar(Cliente cli) {
        dao.alterar(cli);
    }

    public Cliente pesquisar(String nome) {
        return dao.pesquisar(nome);
    }

    public ArrayList<Cliente> selecionar() {
        return dao.selecionar();
    }

    public ArrayList<Cliente> selecionar(String nome) {
        return dao.selecionar(nome);
    }
}

```

Fonte: Elaborada pelo autor.

Figura 26: Trecho do código implementado na camada VIEW.

```

public void pesquisar() {

    Cliente cli = controllercli.pesquisar(JTFpesquisar.getText());

    if (cli.getCodigo() == 0) {

        this.habilitarCampus(3);
        JOptionPane.showMessageDialog(null, "Cliente não encontrado",
            "Mensagem do Sistema", JOptionPane.INFORMATION_MESSAGE);

    } else {

        JTFcodigo.setText(Integer.toString(cli.getCodigo()));
        JTFnome.setText(cli.getNome());
        JTFemail.setText(cli.getEmail());
        JTFtelefone.setText(cli.getTelefone());
        JTFcelular.setText(cli.getCelular());

        JTFrua.setText(cli.getEndereco().getRua());
        JTFbairro.setText(cli.getEndereco().getBairro());
        JTFnumero.setText(Integer.toString(cli.getEndereco().getNumero()));
        JTFcep.setText(cli.getEndereco().getCep());

        JCBuf.setSelectedItem(cli.getEndereco().getCidade().getUf());
        JCBcidade.setSelectedItem(cli.getEndereco().getCidade().getNome());

        if(cli.getTipo().equals("Físico")){
            habilitarCampus(2);
            JTFcpf.setText(cli.getCPF());
            JTFrg.setText(cli.getRg());
            JTFdataNascimento.setText(cli.getDataNascimento());
            RtbpFisica.setSelected(true);
        }

        if(cli.getTipo().equals("Jurídico")){
            habilitarCampus(1);
            JTFcnpj.setText(cli.getCnpj());
            JTFincricaoEstadual.setText(cli.getIncricaoEstadual());
            RTBpJuritica.setSelected(true);
        }

    }

}

```

Fonte: Elaborada pelo autor.