

**PONTIFÍCIA UNIVERSIDADE CATÓLICA DE MINAS GERAIS**  
**Curso de Bacharelado em Sistemas de Informação**



**Michael Fagundes Marques**

**DESENVOLVIMENTO DE UM PDV HABILITADO PARA EMITIR  
NFC-E UTILIZANDO NODE.JS E NW.JS**

Belo Horizonte

2018



**Michael Fagundes Marques**

**DESENVOLVIMENTO DE UM PDV HABILITADO PARA EMITIR  
NFC-E UTILIZANDO NODE.JS E NW.JS**

Monografia apresentada ao Curso de Graduação em Sistemas de Informação da Pontifícia Universidade Católica de Minas Gerais, como requisito parcial para obtenção do grau de Bacharel em Sistemas de Informação.

Orientador: Prof. Dr. Júlio César Dillinger Conway

Belo Horizonte

2018

## **RESUMO**

Este trabalho apresenta o desenvolvimento de um ponto de venda utilizando a tecnologia NW.JS, com a capacidade de emissão de NFC-e e de fácil instalação em diversos sistemas operacionais. Com a Tecnologia de Informação se tornando indispensável para a gestão do negócio e a necessidade de automatização do processo de venda, a Secretaria da Fazenda criou o projeto de NFC-e com o objetivo de facilitar a emissão de notas fiscais por conta do vendedor, sendo um documento totalmente digital. A solução para o varejista é a utilização de um PDV que emita esse documento eletrônico de forma simples e que seja possível ser executado na maior quantidade de computadores. Para este trabalho foi utilizado o modelo de desenvolvimento de software em cascata onde cada etapa é descrita neste documento. Foi utilizada a modelagem UML na criação dos diagramas para o entendimento das funcionalidades e requisitos do sistema. Ao final, é apresentado o software desenvolvido e também as dificuldades encontradas na sua concepção, demonstrando como pode ser difícil a utilização de uma tecnologia relativamente nova e ainda em desenvolvimento.

Palavras-chave: PDV, Emissão de NFC-e, NW.JS, Javascript, HTML, CSS, SQLite, Impressão de DANFE, Sistema para o varejo, ERP para o Varejo.

## LISTA DE ILUSTRAÇÕES

Figura 1 – Tela do PDV - NFCe desenvolvido pela SerraPlan.	17
Figura 2 – Interface do Vixen PDV	18
Figura 3 – Interface do Sistema <i>Enterprise Resource Planning (ERP)</i> - Soften Siem	19
Figura 4 – Interface do PDVend	20
Figura 5 – Tela principal do NovoERP	22
Figura 6 – Diagrama de Componentes do PDV	26
Figura 7 – Diagrama de Casos de Uso do PDV	27
Figura 8 – Diagrama de Classes do PDV	28
Figura 9 – Diagrama do Processo Acesso ao PDV	29
Figura 10 – Diagrama do Processo Adicionar Produto	31
Figura 11 – Diagrama do Processo Adicionar Forma de Pagamento	32
Figura 12 – Diagrama do Processo Finalizar Venda	33
Figura 13 – Tela de configuração.	37
Figura 14 – Tela de login.	38
Figura 15 – Tela de realização de venda. Adicionando um produto na venda	39
Figura 16 – Tela de realização de venda. Após adicionar um produto na venda.	39
Figura 17 – Menu de opções	40
Figura 18 – Tela com a lista de notas geradas em contingência	41
Figura 19 – Tela para cancelar item da venda.	41
Figura 20 – Após remover um item da venda.	42
Figura 21 – Alerta de cancelamento de cupom.	42
Figura 22 – Após cancelar a venda.	43
Figura 23 – Tela de Pesquisa de Produto.	43
Figura 24 – Navegando pelos produtos filtrados.	43
Figura 25 – Após selecionar um produto da lista.	44
Figura 26 – Campos de aplicação de desconto/acréscimo	44
Figura 27 – Destaque do valor da operação de acréscimo realizada no produto	45
Figura 28 – Tela de Pagamento	45
Figura 29 – DANFE de NFC-e impressa	46
Figura 30 – PDV sendo executado no Sistema Operacional (SO) Windows.	48
Figura 31 – PDV sendo executado no SO Ubuntu	48
Figura 32 – PDV sendo executado no SO macOS.	49



## **LISTA DE TABELAS**

Tabela 1 – Requisitos levantados . . . . .	25
--------------------------------------------	----



## **Lista de Abreviaturas e Siglas**

ABRAS – Associação Brasileira de Supermercados

API – *Application Programming Interface*

CSS – *Cascading Style Sheet*

DANFE – Documento Auxiliar da NF-e

ECF – Emissor de Cupom Fiscal

ERP – *Enterprise Resource Planning*

HTML – *HyperText Markup Language*

HTTP – *HyperText Transfer Protocol*

JSON – *Javascript Object Notation*

MFE – Módulo Fiscal Eletrônico

NF-e – Nota Fiscal Eletrônica

NFC-e – Nota Fisca de Consumidor Eletrônica

PAF-ECF – Programa Aplicativo Fiscal - Emissor de Cupom Fiscal

PDF – Portable Document Format

PDV – Ponto de Venda

SAAS – *Software as a Service*

SAT CF-e – Sistema Autenticador e Transmissor de Cupons Fiscais Eletrônicos

SEFAZ – Secretaria da Fazenda

Sinief – Sistema Nacional de Informações Econômicas e Fiscais

SO – Sistema Operacional

SOAP – *Simple Object Access Protocol*

SQL – *Structured Query Language*

TI – Tecnologia da Informação

XML – *Extensible Markup Language*

## SUMÁRIO

1	INTRODUÇÃO . . . . .	11
1.1	Objetivo . . . . .	11
1.1.1	<i>Objetivos Específicos</i> . . . . .	12
1.2	Organização do Texto . . . . .	12
2	REVISÃO BIBLIOGRÁFICA . . . . .	13
2.1	PDV . . . . .	13
2.2	A importância do Ponto de Venda . . . . .	13
2.3	Nota Fiscal do Consumidor Eletrônica (NFC-e) . . . . .	14
2.4	Estado da Arte . . . . .	16
2.4.1	<i>PDV - NFCe da SerraPlan</i> . . . . .	16
2.4.2	<i>Vixen PDV da PWI</i> . . . . .	17
2.4.3	<i>PDV NFC-e da Soften Sistemas</i> . . . . .	18
2.4.4	<i>Ponto de Venda Móvel da PDVEND</i> . . . . .	19
2.4.5	<i>Conclusão da Análise</i> . . . . .	20
2.5	NovoERP . . . . .	21
3	METODOLOGIA . . . . .	23
3.1	Análise de requisitos . . . . .	24
3.2	Projeto de Sistema . . . . .	25
3.2.1	<i>Diagrama de Componentes</i> . . . . .	25
3.2.2	<i>Casos de Uso</i> . . . . .	26
3.2.3	<i>Diagrama de Classes</i> . . . . .	27
3.2.4	<i>Diagramas do Processo</i> . . . . .	29
3.2.5	<i>Tecnologias Utilizadas</i> . . . . .	33
3.2.5.1	<u>Node.JS</u> . . . . .	33
3.2.5.2	<u>Javascript</u> . . . . .	33
3.2.5.3	<u>NW.js</u> . . . . .	34
3.2.5.4	<u>AngularJS</u> . . . . .	34
3.2.5.5	<u>SQLite</u> . . . . .	34
4	IMPLEMENTAÇÃO . . . . .	37
4.1	Tela de Configuração . . . . .	37
4.2	Tela de <i>Login</i> . . . . .	38
4.3	Tela Realizar Venda . . . . .	38
4.4	Menu (F4) . . . . .	39
4.5	Cancelando Produto (F1) . . . . .	41

4.6	Cancelando Cupom (F2) . . . . .	42
4.7	Lista de Produtos (F5) . . . . .	43
4.8	Tela de Desconto/Acréscimo (F8) . . . . .	44
4.9	Tela de Pagamento (F10) . . . . .	45
5	<b>TESTES</b> . . . . .	47
5.1	Teste de Compatibilidade . . . . .	47
5.1.1	<i>SO Windows</i> . . . . .	47
5.1.2	<i>SO Linux</i> . . . . .	48
5.1.3	<i>SO macOS</i> . . . . .	48
6	<b>CONCLUSÃO</b> . . . . .	51
6.1	Dificuldades Encontradas . . . . .	51
6.2	Trabalhos Futuros . . . . .	52
	<b>REFERÊNCIAS</b> . . . . .	53
	<b>APÊNDICES</b>	55
	<b>APÊNDICE A – REPOSITÓRIO DO CÓDIGO FONTE</b> . .	57



## 1 INTRODUÇÃO

O mundo empresarial está experimentando nessas últimas décadas transformações significativas instigadas pela globalização do mercado e internacionalização das empresas, no qual concorrentes de diversas partes passaram a disputar o mesmo espaço no mercado. Com isso observou-se alterações na postura das organizações para sobreviver e competir nesse mercado, empregando estratégias para redução de custos e empregados, terceirizando as atividades-meio da empresa, redesenhando a sua estrutura de produção, entre outras (SOUZA et al., 2009).

Os especialistas do mercado financeiro afirmam que o gerenciamento adequado dos recursos financeiros é um dos requisitos básicos para a sobrevivência e sucesso de qualquer empresa TURKLE (1989).

SOUZA et al. (2009) ressaltam ainda que o ambiente em que as organizações estão inseridas apresenta-se cada vez mais complexo e a demanda por informações acuradas para auxiliar na tomada de decisão gerencial é crescente.

O controle diário do Caixa, antes realizado pelas Caixas Registradoras simples e mecânicas deu lugar a sistemas robustos de Ponto de Venda (PDV) que permitem, entre outras funcionalidades: Rastrear Vendas por Empregado ou Cliente, Processar Cartões de Crédito e Débito, Conectar-se a outros Sistemas em uma Rede e Gerenciar Inventários.

À medida que as necessidades das empresas aumentavam e a tecnologia se tornava uma ferramenta competitiva os processos das organizações tiveram que se tornar dinâmicos, tornando a utilização de softwares, que antes eram instalados no disco rígido, um grande problema.

### 1.1 Objetivo

Baseado nas informações de SOUZA et al. (2009) e nas necessidades do mercado descritas por TURKLE (1989) o objetivo do presente trabalho é desenvolver um PDV em Node.JS com foco em portabilidade, facilidade de desenvolvimento, de implantação da aplicação nos mais variados tipos de dispositivos e que atende às novas normas fiscais especificadas pelo projeto Nota Fisca de Consumidor Eletrônica (NFC-e).

O *Software* desenvolvido alinha a dinamicidade proporcionada pelos processos de Tecnologia da Informação (TI) ao processo de venda, além de ser intuitivo para o operador do sistema. A instalação pode ser feita facilmente pelo próprio usuário. Seu uso é alinhado com os cadastros das informações de produtos e fiscais feitos na plataforma web NovoERP (<http://novoerp.com.br>) desenvolvido pela empresa Avanço Informática.

### **1.1.1 *Objetivos Específicos***

Diante deste cenário, neste trabalho é realizado o desenvolvimento de um software de **PDV** em Node.JS que possibilite o lojista realizar a fácil instalação, configuração e venda de seus produtos cadastrados no software NovoERP. Deve, no final da venda, emitir uma **NFC-e** atendendo as normas fiscais elaboradas pela **GROSSO (2018)**. Deve também poder ser executado nos sistemas operacionais Windows, Linux e macOS.

## **1.2 Organização do Texto**

O restante deste trabalho está dividido da seguinte forma. No capítulo 2 é apresentada a revisão bibliográfica sobre o tema do trabalho abordando os principais tópicos envolvendo **PDV**, emissão de **NFC-e** e ferramentas utilizadas neste trabalho.

No capítulo 3 é apresentada a metodologia utilizada para o desenvolvimento do trabalho com foco no modelo de desenvolvimento utilizado, levantamento dos requisitos e concepção de todos os artefatos necessários para o desenvolvimento.

O capítulo 4 contém o desenvolvimento prático do **PDV** apresentando como foi criada as interfaces e suas funcionalidades e o teste de integração feito com o software em seu estágio final.

No capítulo 5 é concluído o trabalho apresentando as dificuldades enfrentadas no decorrer do desenvolvimento e possíveis trabalhos futuros e finalmente são apresentadas as referências bibliográficas utilizadas neste trabalho.

## 2 REVISÃO BIBLIOGRÁFICA

### 2.1 PDV

A **TI** se tornou indispensável para a gestão do negócio. A sua utilização aumentou devido a vários fatores, dentre eles: a busca por mais eficiência no desempenho operacional, a necessidade de redução de custos e integração da cadeia logística, aumento das exigências legais, fiscais e tributária, além do aumento da competitividade do mercado.

Com o tempo as organizações passaram a entender que não bastava automatizar certos processos, era necessário a implantação da **TI** em todos os setores da empresa, interligando os diversos processos e demandas. Os resultados que fossem obtidos iriam contribuir para as decisões gerenciais e estratégicas das empresas.

A adoção da **TI** pelo varejo começou inicialmente na organização interna da empresa.

“A adoção da Tecnologia da Informação pelos supermercados começou a ser implantada primeiramente para organização interna da empresa, como em departamento financeiro, recursos humanos, contabilidade e operacionais nos setores de estoque e compras, na retaguarda, jargão adotado pelo setor. Posteriormente, iniciou-se a automação das frentes das lojas, ou frente de caixa. Nesta fase, se deu início à utilização dos leitores de códigos de barras / scanners, Emissor de Cupom Fiscal (ECF), balanças eletrônicas, Transferência Eletrônica de Fundos (TEF) e inúmeras outras tecnologias que visam a aumentar a produtividade dos caixas.” (**WERNER; SEGRE, 2002**).

Segundo a Associação Brasileira de Automação Comercial (EAN BRASIL), o setor que mais utiliza leitores ópticos de códigos de barras é o setor de supermercados, seguido das lojas de departamentos e eletrodomésticos, drogarias e lojas de conveniência. Em sua última pesquisa, a Associação Brasileira de Supermercados (**ABRAS**) detectou que o setor é maciçamente automatizado, com cerca de 97% do número de lojas **LUKIANOCENKO (2006)**.

O termo Ponto de Venda assinala “os processos e atividades ligadas diretamente ao processo de vendas”, como informações ao cliente, formas de pagamento, serviços de entrega e emissão de cupons e cheques. A Automação do Ponto de Venda, por sua vez, “busca tornar mais eficiente os processos de vendas e atendimento ao cliente, além de outros serviços ligados ao *check out*, caixa, saída do cliente da loja” **ASSAF (2002)**.

### 2.2 A importância do Ponto de Venda

O **PDV** – também conhecido como Frente de Caixa faz parte da última etapa do processo de venda. Esse processo se inicia no momento em que é realizado o pedido do produto ao fornecedor, “passa por todo o controle no recebimento e cadastro de produto,

até chegar no controle do estoque que é fundamental e serve de base para a concretização da venda no seu PDV” ([GOLPI, 2017](#)).

Um sistema de [PDV](#) é fundamental para alinhar e associar todos esses processos, além de oferecer agilidade e evitar erros.

Esse sistema é responsável por conectar com a parte fiscal da loja com todo o processo do atendimento, funcionando de forma sincronizada, tornando o processo de compra mais agradável ao cliente. Processos automatizados diminuem filas e problemas.

### **2.3 Nota Fiscal do Consumidor Eletrônica (NFC-e)**

A tecnologia de informação está cada vez mais presente na vida dos lojistas do Brasil, e um dos projetos do governo Brasileiro visa facilitar a emissão de notas fiscais, automatizando esse processo [TECNOSPEED \(2013\)](#).

A [NFC-e](#) é um modelo de documento fiscal totalmente eletrônico, emitido e armazenado eletronicamente. De existência apenas digital, tem como objetivo o registro de operações de venda presencial ao consumidor final no varejo. Sua validade jurídica é garantida pela assinatura digital do emissor e a autorização de uso concedida pela Secretaria de Estado da Fazenda [GROSSO \(2018\)](#).

Baseado no padrão e normas já existentes da Nota Fiscal Eletrônica ([NF-e](#)) com alterações para atender o mercado varejista, sua validade jurídica foi determinada no Ajuste SINIEF nº 07/2005 que instituiu a [NF-e](#) como um documento jurídico com o seguinte conceito legal:

“Um documento emitido e armazenado eletronicamente, de existência apenas digital, com o intuito de documentar operações e prestações, cuja validade jurídica é garantida pela assinatura digital do emitente e autorização de uso Pela administração tributária da unidade federada do contribuinte, antes da ocorrência do fato gerador.” ([FAZENDA, 2005](#)).

É necessário enfatizar a [NFC-e](#) representa uma imposição da legislação, pois sua utilização está atrelada a adoção do padrão pelo estado do contribuinte, portanto, sua utilização deve ser adotada até o prazo limite para que não seja aplicada nenhuma punição referente ao descumprimento das normas legais exigidas do estado ([FAZENDA, 2005](#)).

A [NFC-e](#) teve seu início no final de 2011 com reuniões técnicas da Secretaria da Fazenda ([SEFAZ](#)) e em 2012 com os primeiros testes compostos por empresas varejistas dos estados Acre, Amazonas, Maranhão, Mato Grosso, Rio Grande do Norte, Rio Grande do Sul e Sergipe. Esse grupo de empresas teve o papel de evoluir o modelo inicial proposto pela [SEFAZ](#) atenuando o impacto desse novo modelo para o varejo brasileiro [TECNOSPEED \(2013\)](#).

O projeto [NFC-e](#) foi criado no Ajuste do Sistema Nacional de Informações Econômicas e Fiscais ([Sinief](#)) de nº 01/2013 que alterou o Ajuste [Sinief](#) de nº 07/2005 [NF-e Fazenda \(2005\)](#).

A [NFC-e](#) tem como objetivo principal a substituição da emissão de documento fiscal em papel, conforme o descritivo retirado do site do Ministério da Fazenda:

“O Projeto NFC-e tem como objetivo a implantação de um modelo nacional de documento fiscal eletrônico que venha substituir a sistemática atual de emissão do documento fiscal em papel, com validade jurídica garantida pela assinatura digital do remetente, simplificando as obrigações acessórias dos contribuintes.” ([FAZENDA, 2005](#)).

Portanto visa substituir gradativamente os seguintes documentos fiscais: [GROSSO \(2018\)](#)

- Notas Fiscais modelo 1 e modelo 1-A quando utilizada na venda a varejo;
- Notas Fiscais modelo 2;
- Cupom Fiscal emitido por equipamentos Emissor de Cupom Fiscal ([ECF](#));
- Notas Fiscais Eletrônicas modelo 55 quando utilizadas na venda a varejo;

A implantação do projeto da [NFC-e](#) representou um grande avanço para facilitar a vida de todos os envolvidos, o contribuinte, consumidor e controle do fisco. Facilita o registro de operações e prestação de tributos na venda de mercadorias destinadas ao consumidor final [TECNOSPEED \(2013\)](#).

Os estados de Acre, Amazonas, Bahia, Distrito Federal, Espírito Santo, Goiás, Maranhão, Mato Grosso, Paraíba, Paraná, Piauí, Rio de Janeiro, Rio Grande do Norte, Rondônia, Roraima, Sergipe já obrigam a emissão de [NFC-e](#) pelo varejista. Alagoas, Amapá, Mato Grosso do Sul, Minas Gerais, Pernambuco, Rio Grande do Sul possuem seus respectivos cronogramas para implantação do projeto com previsão de finalização em 2018 até 2020. O estado do Ceará e São Paulo utilizam os padrões Módulo Fiscal Eletrônico ([MFE](#)) e Sistema Autenticador e Transmissor de Cupons Fiscais Eletrônicos ([SAT CF-e](#)) respectivamente. São padrão semelhantes ao de [NFC-e](#). O único estado que ainda não possui cronograma de implantação é o de Tocantins [Karine \(2018\)](#).

A [NFC-e](#) é constituída a partir de um arquivo *Extensible Markup Language (XML)* contendo informações do estabelecimento emissor, entidade destinatária (nesse caso o consumidor final), informações da venda, informações dos produtos envolvidos bem como valores de venda e valores referentes a tributos e informações de pagamento [TECNOSPEED \(2013\)](#). Após a construção desse arquivo o mesmo é assinado digitalmente utilizando o certificado digital do estabelecimento emissor no padrão ICP-Brasil [GROSSO \(2018\)](#).

A transmissão desse [XML](#) se dá via *webservice Simple Object Access Protocol (SOAP)* disponibilizado por cada unidade federada. A comunicação é feita utilizando o certificado digital da empresa emitente para assim manter segurança e legitimidade das informações transmitidas [GROSSO \(2018\)](#). O envio do arquivo é feito para adquirir a autorização de uso da [SEFAZ](#), tornando assim o arquivo como um documento fiscal digital [GROSSO \(2018\)](#).

Após a validação do arquivo deve ser impresso, em duas vias, o Documento Auxiliar da NF-e ([DANFE](#)). [DANFE](#) é uma representação da venda realizada, contendo, de forma simplificada, informações da transação e tem como objetivo facilitar a consulta do documento fiscal eletrônico pelo cliente ([FAZENDA, 2017](#)).

## 2.4 Estado da Arte

Tendo em vista o objetivo do trabalho e a fim de realizar a análise do cenário atual de softwares [PDV](#) com suporte a emissão de [NFC-e](#), foram analisadas as soluções das seguintes empresas:

- PDV - NFCe da SerraPlan;
- Vixen PDV da PWI;
- PDV NFC-e da Soften Sistemas;
- Ponto de Venda Móvel da PDVEND.

### 2.4.1 *PDV - NFCe da SerraPlan*

A empresa SerraPlan, situada em Petrópolis - RJ, possui diversos produtos fiscais sendo um deles o PDV - NFC-e, sendo capaz de realizar vendas e emitir [NFC-e](#) a partir destas vendas. Seu produto possui as funcionalidades básicas de um [PDV](#) que atendem as especificações do projeto [NFC-e](#) bem como funcionalidades mais elaboradas como por exemplo controle de estoque, capacidade de realizar pré-vendas e orçamentos, programa de fidelidade de clientes, múltiplas tabelas de preço e controle de desconto máximo.

De acordo com o *site* do produto existe a possibilidade da contratação do [PDV](#) com um software de gestão comercial via *web*, agregando diversas funcionalidades de gestão e administração ao [PDV](#), desde gestão de pessoas, a controle financeiro, relatório entre outras.

Para utilizar esse [PDV](#) é necessário o preenchimento de um formulário solicitando mais informações sobre o plano desejado. A Figura 1 apresenta a interface do PDV desenvolvido pela SerraPlan.

Figura 1 – Tela do PDV - NFCe desenvolvido pela SerraPlan.



Fonte: <http://www.spcommerce.com.br>

O PDV - NFCe funciona localmente, em cada caixa, utilizando da internet na transmissão do [XML](#) da [NFC-e](#) e/ou na sincronização das informações com o [ERP](#) web.

#### **2.4.2 Vixen PDV da PWI**

Nas palavras da PWI, empresa que desenvolveu o Vixen PDV:

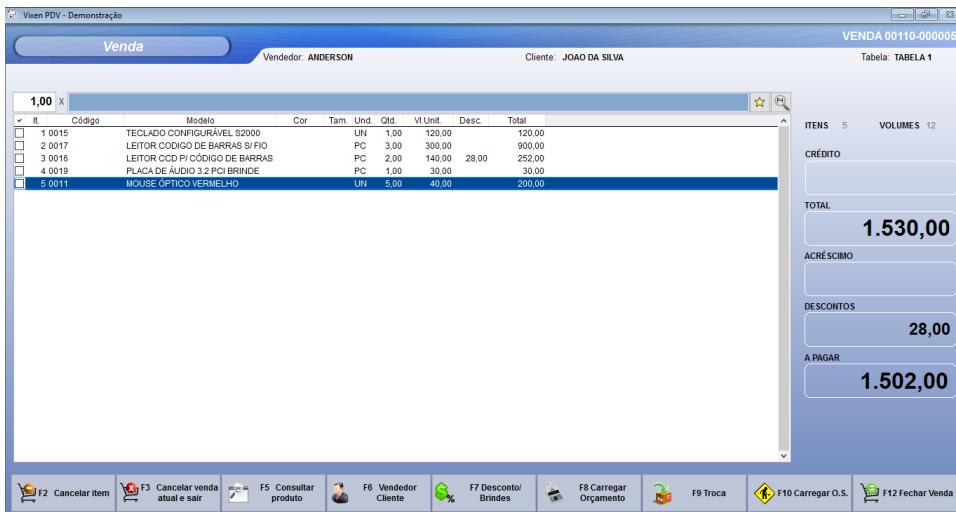
“O Vixen PDV é um software que foi desenvolvido para o gerenciamento de lojas, atacado ou varejo, auxiliando no atendimento aos clientes, automatizando controles internos e gerando informações gerenciais precisas. Homologado para uso em todo o território nacional, o sistema de emissão de cupom fiscal atende a todas as tecnologias exigidas pelo FISCO.” [PWI WEV STUDIO \(2017b\)](#).

Este software além de emitir [NFC-e](#) é capaz de emitir [SAT CF-e](#), o padrão utilizado no estado de São Paulo, e também Programa Aplicativo Fiscal - Emissor de Cupom Fiscal ([PAF-ECF](#)), o antigo cupom fiscal. É, portanto, uma ferramenta completa, sendo capaz de sozinha atender os padrões fiscais novos e antigos na mesma aplicação. Possui as funcionalidades básicas de venda e também mais avançadas:

- Consulta de Cheque;
- Orçamentos, Pré-Vendas, Ordem de Serviços;
- Devolução;
- Cartão Fidelidade;
- Consulta de Cheque;
- Controle de Conta Corrente (Vendas com pagamento posterior, o conhecido Fiado).

A Figura 2 mostra a interface do Vixen PDV.

Figura 2 – Interface do Vixen PDV



Fonte: Pwi Web Studio

Junto com o **PDV** é necessário a utilização de um sistema **ERP**, desenvolvido e fornecido pela própria empresa, que possui as seguintes funcionalidade em sua versão básica:

- Cadastros básicos: fornecedores, clientes, alíquotas fiscais, estoque;
- Cadastro de produtos (para venda e consumo interno);
- Módulo de vendas;
- Módulo de compras;
- Módulo financeiro.

Para contratar o **PDV** e o **ERP** é necessário entrar em contato com a empresa para orçar o valor da compra e implantação dos softwares.

#### 2.4.3 PDV NFC-e da Soften Sistemas

O **PDV** NFC-e desenvolvido pela Soften Sistemas é um módulo do Sistema **ERP** - Soften Siem da própria empresa, isso quer dizer que ele não funciona isoladamente, diferente dos dois últimos aplicativos. Por ser um módulo ele funciona totalmente integrado com a solução **ERP** da empresa, sendo que a interface do emissor é acionada na opção **Frente de Caixa** do sistema **ERP**. Por ser integrado ao **ERP**, além de possuir as funcionalidades básicas de um **PDV**, ele possui todas as ferramentas gerenciais, administrativas e fiscais

incluídas no mesmo pacote, sendo que essas ferramentas podem ser contratadas como módulos adicionais. A Soften descreve seu [ERP](#) da seguinte maneira:

“O Soften SIEM é um software (programa de computador) que tem por especialidade administrar de maneira integrada as diversas rotinas administrativas de micros e pequenas empresas. Os principais recursos do software são os módulos de emissão de documentos fiscais, tais como NF-e (Nota Fiscal Eletrônica), CT-e (Conhecimento de Transporte Eletrônico), NFC-e (Nota Fiscal do Consumidor Eletrônica), MDF-e (Manifesto de Documento Fiscal Eletrônico) e SPED (Sistema Público de Escrituração Digital) e muito mais.” [SOFTEN SISTEMAS \(2018\)](#).

A Figura 3 exibe as interfaces do Sistema ERP da Soften Siem em vários dispositivos.

Figura 3 – Interface do Sistema [ERP](#) - Soften Siem



Fonte: Soften Sistemas

Esse software é executado totalmente local, em cada caixa, utilizando a internet somente no envio do [XML](#) da [NFC-e](#). Para contratação do [ERP](#), o formato é igual ao utilizado pela PWI, é necessário entrar em contato com a empresa para orçar o valor do projeto.

#### **2.4.4 Ponto de Venda Móvel da PDVEND**

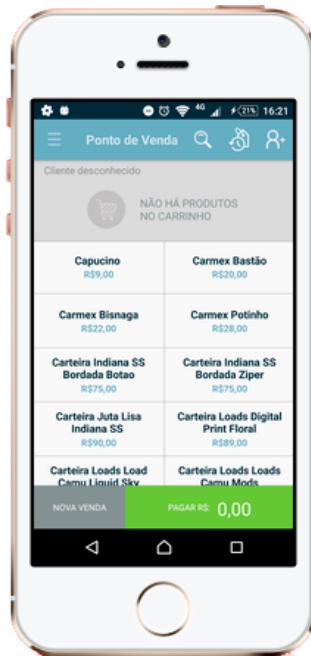
PDVend é um [PDV](#) da *startup* de mesmo nome, fundada em 2015, que foi finalista da competição de *startups Like a Boss* organizada pela Sebrae em 2017 ([REDAÇÃO PAGAMENTO.ME, 2016](#)). Com foco em pequenos comércios, como por exemplo loja de roupas, restaurantes, entre outros ([PDVEND – SISTEMA DE VENDAS PARA PDV E CAIXA DE LOJA, 2018](#)), ele é um software com o objetivo de ser simples e prático.

É possível utilizá-lo somente via WEB, pelo navegador do usuário, ou instalado em dispositivos *Android*. A interface do PDVend é mostrada na Figura 4.

Ele possui um plano que permite somente a venda e o histórico de vendas, um para gestão de lojas sem emissão de [NFC-e](#) e o plano completo com gestão de lojas e vendas. Pelo site é possível utilizar gratuitamente a ferramenta por 7 dias

A plataforma, isto é, o [PDV](#) mais a ferramenta de gestão de loja, possui as funcionalidades de controle de estoque, relatórios, envio automático das notas enviadas para o contador, mas a mais importante é a possibilidade de utilização do [PDV](#) em qualquer dispositivo com um navegador de *internet*.

Figura 4 – Interface do PDVend



Fonte: PDVend

#### 2.4.5 Conclusão da Análise

Pode-se entender que existem diferentes configurações de [PDV](#) utilizadas no mercado, desde a precificação do sistema, plataformas e principalmente se tratando de como é gerenciado os dados necessários para que o mesmo possa funcionar.

No quesito comercial é possível identificar que variam as formas em que se pode adquirir um [PDV](#). Para os mais simples, é possível a compra do direito de uso do programa responsável por somente vender e enviar a [NFC-e](#), que é o caso do PDV - NFC-e da SerraPlan. Para os mais complexo, em que a utilização do [PDV](#) está atrelada ao uso de um [ERP](#) para gestão das informações, é necessário o levantamento do que é necessário para o lojista para assim então chegar ao preço desse pacote de soluções. Para as soluções mais novas, como por exemplo o PDVend que pode funcionar totalmente em uma plataforma web

ou móvel, já é utilizado uma especificação mais ao estilo de um *Software as a Service (SAAS)* no qual o cliente paga por tempo de utilização do produto, geralmente mensalmente.

Em todas as soluções apresentadas é possível identificar a necessidade de um programa, geralmente a parte, que gerencia os dados utilizados pelo **PDV**. Geralmente chamado de retaguarda. PWI Web Studio define que:

“Sistema de retaguarda é o software onde são “trabalhadas” as informações para o que o sistema de frente de caixa possa realizar as vendas. Todos os cadastros que mencionamos acima são realizados através do sistema de retaguarda. O sistema de retaguarda consiste em um sistema ERP.” ([PWI WEV STUDIO, 2017a](#)).

Este sistema controla os dados de cadastros dos produtos, dados tributários, vendas e outras informações adicionais que agregam na utilização do **PDV**. Ele é importante para a utilização do **PDV** pois é com esse sistema é realizado o cadastro das informações necessárias para a venda e geração da **NFC-e**.

Outro ponto que é importante salientar é a necessidade de conexão com a internet. Devido ao requisito de que a **NFC-e** seja enviada pela internet para os servidores da **SEFAZ**, o lojista que tenha que utilizar um **PDV** com emissão de **NFC-e** deve possuir uma conexão com a internet para que este envio seja realizado.

## 2.5 NovoERP

O NovoERP é um **ERP** desenvolvido pela empresa Avanço Informática com o objetivo se ser uma solução simples para atender o pequeno varejista. Ele é um **SAAS**, funcionando na nuvem, da estrutura e serviços disponibilizados pela *Amazon Web Services*.

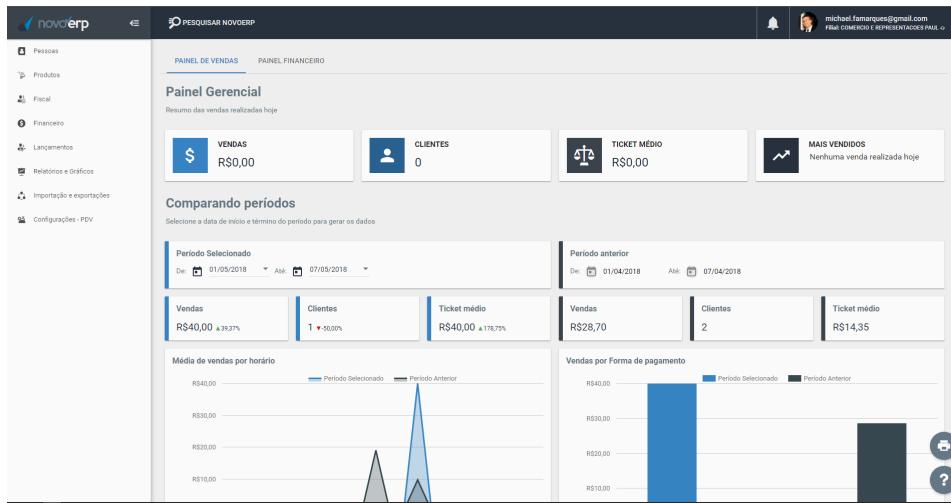
É um software relativamente novo, com menos de 2 anos de mercado, desenvolvido totalmente utilizando Javascript, *HyperText Markup Language (HTML)* e *Cascading Style Sheet (CSS)* em sua interface e Node.JS para seu *backend*, as mesmas tecnologias utilizadas para a criação do **PDV**.

Ele é o software utilizado como retaguarda para o **PDV** desenvolvido, sendo responsável pelo cadastro de informações e também envio do **XML** para os servidores da **SEFAZ**.

Foi selecionado como **ERP** do **PDV** porque o autor deste trabalho é um dos desenvolvedores do sistema e já possui conhecimento da *Application Programming Interface (API)* e das informações necessárias que o NovoERP necessita para utilização.

Na Figura 5 é possível ver a tela principal do sistema, que possui dados e gráficos referente as vendas realizadas.

Figura 5 – Tela principal do NovoERP



Fonte: Elaborado pelo autor

### 3 METODOLOGIA

Na engenharia de software é importante a escolha de uma metodologia de desenvolvimento, conforme PAULA FILHO (2009) salienta:

“Em engenharia de software, processos podem ser definidos para atividades como desenvolvimento, manutenção, aquisição e contratação de software. Podem-se também definir subprocessos para cada um desses; por exemplo, um processo de desenvolvimento abrange subprocessos de determinação dos requisitos, análise, desenho, implementação e testes. Em um processo de desenvolvimento de software, o ponto de partida para a arquitetura de um processo é a escolha de um modelo de ciclo de vida ou método de engenharia de software.” (PAULA FILHO, 2009, p. 23).

O PDV foi desenvolvido utilizando o modelo de cascata com as seguintes etapas: levantamento de requisitos, criação do projeto e arquitetura do sistema, implementação e validação com realização de testes. Todas estas etapas são documentadas neste trabalho.

O modelo em cascata é um processo sequencial de etapas no qual a próxima etapa é executada somente após a conclusão de sua fase anterior. Cada fase pode possuir documentos que serão utilizados na etapa posterior SOMMERVILLE (2011). Este modelo possui 5 etapas principais:

1. Definição de Requisitos;
2. Projeto do Sistema;
3. Implementação;
4. Validação;
5. Entrega e Manutenção.

SOMMERVILLE (2011) também salienta que podem ser identificados problemas nos resultados da etapa anterior, como por exemplo problemas de requisitos, problemas de implementação, entre outro. Portanto pode existir um *feedback* entre as etapas, podendo então um documento da etapa anterior pode ser modificado a ponto de refletir sua alteração realizada na etapa posterior.

Na primeira etapa, a de definição de requisitos, foram realizadas entrevistas com desenvolvedores com experiência em desenvolvimento de PDV para identificar quais as funcionalidades principais e básicas que este tipo de ferramenta deve possuir para a realização de uma venda em um caixa e a geração de uma NFC-e. A partir do levantamento destas funcionalidades foram gerados requisitos funcionais para o sistema em conjunto com os requisitos já descritos para a emissão de NFC-e.

Na etapa de projeto de sistema, a partir dos requisitos levantados foi criado um projeto definindo como o **PDV** deve operar, em termos de interface, hardware, processos e software.

Já na implementação foi desenvolvido todo o código do **PDV** com o objetivo de atender os requisitos especificados na primeira etapa de definição de requisitos, seguindo o planejamento criado na segunda etapa.

O **PDV** precisa ser validado para garantir que ele executa o seu objetivo proposto. Para isso, na quarta etapa, o **PDV** passou por testes de compatibilidade para validar seu funcionamento nos três sistemas operacionais propostos.

### 3.1 Análise de requisitos

De acordo com Sommerville (2011) requisitos de sistema são as descrições do que o sistema deve fazer, os seus serviços oferecidos e suas restrições funcionais. Eles devem refletir as necessidades dos clientes para que com o sistema e como ele deve resolver estes problemas, por exemplo, enviar um pedido, encontrar informações ou controlar um dispositivo.

Utilizando técnicas de modelagem de processos de negócio, é possível entender melhor o sistema que será desenvolvido e como deve funcionar, possibilitando a identificação de requisitos que correspondem as reais necessidades do negócio (BAKER, 2001). Este trabalho apresenta uma modelagem de requisitos utilizando a notação UML. Os requisitos foram levantados por meio de entrevistas, método já utilizado na maioria dos processos de engenharia de requisitos. Foram selecionados desenvolvedores que já trabalham com o desenvolvimento deste tipo de ferramenta e realizadas questões aos mesmos sobre quais funcionalidades seriam importantes para uma ferramenta deste tipo funcionar. A partir das respostas destas entrevistas foram levantados os requisitos. E conjunto do resultado destas entrevistas foram utilizados alguns requisitos existentes do projeto NFC-e.

Na Tabela 1 são apresentados os requisitos levantados nesta etapa:

Tabela 1 – Requisitos levantados

Análise de Requisitos levantados	
Requisito	Descrição
Configuração	i. O operador de caixa deve ser capaz de instalar e configurar a ferramenta ii. Sincronização de dados com o ERP
Autenticação	i. O PDV deve solicitar ao operador de caixa o seu número de operador e senha para liberar o acesso
Processo de Venda	i. Adição de produto na venda ii. Remoção de produto na venda iii. Adição de acréscimo/desconto no produto iv. Validar regras de produto ao adicioná-lo na venda v. Adicionar forma de pagamento vi. Validar regras de forma de pagamento vii. Cancelar a venda corrente viii. Cancelar uma venda já realizada ix. Concluir a venda x. Emitir a NFC-e xi. Imprimir um cupom fiscal
Emitir NFC-e	i. Gerar XML ii. Assinar XML iii. Enviar XML via API iv. Tratar modo de Contingência v. Tratar retorno

Fonte: Elaborado pelo autor

## 3.2 Projeto de Sistema

Sommerville (2011) define projeto de software da seguinte maneira:

“Um projeto de software é uma descrição da estrutura do software a ser implementado, dos modelos e estruturas de dados usados pelo sistema, das interfaces entre os componentes do sistema e, às vezes, dos algoritmos usados. Os projetistas não chegam a um projeto final imediatamente, mas desenvolvem-no de forma iterativa. Eles acrescentam formalidade e detalhes, enquanto desenvolvem seu projeto por meio de revisões constantes para correção de projetos anteriores.” (SOMMERVILLE, 2011, p. 25).

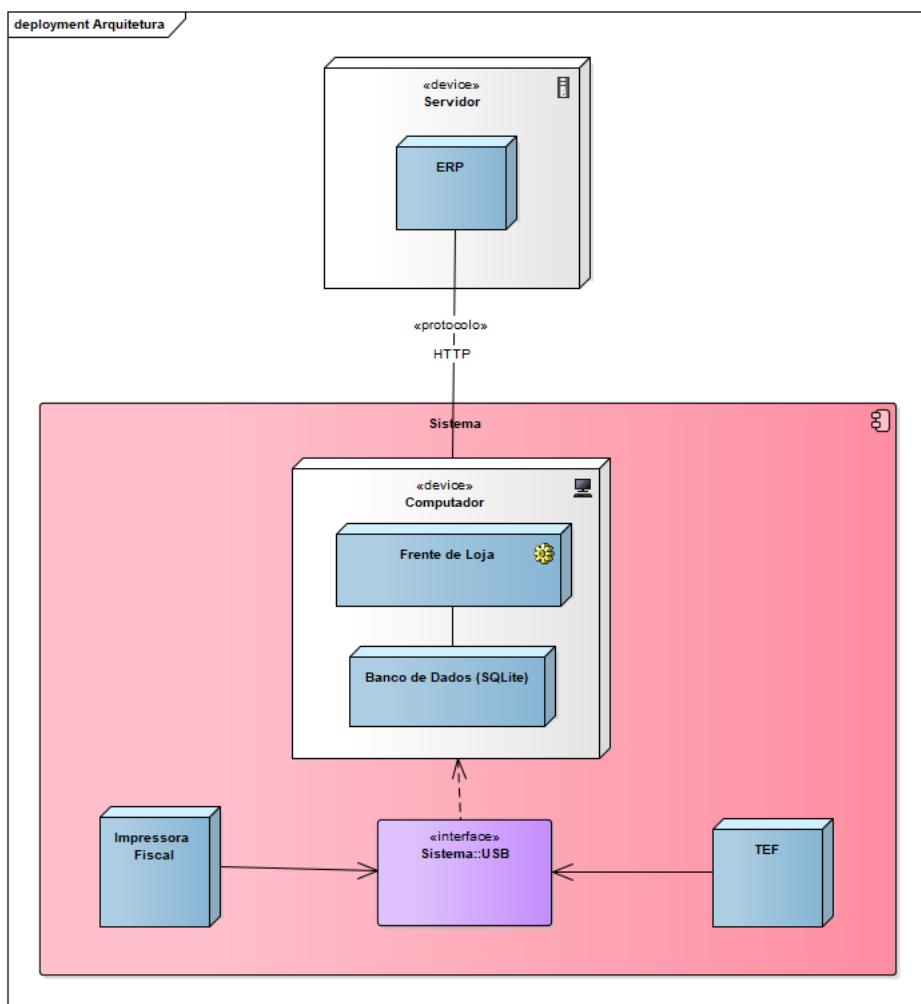
Serão apresentados a seguir a estrutura levantada para execução deste trabalho.

### 3.2.1 Diagrama de Componentes

A Figura 6 apresenta os componentes do sistema e a interação entre eles. No componente sistema tem-se o computador no qual está instalado o PDV. No mesmo computador está o arquivo de banco de dados em SQLite, instalado junto com o programa

e consumido por uma biblioteca nativa do sistema operacional que o **PDV** utiliza, sem a necessidade de ter um servidor instalado como em outros sistemas de banco de dados. Por meio de uma interface USB o **PDV** pode se comunicar com diversos dispositivos, entre ele a impressora fiscal. Inicialmente foi implementada a comunicação com impressoras para a impressão da **DANFE**, requisito do padrão **NFC-e**. O sistema pode se comunicar com o **NovoERP**, que se encontra disponível em um servidor na internet por meio do protocolo *HyperText Transfer Protocol (HTTP)*, utilizando o formato de dados *Javascript Object Notation (JSON)*. **JSON** foi escolhido pois é o formato de dados utilizado pela **API** fornecida do **NovoERP**.

Figura 6 – Diagrama de Componentes do PDV



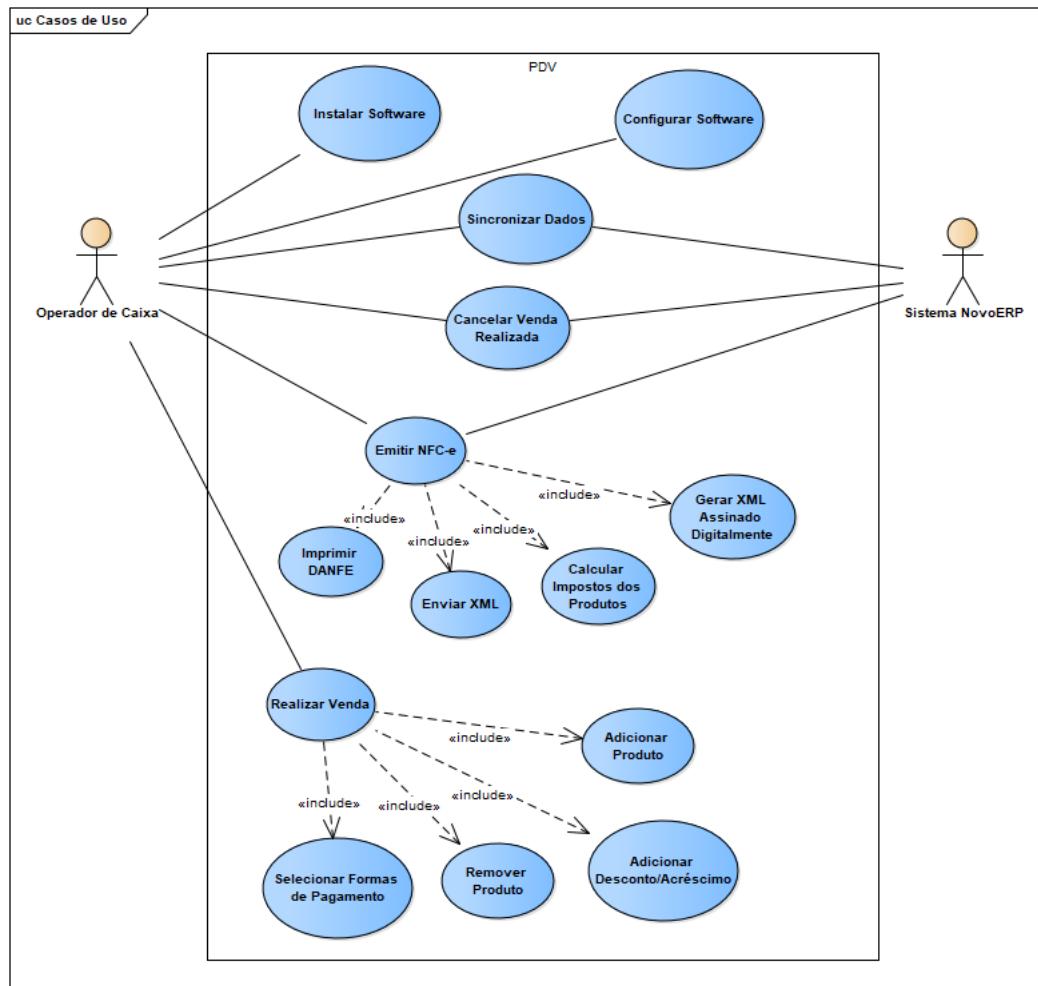
Fonte: Elaborado pelo autor

### 3.2.2 Casos de Uso

A Figura 7 exibe os principais casos de usos, atores e interações entre eles com relação ao **PDV** desenvolvido. Nesse diagrama tem-se os casos de uso de instalação e configuração inicial do sistema pelo Operador de Caixa, ações que são realizadas somente uma vez, na primeira execução do programa. Tem-se também o processo de venda de

produtos, que engloba outros pequenos processos, desde a inclusão de produtos à escolha da forma de pagamento.

Figura 7 – Diagrama de Casos de Uso do PDV



Fonte: Elaborado pelo autor

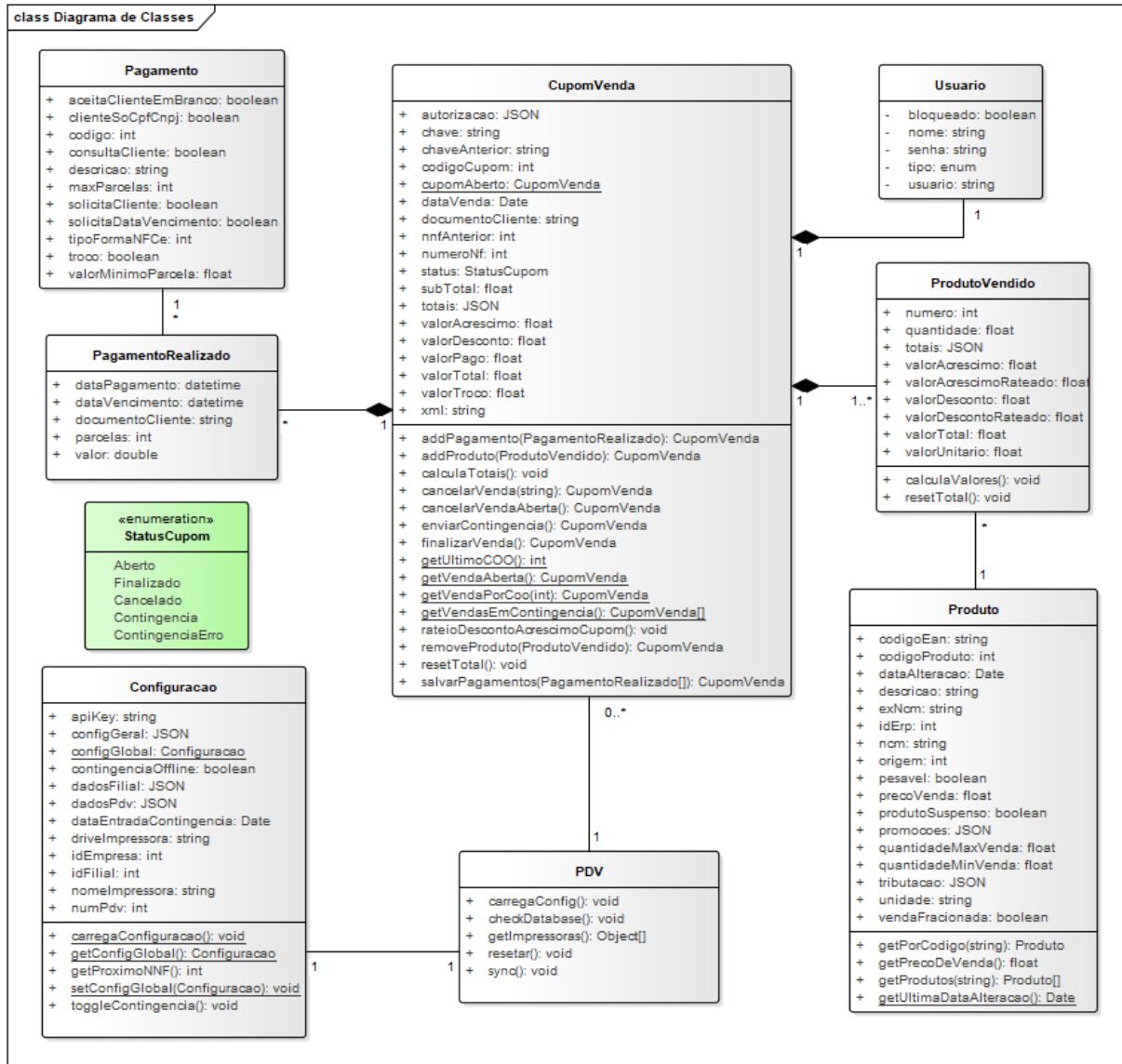
É especificada a ação de cancelar uma venda, que envolve a utilização da comunicação com o [ERP](#) NovoERP para cancelamento de vendas já realizadas ou então o cancelamento de uma venda em execução. Possui também o processo de emissão da [NFC-e](#). Este processo envolve desde a criação do arquivo XML com as informações da venda, assinatura digital e envio do mesmo para o [ERP](#) realizar a comunicação com os servidores da [SEFAZ](#).

### 3.2.3 Diagrama de Classes

A Figura 8 apresenta as entidades utilizadas pelo sistema. A entidade Produto remete ao produto disponível para venda. Cada produto possui suas informações fiscais necessárias para preenchimento das informações na [NFC-e](#), de venda e de identificação. Já a entidade Promoção serve para aplicar um valor diferente do normal a produtos

relacionados a partir da data da venda estar dentro do período definido na promoção.

Figura 8 – Diagrama de Classes do PDV



Fonte: Elaborado pelo autor

A entidade Usuario representa o usuário que acessa o sistema, com seus dados de acesso. A entidade Pagamento representa a forma de pagamento que o cliente que está realizando a compra pode utilizar, podendo ser parametrizado para realizar validações de utilização da forma de pagamento como por exemplo quantidade máxima de parcelas, se permite troco ao cliente entre outras.

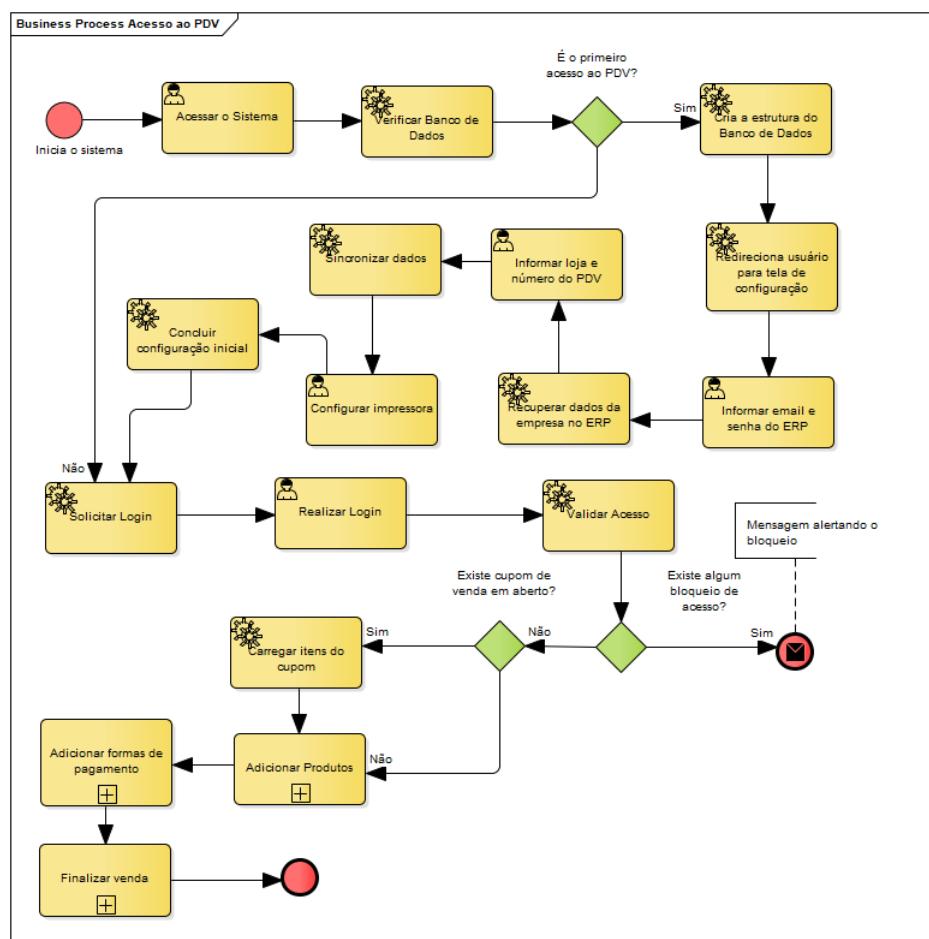
A entidade CupomVenda que representa o ato da venda e possui informações relacionadas a ela. A entidade ProdutoVendido representa a ligação entre uma venda e o produto adicionado. Já a entidade PagamentoRealizado representa a relação do CupomVenda com o Pagamento, já que uma venda pode possuir um ou mais formas de pagamento.

A entidade PDV representa o sistema. Ela é a peça central do sistema e a partir dela é realizada ações de sincronização com o NovoERP e tratamento das configurações do PDV.

### 3.2.4 Diagramas do Processo

A Figura 9 (Acesso ao PDV) apresenta atividades executadas pelo sistema e usuário no primeiro acesso. O sistema verifica se existe o banco de dados e o mesmo já foi configurado. Caso o PDV já tenha sido configurado o usuário é redirecionado para a tela de *login* solicitando os dados de acesso do operador, previamente cadastrados no NovoERP, mas se é o primeiro acesso ele é redirecionado para a tela de configuração. Nesta tela o usuário informa o *e-mail* e senha de um usuário já cadastrado no NovoERP. Com os dados validados o PDV realiza o download dos dados da empresa previamente configurados no ERP e solicita ao usuário a seleção da loja no qual o PDV pertence e o número do PDV a ser configurado.

Figura 9 – Diagrama do Processo Acesso ao PDV



Fonte: Elaborado pelo autor

A seleção da loja serve para que o PDV realize o download das informações

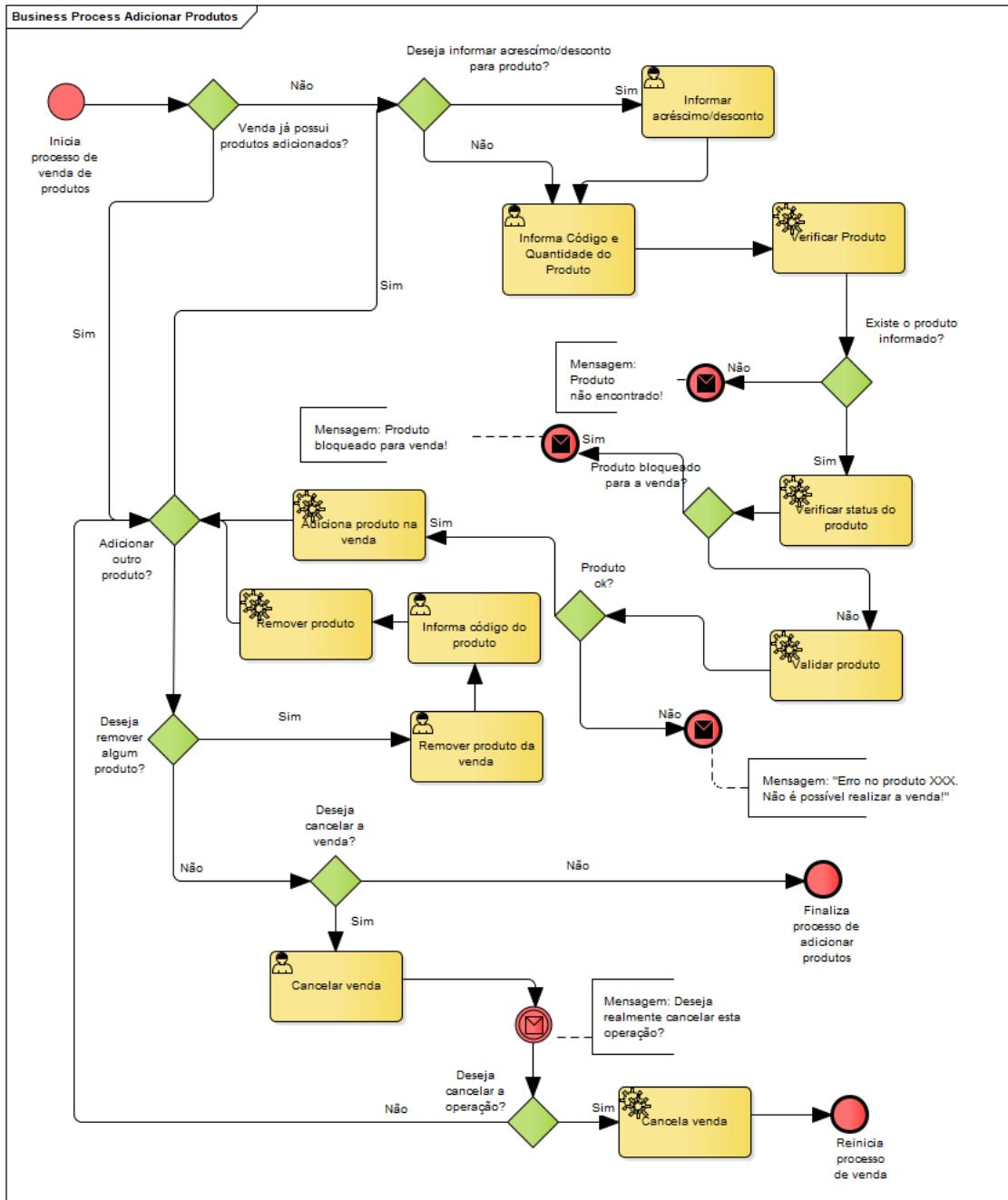
relacionadas somente a ela. A seleção do número do **PDV** é para identificar a estação de venda pois em uma loja pode existir uma ou mais instalações de **PDV** funcionando ao mesmo tempo. Após informados esses dados o **PDV** realiza a sincronização dos dados dos produtos que serão vendidos.

Estes dados de produtos são desde as informações de cadastro simples como descrição, valor, se é um produto pesável, origem do produto e também informações de tributação. Estas informações podem servir tanto para o processo de venda e também para a emissão da **NFC-e**.

Após este processo ele solicita ao usuário o login e senha do mesmo para validação do acesso. Caso o usuário esteja com o acesso liberado, o sistema verifica se existe alguma venda pendente que não foi finalizada anteriormente para carregá-la antes de iniciar o processo de venda descrito no diagrama abaixo.

A Figura 10 representa o fluxo de adição de produtos na venda pelo operador de caixa. O processo é iniciado com o usuário informando o código do produto e a quantidade desejada. Após informado esses dados o sistema realiza uma série de verificações nos dados do produto que, caso esteja sem problema, é adicionado a venda. Caso não exista uma venda em aberto o sistema do **PDV** automaticamente inicia uma nova venda dando sequência a numeração de vendas realizadas.

Figura 10 – Diagrama do Processo Adicionar Produto



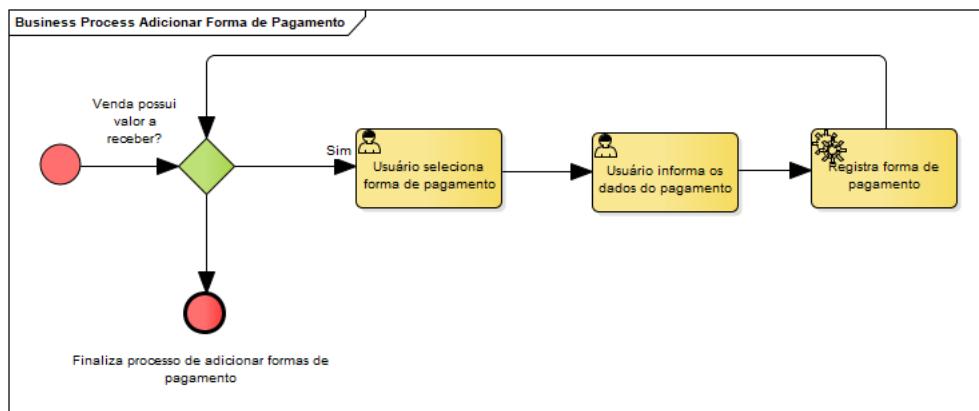
Fonte: Elaborado pelo autor

Além de adicionar produtos o operador tem a possibilidade de aplicar um desconto ou acréscimo no próximo produto a ser adicionado. Ele também pode cancelar um produto já adicionado a venda.

O operador de caixa pode repetir esse processo quantas vezes quiser até ter adicionado todos os produtos desejados na venda. O operador pode, no final, cancelar a venda em aberto reiniciando o processo de venda, removendo assim todos os produtos adicionados à venda, ou então encerrar a venda indo para o processo de escolha de formas de pagamento.

Para o processo de adicionar formas de pagamento, demonstrado na Figura 11, o usuário pode adicionar uma ou mais formas sempre informando valor e se desejadas opções de parcelamento. Após o usuário informar todas as formas de pagamento, ele finaliza o processo de venda registrando o mesmo no sistema com todas suas informações de produtos e pagamentos. Após isso o sistema **PDV** se encarrega de emitir a **NFC-e**, processo descrito a seguir.

Figura 11 – Diagrama do Processo Adicionar Forma de Pagamento

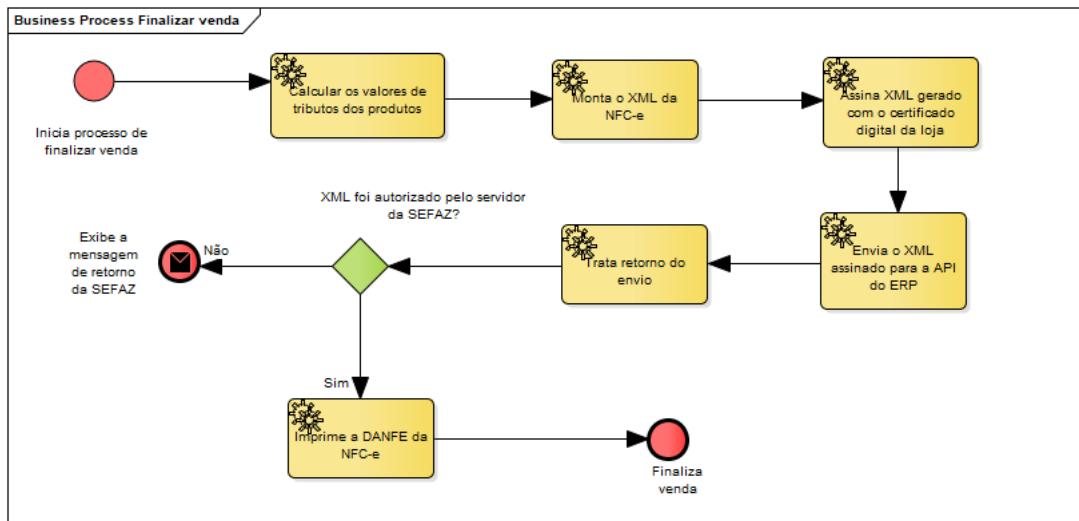


Fonte: Elaborado pelo autor

Na Figura 12 é exibido o processo de finalização da venda. O **PDV** primeiro calcula todos os valores de tributação dos produtos, valores que vão de impostos estaduais e federais. Após este cálculo é então gerado o **XML** de acordo com a especificação criada pela **SEFAZ SPED (2012)**. Estas informações são os dados da empresa emissora, informações da venda realizada, dados dos produtos e suas tributações, informações dos pagamentos realizados e valores totais da venda.

Após a criação do arquivo **XML** o mesmo então é assinado digitalmente utilizando o certificado digital da empresa. Essa assinatura segue as especificações descritas pela **SEFAZ** no manual **SPED (2012)**. Após **XML** criado e assinado o mesmo é enviado para a **API** do sistema NovoERP. Ele é encarregado de realizar a comunicação com os serviços da **SEFAZ**, enviando o arquivo e retornando a resposta do serviço. Após a resposta retornada, com as informações de autorização de uso da venda realizada, o **PDV** imprime então a **DANFE** da **NFC-e** criada, em duas vias, uma para a loja e outra para o cliente. Após a impressão o **PDV** então reinicia todo o processo de venda, retornando então para a tela de adição de produtos.

Figura 12 – Diagrama do Processo Finalizar Venda



Fonte: Elaborado pelo autor

### 3.2.5 Tecnologias Utilizadas

A proposta deste trabalho é o desenvolvimento de um [PDV](#) utilizando o Node.JS. Sendo assim, nos tópicos abaixo, são apresentadas as tecnologias utilizadas em conjunto no desenvolvimento deste trabalho.

#### 3.2.5.1 Node.JS

“Node.js é uma plataforma construída sobre o motor JavaScript do Google Chrome para facilmente construir aplicações de rede rápidas e escaláveis. Node.js usa um modelo de I/O direcionada a evento não bloqueante que o torna leve e eficiente, ideal para aplicações em tempo real com troca intensa de dados através de dispositivos distribuídos.” ([MOREIRA, 2013](#))

O Node.JS foi criado em 2009 por Ryan Dahl, essencialmente como uma forma de executar Javascript fora do contexto de um *browser*, isso para que o Javascript pudesse trabalhar como linguagem de uso geral. Em resposta a esta necessidade, o Node.JS seguiu a filosofia assíncrona que já permeava o Javascript dentro do *browser* [PFÜTZENREUTER \(2016\)](#).

#### 3.2.5.2 Javascript

A linguagem Javascript foi criada pela Netscape para funcionar nas páginas [HTML](#) de forma interativa.

[ALVAREZ \(2013\)](#), define que:

“a linguagem Java é usada na criação de objetos e os chamados Applets

(aplicativos que são executados em uma página web). Já a JavaScript não cria Applets ou aplicativos independentes. Em sua forma mais comum hoje, JavaScript reside dentro de documentos HTML e pode fornecer níveis de interatividade com páginas da Web que não podem ser conseguidos com HTML simples.”

Apesar de ter sintaxe análoga à do Java, o Javascript possui conceito e uso diferentes. O Javascript disponibiliza tipagem dinâmica, é interpretada ao invés de compilada, possui excelentes *frameworks* ferramentas padrões para listagem e oferecem suporte a expressões regulares [JAVA \(2013\)](#).

### **3.2.5.3 NW.js**

O NW.js é uma aplicação baseada em *Chromium* e Node.js, no qual é possível desenvolver “aplicativos nativos para Windows, Linux e Mac, usando tecnologias web e usufruindo dos pacotes do Node.js.” [ZONTA \(2017\)](#).

NW.js permite que você ligue todos os módulos Node.js diretamente do DOM e ative uma nova maneira de escrever aplicativos com todas as tecnologias da Web.

Tem como características: suporte total para os recursos no navegador, suporte completo para APIs do Node.js e todos os módulos de terceiros, além de ser uma nova forma de escrever aplicativos nativos usando tecnologias da web como por exemplo [HTML](#) e [CSS](#) ([ZONTA, 2017](#)).

### **3.2.5.4 AngularJS**

AngularJS é um *framework* Javascript *open-source*, cultivado pelo Google, que auxilia na execução de aplicações de páginas únicas [WALTENBERG \(2017\)](#).

É um *framework* estrutural para aplicações web dinâmicas, que permite que se utilize [HTML](#) como linguagem, admitindo expandir a sintaxe do [HTML](#) para expressar os componentes da sua aplicação, de forma clara e sucinta. A ligação de dados e a injeção de dependência da AngularJS eliminam muito os códigos além de que todo os processos são realizados dentro do navegador [WALTENBERG \(2017\)](#).

### **3.2.5.5 SQLite**

SQLite é uma biblioteca desenvolvida em C que implementa um banco de dados relacional, transacional e sem a necessidade de ser cliente/servidor [S.T., PATIL e PATIL \(2015\)](#).

Esta biblioteca é distribuída junto com a aplicação e fica encarregada de ler e escrever do arquivo de banco de dados que fica alojado no disco. Ela implementa a linguagem *Structured Query Language (SQL)* de banco de dados e suporta a diversas

linguagem de programação como C, C++, BASIC, C#, Python, Java e Delphi [S.T., PATIL e PATIL \(2015\)](#).



## 4 IMPLEMENTAÇÃO

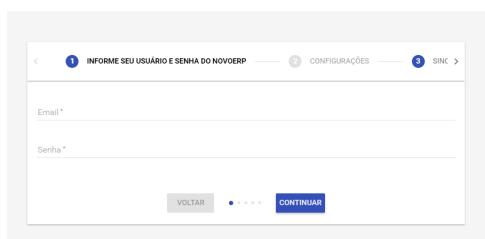
A etapa de implementação do software é o processo no qual é criado um executável do sistema a partir de sua especificação. As atividades de projeto de sistema e sua implementação são intercaladas. O projeto de software é uma atividade criativa em que é identificado as partes do software e seus relacionamentos com base nos requisitos. Já a implementação é o processo de concretização deste projeto SOMMERVILLE (2011).

Neste trabalho são apresentadas as telas do sistema e um descriptivo de como foi elaborada cada interface.

### 4.1 Tela de Configuração

Na primeira execução do **PDV**, ou quando o usuário escolhe reconfigurar o **PDV**, é apresentada a tela de configuração conforme a Figura 13.

Figura 13 – Tela de configuração.



Fonte: Elaborado pelo autor

A tela de configuração é dividida em 5 passos, executados em ordem e com as seguintes interações:

1. O operador deve informar o *e-mail* e senha de utilização do software NovoERP. Com estes dados o **PDV** acessa via **API** o software e busca as lojas cadastradas na conta do e-mail informado e prossegue para o próximo passo. Caso o operador informe um e-mail ou senha inválidos o sistema alerta o usuário que os dados estão inválidos com uma mensagem de erro.
2. No segundo passo o operador deve selecionar a loja no qual o **PDV** pertence e o número do **PDV** que está sendo configurado. A seleção da loja serve para que o **PDV** capture via **API** todos os dados de produtos e fiscais relacionados aquela loja para realização das vendas. Já o número do **PDV** serve para identificar o caixa no qual ele está instalado.
3. No terceiro passo o **PDV** faz a sincronia dos dados via **API**. Em *background* ele realiza a consulta das informações necessárias e salva no seu banco de dados local.

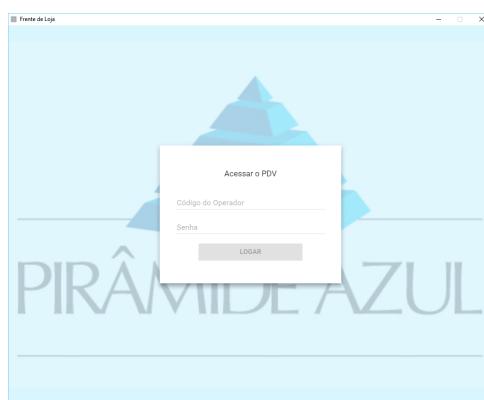
Enquanto é realizada está sincronia para o usuário é exibida uma animação de *loading*.

4. Após a sincronia de dados ser concluída a configuração vai automaticamente para o passo 4 no qual solicita o usuário a configuração da impressora que o **PDV** deve utilizar. É solicitado ao operador o nome da impressora que está instalada no computador no qual o **PDV** está sendo executado e também qual o formato de impressão deve ser utilizado. Esse formato serve para especificar qual o formato de dados a ser enviado para a impressora. Se ela for uma impressora fiscal o operador selecionar o modelo da mesma na lista apresentada, ou então se for uma impressora comum ele seleciona o formato Portable Document Format (**PDF**) que faz com que o **PDV** envie o cupom como um **PDF** para a impressora.
5. No último passo é exibida a mensagem para o usuário de que o **PDV** está configurado e que ele pode começar a utilizar.

## 4.2 Tela de *Login*

A Figura 14 mostra a tela de *login* do sistema. É uma tela simples, contendo um formulário de *login* e senha no qual o usuário deve preencher para obter acesso ao sistema. Caso o mesmo informe um usuário ou senha inválidos o sistema exibe uma mensagem de erro alertando o usuário do erro.

Figura 14 – Tela de login.



Fonte: Elaborado pelo autor

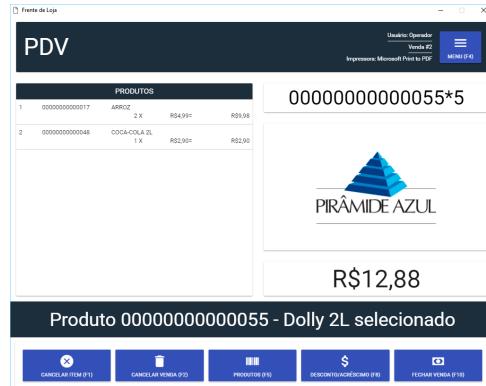
Após informar os dados corretos, o sistema valida o acesso do usuário e carrega rapidamente informações do **PDV**, sendo redirecionado para a tela de realização de venda.

## 4.3 Tela Realizar Venda

A Figura 15 exibe a principal tela do sistema onde é realizada uma venda. Nesta interface o usuário pode adicionar e remover produtos da venda. Esta tela também conta

com informações sobre a venda, como por exemplo o nome do usuário que está realizando a venda, o código e a impressora configurada na parte superior, subtotal da venda e uma área especial para exibir a última ação realizada pelo usuário.

Figura 15 – Tela de realização de venda. Adicionando um produto na venda



Fonte: Elaborado pelo autor

O usuário adiciona produtos digitando o código dela acrescido de um \* (asterisco) mais a quantidade desejada do produto. Após adicionado um produto o sistema calcula o subtotal da venda conforme a Figura 16.

Figura 16 – Tela de realização de venda. Após adicionar um produto na venda.

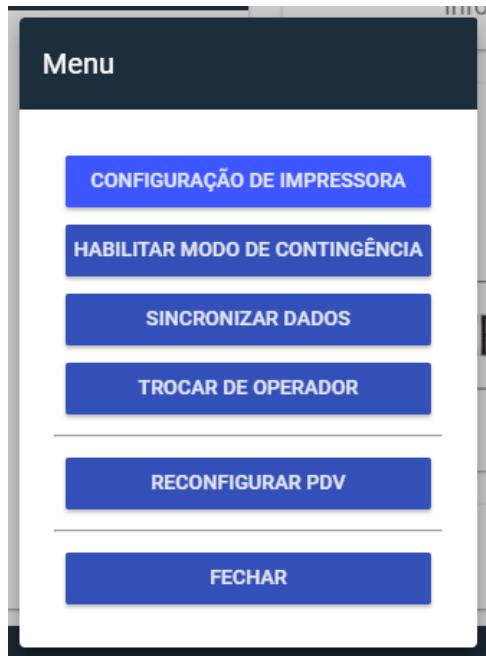


Fonte: Elaborado pelo autor

#### 4.4 Menu (F4)

O menu de opções é acessado através do botão posicionado no topo direito da página ou com a tecla F4 do teclado. A Figura 17 mostra o menu de opções.

Figura 17 – Menu de opções



Fonte: Elaborado pelo autor

O menu possui as seguintes opções:

**Configuração de Impressora:** O **PDV** apresenta uma nova tela solicitando ao operador que selecione o nome da impressora e o formato de impressão igual ao apresentado na tela de configuração.

**Habilitar modo de contingência:** Esta opção é exibida quando o **PDV** não está operando em modo de contingência. Ao ser selecionar esta opção o **PDV** solicita confirmação do usuário para prosseguir. Se o usuário desejar continuar então o **PDV** primeiro verifica se possui conexão com a **API** do NovoERP, se sim ele exibe uma mensagem que possui conexão com a Internet e interrompe o processo, se não possui ele habilita o modo de contingência do **PDV**. No modo de contingência o **PDV** realiza todo o processamento de emissão de nota fiscal *OFF-LINE* e salva o resultado no banco de dados para enviar posteriormente.

**Desabilitar modo de contingência:** Esta opção é exibida quando o **PDV** está operando em modo de contingência. Quando o operador desabilita o modo de contingência o **PDV** verifica se possui acesso com a **API** do NovoERP e se confirmado ele desabilita o modo e realiza o envio de todas as notas geradas para o NovoERP. Conforme exibido na Figura 18 é exibida uma lista com as notas geradas, e em cada nota o seu status de transmissão. Quando este processo é realizado o uso do **PDV** é bloqueado até o envio de todas as notas geradas.

**Sincronizar dados:** É feito o download das informações de produtos e fiscais do NovoERP.

Está opção existe para a situação no qual o operador atualiza algum dado no ERP e deseja atualizar essa informação no **PDV**.

**Trocar de operador:** É deslogado o operador atual e redirecionado para a tela de login para que o novo operador informe seus dados de acesso.

**Reconfigurar PDV:** Serve para apagar todas as informações do banco de dados do **PDV** e possibilitar que o operador configure novamente com novas informações. É utilizada na situação em que o **PDV** muda de loja ou de número de cadastro.

**Fechar:** Fecha o **PDV** encerrando a aplicação.

Figura 18 – Tela com a lista de notas geradas em contingência

Enviando NFC-e geradas em contingência		
Número Venda	Data	Status
1	24/05/2018	Autorizada
2	24/05/2018	Autorizada
3	24/05/2018	Autorizada

Fonte: Elaborado pelo autor

## 4.5 Cancelando Produto (F1)

Para cancelar um produto específico da venda o usuário deve utilizar a tecla de atalho F1 ou utilizar o seu botão de ação, com isso é exibido uma tela no qual deve ser informado o número do produto na venda, conforme a Figura 19.

Figura 19 – Tela para cancelar item da venda.

Cancelar item

Informe abaixo o número do item que deseja cancelar:

OK

Fonte: Elaborado pelo autor

Ao informar o número do produto e teclar *enter* o produto é removido da venda e o usuário retorna à tela de adicionar produto conforme a Figura 20.

Figura 20 – Após remover um item da venda.



Fonte: Elaborado pelo autor

## 4.6 Cancelando Cupom (F2)

Para cancelar a venda e reiniciar o processo o usuário utiliza a tecla F2. Ao clicar o sistema exibe um alerta questionando o usuário se ele deseja continuar com a ação conforme Figura 21.

Figura 21 – Alerta de cancelamento de cupom.



---

Fonte: Elaborado pelo autor

Caso o usuário continue com a ação, o sistema considera a venda atual como cancelada, cria um novo cupom com a numeração sequencial e reinicia o produto de venda.

A Figura 22 mostra o sistema após uma venda cancelada.

Figura 22 – Após cancelar a venda.



Fonte: Elaborado pelo autor

## 4.7 Lista de Produtos (F5)

A Figura 23 exibe a lista de produtos para pesquisa que é exibida após o usuário clicar na tecla de atalho F5. Essa tela possui um campo no qual o usuário pode pesquisar um produto pelo seu código ou pela sua descrição.

Figura 23 – Tela de Pesquisa de Produto.

Pesquisar Produto		
Pesquisar		
Código	Produto	Valor Unit.
0000000000017	Arroz	R\$4,99
0000000000079	Carneíço	R\$1,00
0000000000048	Coca-Cola 2L	R\$2,90
0000000000055	Dolly 2L	R\$4,00
0000000000086	Farinha	R\$1,00
0000000000024	Feijão	R\$2,00
0000000000031	Macarrão	R\$2,00
0000000000062	Pão	R\$10,00
0000000000093	Uva	R\$2,00

Fonte: Elaborado pelo autor

Ao digitar o termo desejado, o sistema automaticamente filtra no banco de dados e atualiza a lista de produtos. Ao teclar as teclas cima e baixo do teclado, o usuário pode navegar pelos itens. Essa navegação é demarcada pelo fundo cinza na linha do produto selecionado, conforme exibido pela Figura 24.

Figura 24 – Navegando pelos produtos filtrados.

Pesquisar Produto		
Pesquisar		
Código	Produto	Valor Unit.
0000000000017	Arroz	R\$4,99
0000000000079	Carneíço	R\$1,00
0000000000048	Coca-Cola 2L	R\$2,90
0000000000055	Dolly 2L	R\$4,00
0000000000086	Farinha	R\$1,00
0000000000024	Feijão	R\$2,00
0000000000031	Macarrão	R\$2,00
0000000000062	Pão	R\$10,00
0000000000093	Uva	R\$2,00

Fonte: Elaborado pelo autor

Após selecionado o produto desejado o usuário pode clicar na tecla *enter* para selecionar o produto e fechar esta tela. Ao fechar o usuário retorna para a tela de adição de produtos com o código de produto já digitado no campo de adição de produtos, visto na Figura 25, faltando o usuário informar a quantidade desejada.

Figura 25 – Após selecionar um produto da lista.

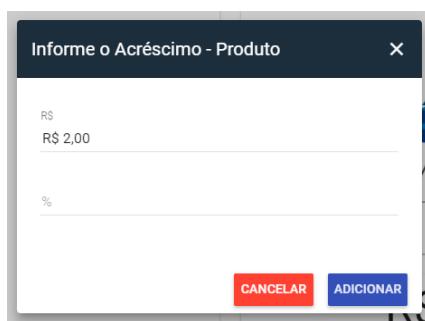


Fonte: Elaborado pelo autor

#### 4.8 Tela de Desconto/Acréscimo (F8)

A função de Desconto/Acréscimo disponibiliza para o operador a possibilidade de adicionar um desconto ou acréscimo a um produto que vai ser inserido na venda. Ao utilizar a função primeiro é exibido uma tela que permite o operador selecionar se deseja informar um desconto ou acréscimo. Após feita a seleção o usuário pode então informar o valor da operação a ser realizada. Este valor pode ser em Real ou porcentagem, conforme visto na Figura 26.

Figura 26 – Campos de aplicação de desconto/acréscimo



Fonte: Elaborado pelo autor

Após adicionado o desconto/acréscimo e inserido o produto, na lista de produtos adicionados a venda é destacado o valor adicionado em uma linha adicional logo abaixo

do nome do produto, conforme mostra a Figura 27.

Figura 27 – Destaque do valor da operação de acréscimo realizada no produto



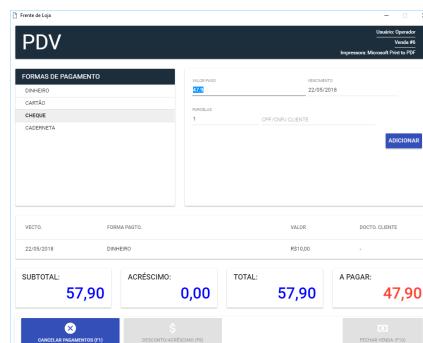
Fonte: Elaborado pelo autor

#### 4.9 Tela de Pagamento (F10)

Após o operador adicionar todos os produtos da venda ele deve prosseguir para a tela de pagamento no qual ele informa os métodos de pagamento que o cliente vai utilizar para realizar a venda. Nesta tela o operador, exibida na Figura 28, possui uma lista de formas de pagamento no qual ele pode selecionar, todas previamente cadastradas no NovoERP. É possível também que o operador adicione um desconto ou acréscimo a venda utilizando a mesma função disponibilizada na tela de produtos.

Ao selecionar a forma de pagamento o operador deve informar o valor do pagamento a ser efetuado para a forma selecionada e também informar algumas informações necessárias caso a forma de pagamento permita. Estas informações adicionais são todas parametrizadas no cadastro da forma de pagamento no NovoERP e são utilizadas em relatórios de venda do sistema. Após o operador quitar a venda, isto é, quando o valor total de pagamentos adicionado é igual ou superior ao total da venda, o PDV encerra a venda executando o processo de Finalizar Venda descrito na seção 3.2.4.

Figura 28 – Tela de Pagamento



Fonte: Elaborado pelo autor

Após criação da NFC-e, envio e retorno do arquivo o PDV imprime a DANFE caso uma impressora esteja configurada. A Figura 29 exibe uma DANFE impressa pelo PDV em formato PDF.

Figura 29 – DANFE de NFC-e impressa

**MACDEL NFCE PR**  
 CNPJ:05.361.065/0001-86  
 Avenida Parana, 3818 - Zona I  
 Umuarama-PR - 87501030

---

**DOCUMENTO AUXILIAR DA NOTA FISCAL DE CONSUMIDOR ELETRÔNICA**

ITEM	COD	DESC	QTD	UN	VL UNIT.	VL TOTAL
001	0000000000017	NOTA FISCAL EMITIDA EM AMBIENTE DE HOMOLOGACAO - SEM VALOR FISCAL				
002	0000000000079	Camarão				
			100	UN	X 4.99 =	499.00
			5	UN	X 1.00 =	5.00
			QTD. TOTAL DE ITENS		2	
			VALOR TOTAL R\$		504.00	
			VALOR A PAGAR R\$		504.00	
			DINHEIRO		100.00	
			CHEQUE		404.00	
			Consulte pela Chave de Acesso em:			
			www.sped.fazenda.pr.gov.br			
			4118 0505 3610 6500 0186 6506 4000 0131			
			4110 0000 0074			
			CONSUMIDOR NAO IDENTIFICADO			
			NFC-e No.:			
			000013141			
			Série:			
			64			
			22/05/2018 19:19			
			Protocolo de Autorizacao:			
			141180000581706			
			Data de autorizacao:			
			22/05/2018 19:19			
			EMITIDA EM AMBIENTE DE HOMOLOGACAO - SEM VALOR FISCAL			
						
			Tributos Incidentes Lei Federal			
			12.741/12			
			Federal: R\$21.62 Estadual: R\$90.72			
			Municipal: R\$0.00 - Fonte: IBPT			
			PDV.: 64		Série: 64	
			DATA: 22/05/2018 19:19:42 Seq.: 000007			

Fonte: Elaborado pelo autor

## 5 TESTES

A realização de testes é feita para mostrar que um programa está executando o que foi proposto e para identificar problemas antes do uso. Os resultados dos testes são analisados à procura de erros, anomalias ou informações sobre os atributos não funcionais do sistema [SOMMERVILLE \(2011\)](#).

Com isso entende-se que o processo de testes possui dois objetivos distintos. O primeiro objetivo é demonstrar ao desenvolvedor e também ao cliente que o software desenvolvido está operando conforme as suas especificações com isso validando seus requisitos. O segundo objetivo é o de descobrir situações em que o software funciona fora de suas especificações, de maneira incorreta, consequência de defeitos de software [SOMMERVILLE \(2011\)](#).

### 5.1 Teste de Compatibilidade

Teste de compatibilidade, também conhecido como teste de interoperabilidade, visa analisar o comportamento da aplicação nos diversos tipos de hardwares disponíveis. Este tipo de teste é necessário para garantir que uma aplicação é executada em todos os dispositivos e versões do sistema operacional para o qual foi projetada [DELAMARO, JINO e MALDONADO \(2016\)](#).

Nesta etapa foi executado um teste de compatibilidade com o objetivo de identificar se a aplicação é executada nos três principais sistemas operacionais do mercado.

#### 5.1.1 SO Windows

O teste inicialmente não foi satisfatório. Quase todas funcionalidades funcionaram perfeitamente com exceção da funcionalidade de impressão em [PDF](#) da [DANFE](#) de [NFC-e](#). Foi identificado que o módulo de impressão utilizado no [PDV](#) não possui a possibilidade de impressão em [PDF](#) no sistema operacional Windows. Foi então proposto a solução de utilizar um executável de terceiros chamado PDFtoPrinter. Este programa imprime, via chamada de terminal, um arquivo [PDF](#) para a impressora especificada. Portanto somente para o [SO Windows](#) e quando o formato for [PDF](#) o [PDV](#) efetua uma chamada desse programa para imprimir o [PDF](#) da [DANFE](#). Após esta correção os testes foram executados novamente e foram satisfatórios, com todas as funcionalidades atendendo as especificações. O teste foi realizado em uma instalação do Windows 10 versão 1709. A Figura 30 apresenta a execução do [PDV](#) no sistema Windows.

Figura 30 – PDV sendo executado no SO Windows.



Fonte: Elaborado pelo autor

### 5.1.2 SO Linux

O teste foi satisfatório. Todas as funcionalidades atenderam às especificações. Foi identificado que o NW.JS possui a dependência da biblioteca libgconf-2-4, portanto é necessário que esta biblioteca esteja instalada na máquina. Este teste foi realizado em uma instalação do SO Ubuntu, versão 18.04, feita em uma máquina virtual utilizando o software VMWare. Na Figura 31 é demonstrado o PDV sendo executado.

Figura 31 – PDV sendo executado no SO Ubuntu



Fonte: Elaborado pelo autor

### 5.1.3 SO macOS

O teste inicialmente não foi satisfatório. Foi identificado que o pacote gerado não era executado neste sistema operacional e não era relatado no *log* do sistema operacional o motivo pelo qual o software não executava. Na tentativa de identificar a causa do problema, foi possível executar o programa via terminal. Com isso foi obtida a mensagem de erro do problema e pesquisou-se na Internet por uma solução. Foi então identificado que o problema era causado pela ausência de um *symlink* dentro da estrutura do executável. Como o pacote era gerado na máquina de desenvolvimento que utiliza o SO Windows, esse *symlink* era perdido pois o Windows não possui suporte a esta funcionalidade. A solução encontrada foi então gerar o pacote no próprio sistema operacional. Com o pacote gerado no macOS, os testes foram executados novamente e satisfatórios, com todas as funcionalidades atendidas. O teste foi realizado na instalação do SO macOS versão 10.12.3 Sierra feita em uma máquina virtual utilizando o software VMWare. Na Figura 32 é exibido

o PDV sendo executado neste SO.

Figura 32 – PDV sendo executado no SO macOS.



Fonte: Elaborado pelo autor



## 6 CONCLUSÃO

Neste trabalho foi desenvolvido um **PDV** que emite **NFC-e** utilizando o NW.JS. O objetivo foi chegar a uma aplicação que execute sem problemas nos três principais sistemas operacionais a partir de um só código base, sem precisar utilizar uma linguagem de programação específica para cada plataforma e que seja de fácil utilização pelo usuário final.

Este programa por exemplo resolveria o problema enfrentado pela empresa Avanço Informática, mesma desenvolvedora do NovoERP. Ela possui o produto de **PDV** chamado Frente de Loja que roda em Windows e Linux. Porem para cada **SO** é um código base diferente, portanto ao implementar alguma funcionalidade ou correção é necessário ser realizada duas vezes, para cada plataforma. Outro problema seria a dificuldade de configuração do mesmo pelo cliente final devido à complexidade de informações necessárias, o que sempre ocasiona um chamado no setor de suporte da empresa.

Com a utilização da solução apresentada estes problemas foram resolvidos facilmente pois evita o problema de manutenção de dois códigos fonte base e também minimiza a parte de configuração por parte do usuário final.

Dificuldades foram encontradas, principalmente com relação às peculiaridades de cada **SO**, mas foram encontradas soluções para essas situações.

Assim, este trabalho pode contribuir com o operador de caixa servindo como uma ferramenta que atender às suas necessidades básicas para realizar a venda e emitir a **NFC-e** sem problemas.

### 6.1 Dificuldades Encontradas

Desde o início deste projeto foram encontradas dificuldades no desenvolvimento da aplicação, listadas a seguir.

Foi identificado que depurar uma aplicação no NW.JS quando desenvolvido no Windows é um problema pois a janela de depuração não era iniciada junto com a aplicação o que fazia a identificação de erros no início difícil. Para isso era preciso ler o arquivo de log gerado pelo NW.JS na pasta do usuário do sistema.

Os módulos de impressão e do banco de dados SQLite precisam ser compilados para cada **SO** de destino, portanto foi necessário configurar cada máquina de Linux e macOS com um ambiente de desenvolvimento para gerar os módulos de cada versão. Outro ponto identificado referente aos módulos é que caso a versão do NW.JS seja atualizada será necessário a compilação de uma nova versão de cada módulo para cada **SO** pois o módulo é compilado para ser usado naquela versão específica do NW.JS.

Outro problema foi a impressão em **PDF** no Windows. Conforme descrito na

seção 5.1.1 foi identificado o problema devido ao módulo de impressão não dar suporte a PDF então implementar a impressão de PDF no Windows foram testadas várias soluções. Foi realizada a tentativa de utilizar diversos módulos para gerar um arquivo EMF a partir do PDF que não teve sucesso pois a compilação de cada módulo era problemática e de difícil configuração. Foi então utilizado um programa de terceiro para converter o PDF para EMF mas ele gerava um arquivo com problema que não era impresso pelo Windows. Então se chegou a solução de utilizar o programa que imprimia direto no Windows o PDF.

Por último foi detectado o problema de criação do pacote final no macOS já descrito na seção 5.1.3. Como o pacote não retornava uma mensagem de erro com a descrição do problema foi gasto um bom tempo para tentar identificar o problema e corrigi-lo.

Apesar das dificuldades listadas acima, o projeto como um todo mostrou-se bem satisfatório, atendendo aos requisitos básicos levantados, e pode contribuir com o estado da arte servindo de base para uma implantação em empresas da área.

## 6.2 Trabalhos Futuros

Foram levantadas diversas funcionalidades na etapa de requisitos que não foram utilizadas para esta versão do software. Funcionalidades como *login* de usuário para cada ação, opções de validação de produto e também de formas de pagamento entre outras.

A ideia é de que o software entre em uso e seja posto em um processo de desenvolvimento incremental no qual seriam corrigidos problemas encontrados e sempre implementadas essas funcionalidades levantadas e também novas funcionalidades.

## REFERÊNCIAS

- ALVAREZ, M. A. **O que é JavaScript.** [S.l.], 2013. Disponível em: <<http://www.criarweb.com/artigos/184.php>>. Acesso em: 01 de setembro de 2017.
- ASSAF, A. **Estrutura e análise de balanços: um enfoque econômico-financeiro.** 7. ed. São Paulo: Atlas, 2002. 320p.
- BAKER, B. **Business Modeling with UML: The Light at the End of the Tunnel.** [S.l.], 2001. Disponível em: <<http://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.577.7437&rep=rep1&type=pdf>>. Acesso em: 02 de maio de 2018.
- DELAMARO, M.; JINO, M.; MALDONADO, J. **Introdução ao Teste de Software.** 2. ed. Rio de Janeiro: Elsevier Brasil, 2016. 448p. ISBN 978-85-3528-352-5.
- FAZENDA, M. da. **AJUSTE SINIEF DO MINISTÉRIO DA FAZENDA.** [S.l.], 2005. Disponível em: <<https://www.confaz.fazenda.gov.br>>. Acesso em: 29 de março de 2018.
- FAZENDA, M. da. **Manual de Especificações Técnicas do DANFE NFC-e e QR Code.** [S.l.], 2017. Disponível em: <<https://www.nfe.fazenda.gov.br>>. Acesso em: 29 de março de 2018.
- GOLPI, G. **Entenda tudo sobre a frente de caixa (PDV).** [S.l.], 2017. Disponível em: <<https://blog.softensistemas.com.br/entenda-tudo-sobre-a-frente-de-caixa-pdv/>>. Acesso em: 05 de setembro de 2017.
- GROSSO, S. D. E. E. F. D. M. **SECRETARIA DE ESTADO E FAZENDA DE MATO GROSSO.** [S.l.], 2018. Disponível em: <<http://www.sefaz.mt.gov.br>>. Acesso em: 29 de março de 2018.
- JAVA. **Qual é a diferença entre o JavaScript e o Java.** [S.l.], 2013. Disponível em: <[http://java.com/pt\\_BR/download/faq/java\\_javascript.xml](http://java.com/pt_BR/download/faq/java_javascript.xml)>. Acesso em: 01 de setembro de 2017.
- KARINE. **NFC-e: calendário de obrigatoriedade em todos os estados do Brasil.** [S.l.], 2018. Disponível em: <<https://enotasgw.com.br/blog/nfc-e-calendario-de-obrigatoriedade-estados/>>. Acesso em: 01 de abril de 2018.
- LUKIANOCENKO, M. R. **Superhiper.** Rio de Janeiro: [s.n.], 2006. 200p.
- MOREIRA, R. **O que é Node.js?** [S.l.], 2013. Disponível em: <<http://nodebr.com/o-que-e-node-js/>>. Acesso em: 01 de setembro de 2017.
- PAULA FILHO, W. d. P. **Engenharia de Software: fundamentos, métodos e padrões.** 3. ed. Rio de Janeiro, RJ: [s.n.], 2009. ISBN 978-85-216-1650-4.
- PDVEND – SISTEMA DE VENDAS PARA PDV E CAIXA DE LOJA. **PDVend.** [S.l.], 2018. Disponível em: <<https://www.pdvend.com.br/>>. Acesso em: 01 de maio de 2018.
- PFÜTZENREUTER, E. **Node.js: ferramenta e filosofia.** [S.l.], 2016. Disponível em: <<https://epxx.co/artigos/nodejs1.html>>. Acesso em: 01 de setembro de 2017.

PWI WEV STUDIO. **O que é sistema PDV? O que é sistema de retaguarda?** [S.l.], 2017. Disponível em: <<http://vixenpdv.pwi.com.br/o-que-e-sistema-pdv-o-que-e-sistema-de-retaguarda/>>. Acesso em: 01 de maio de 2018.

PWI WEV STUDIO. **Sistema de Frente de Caixa, escolha o melhor! Sistema PDV Vixen. Home.** [S.l.], 2017. Disponível em: <<http://vixenpdv.pwi.com.br/>>. Acesso em: 01 de maio de 2018.

REDAÇÃO PAGAMENTO.ME. **PDVend, o ponto de venda 3.0, é finalista do Sebrae Like a Boss!** [S.l.], 2016. Disponível em: <<https://pagamento.me/pdvend-o-ponto-e-venda-3-0-e-finalista-do-sebrae-like-boss/>>. Acesso em: 01 de maio de 2018.

SOFTEN SISTEMAS. **Software Nota Fiscal do Consumidor Eletrônica - NFC-e | Soften Sistemas.** [S.l.], 2018. Disponível em: <<https://www.softensistemas.com.br/produtos/software-pdv-nfce/>>. Acesso em: 01 de maio de 2018.

SOMMERVILLE, I. **Engenharia de software.** 9. ed. São Paulo: Pearson Education do Brasil Ltda, 2011. 529p. ISBN 978-85-7936-108-1.

SOUZA, A. A. et al. Controle de gestão em organizações hospitalares. **REGE Revista de Gestão**, v. 16, n. 3, p. 15–29, 2009. ISSN 2177-8736. Disponível em: <<http://www.revistas.usp.br/rege/article/view/36675>>.

SPED. **Manual de Orientação do Contribuinte - versão 5.0 - Março 2012.** [S.l.], 2012. Disponível em: <<http://sped.rfb.gov.br>>. Acesso em: 30 de março de 2018.

S.T., B.; PATIL, T.; PATIL, P. Sqlite: Light database system. **IJCSCMC**, v. 4, n. 4, 2015. Disponível em: <<https://ijcsmc.com/docs/papers/April2015/V4I4201599a35.pdf>>.

TECNOSPEED. **Dossiê NFC-e TecnoSpeed v. 2.0.** [S.l.], 2013. Disponível em: <<http://tsdn.tecnospeed.com.br/group/ebook-nfc-e>>. Acesso em: 30 de março de 2018.

TURKLE, S. **O Segundo Eu - os computadores e o espírito humano.** Lisboa: [s.n.], 1989.

WALTENBERG, R. **AngularJS: O que é e porquê utilizar.** [S.l.], 2017. Disponível em: <<http://blog.algaworks.com/o-que-e-angularjs>>. Acesso em: 06 de setembro de 2017.

WERNER, A.; SEGRE, L. M. **Uma análise do segmento de supermercados: estratégias, tecnologias e emprego.** [S.l.], 2002. Disponível em: <<http://www.senac.br/informativo/bts/281/boltec281e.htm>>. Acesso em: 01 de outubro de 2017.

ZONTA, J. A. **Aplicativos Desktop com NW.js – Node Webkit – Introdução.** [S.l.], 2017. Disponível em: <<https://tableless.com.br/aplicativos-desktop-com-nw-js-node-webkit-introducao>>. Acesso em: 06 de setembro de 2017.

## **Apêndices**



## APÊNDICE A – REPOSITÓRIO DO CÓDIGO FONTE

URL de acesso ao repositório que contém o código fonte do software desenvolvido.

<<https://bitbucket.org/michaelfamarques/pdv-nwjs/>>