GOVERNO DO ESTADO DO RIO DE JANEIROSECRETARIA DE ESTADO DE CIÊNCIA E TECNOLOGIAFUNDAÇÃO DE APOIO À ESCOLA TÉCNICAFACULDADE DE EDUCAÇÃO TECNOLÓGICA DO ESTADO DO RIO DEJANEIRO FAETERJ/PARACAMBI

CURSO SUPERIOR DE TECNOLOGIA EM SISTEMAS DE INFORMAÇÃOTRABALHO DE CONCLUSÃO DE CURSO

**RAFAEL SILVA PATRICIO**

**DESENVOLVIMENTO DE SOFTWARE: UM SISTEMA DE VENDAS PARA MODELO DE NEGÓCIOS B2B**

**PARACAMBI, RJ  
2019**

**RAFAEL SILVA PATRICIO**

**DESENVOLVIMENTO DE SOFTWARE: UM SISTEMA DE VENDAS PARA MODELO DE NEGÓCIOS B2B**

Trabalho de Conclusão de Curso (TCC)  
apresentado ao Curso Superior de Tecnologia  
em Sistemas de Informação da Faculdade de Educação Tecnológica do Estado do Rio de Janeiro, Campus Paracambi – FAETERJ / Paracambi, como requisito parcial para a obtenção do grau de Tecnólogo em Sistemas de Informação.

**Orientador**: Prof. Artur Sérgio Lopes

**PARACAMBI, RJ  
2019**

|  |
| --- |
|  |

**RAFAEL SILVA PATRICIO**

**DESENVOLVIMENTO DE SOFTWARE: UM SISTEMA DE VENDAS PARA MODELO DE NEGÓCIOS B2B**

Trabalho de Conclusão de Curso (TCC)  
apresentado ao Curso Superior de Tecnologia  
em Sistemas de Informação da Faculdade de Educação Tecnológica do Estado do Rio de Janeiro, Campus Paracambi – FAETERJ / Paracambi, como requisito parcial para a obtenção do grau de Tecnólogo em Sistemas de Informação.

**TRABALHO DE CONCLUSÃO DE CURSO APROVADO EM 09/07/2019**

**BANCA EXAMINADORA:**

**\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_**Prof. Doutor Artur Sérgio Lopes  
Faculdade de Educação Tecnológica do Estado do Rio de Janeiro  
FAETERJ/Paracambi  
Orientador

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_  
Prof. Doutor Eluã Ramos Coutinho  
Faculdade de Educação Tecnológica do Estado do Rio de Janeiro  
FAETERJ/Paracambi

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_  
Prof. Especialista José Augusto Teixeira de Lima Júnior  
Instituto Federal do Rio de Janeiro  
IFRJ/Engenheiro Paulo de Frontin

**RESUMO**

PATRICIO, RAFAEL. **Desenvolvimento de Software:** Um sistema de  
vendas para modelo de negócios B2B.2019. Trabalho de Conclusão de Curso (Curso Superior de Tecnologia em Sistemas de Informação). Faculdade de Educação Tecnológica do Estado do Rio de Janeiro, campus Paracambi – FAETERJ/Paracambi. 2019.

Este trabalho apresenta o desenvolvimento de um software para gerenciamento de estoque e realização de vendas em um modelo de negócios entre empresas.   
A partir da observação de vendedores visitando clientes, especificamente aqueles que oferecem seus produtos para empresas e percebendo a manualidade do seu processo em muitos dos casos, decidiu-se pela criação de uma solução que atendesse aos praticantes desse nicho de negócios. Definiu-se a utilização do Java como linguagem de programação, complementado pelo Hibernate/JPA e Spring no back-end e HTML5, CSS3, Javascript e Thymeleaf no front-end. O software possibilitará a criação e controle de estoque, execução de vendas, envio do resumo da venda por e-mail, geração de relatório, resumo financeiro com gráficos e separação de responsabilidades através de níveis de acesso de usuários. Dessa forma então, assistindo tanto os vendedores durante o processo de vendas, à empresa nos processos gerenciais e consequentemente os clientes, com mais agilidade e informação.

Palavras-Chave: Java; Spring; Vendas; Negócios; Informação

**ABSTRACT**

PATRICIO, Rafael. **Software Development:** A sales system for B2B business model. 2019. Final Paper (Course of Information Systems) – Technological Education College of The State of Rio de Janeiro, Paracambi campus – FAETERJ/Paracambi. 2019

This work presents the software development for inventory management and sales in an intercompany business model. From observing salespeople visiting customers, specifically those who offer their products to companies, and realizing the manuality of their process in many cases, it was decided to create a solution that would fit the practitioners of this business niche. The use of Java as a programming language was defined, complemented by Hibernate / JPA and Spring on the back-end and HTML5, CSS3, Javascript and Thymeleaf on the front-end. The software will enable inventory creation and control, sales execution, email sales summary, report generation, graphical financial summary, and separation of responsibilities across user access levels. Thus, assisting both salespeople during the sales process, the company in the management processes and consequently the customers, with more agility and information.

Keywords: Java; Spring; Sales; Business; Information.

**LISTA DE FIGURAS**

Figura 1 - Interface do programa StarUML …………………………………………..Figura 1 – Diagrama de Caso de Uso …………………………….…………...……… 18  
Figura 2 – Diagrama de Classes …………………………………….……..….……… 19  
Figura 3 – Modelo Entidade-Relacionamento ………………………………………... 20  
Figura 4 – Identidade Visual da Empresa Fictícia Mago Digital …………………….. 21  
Figura 5 – Parte separada da identidade visual ………………………...……………... 22  
Figura 6 – Parte textual da identidade visual ………………………..………..…….… 22  
Figura 7 – Tela de login ………………………………………………….…………… 23  
Figura 8 – *Dashboard* ou tela inicial ………...………………………….……………. 24  
Figura 9 – Cadastro de produto ………………………………………….…………… 25  
Figura 10 – Cadastro de cliente …………………………………………….………… 25  
Figura 11 – Cadastro de subcategoria …………………………………….…………... 26  
Figura 12 – Cadastro de marca …………………………………………….…………. 26  
Figura 13 – Cadastro de cidade …………………………………………….………… 27  
Figura 14 – Cadastro de usuário …………………………………………….………... 27  
Figura 15 – Consulta de produtos ………………………………………….…………. 28  
Figura 16 – Consulta de clientes ……………………………………………..……….. 28  
Figura 17 – Consulta de marcas ……………………………………………..………... 29  
Figura 18 – Consulta de subcategorias …………………………………….…………. 29  
Figura 19 – Consulta de cidades ………………………………………………….…... 30  
Figura 20 – Consulta de usuários ……………………………………………….……. 30  
Figura 21 – Nova venda (carrinho vazio) …………………………………….………. 31  
Figura 22 – Nova venda (Itens no carrinho) ………………………………...….…….. 31  
Figura 23 – Consulta de vendas ……………………………………………….……… 32  
Figura 24 – Página de erro 404 …………………………………………….…………. 32  
Figura 25 – Página de erro acesso negado ………………………………….………… 33  
Figura 26 – E-mail com resumo de vendas ……………………………….………….. 33

**LISTA DE ABREVIATURAS E SIGLAS**

B2B Business to Business - Negócio para negócio  
CRUD Create, Read, Update, Delete - Criar, Ler, Atualizar, Apagar.  
CSS Cascading Style Sheets – Folhas de Estilo em Cascata HTML Hypertext Markup Language – Linguagem de Marcação de Hipertexto  
ERP Interface Resource Planning – Sistema Integrado de Gestão Empresarial  
JSP JavaServer Pages – Páginas de Servidores Java  
MVC Model, View, Control – Modelo, Visão, Controle  
ORM Object-Relational Mapping – Mapeamento Objeto-Relacional  
SGBD Sistema de Gerenciamento de Banco de Dados  
SQL Structured Query Language – Linguagem de Consulta Estruturada  
TI Tecnologia da Informação  
UML Unified Modeling Language – Linguagem Unificada de Modelagem  
URL Uniform Resource Locale – Local Uniforme de Recursos

**SUMÁRIO**

**1. INTRODUÇÃO** ……………....…………………………………………….……... 09

1.1. Objetivos ……………………………………………………………………… 11

1.1.1 Geral ……………………………………………………………………. 11

1.1.2 Específico ………………………………………………………………. 11

**2. METODOLOGIA** ………………………………………………………………… 12

2.1 Tecnologias e Ferramentas utilizadas ………………………………....….. 12

2.2 Requisitos do Sistema …………………………………………………….. 16

2.2.1 Requisitos Funcionais ………………………………….……….. 16

2.2.2 Requisitos Não Funcionais ……………….…………………..…. 17

2.3 Diagramas ……………………………………………………………...… 18

2.3.1 Caso de uso ……………………………………………………... 18

2.3.2 Classe …………………………………………………………… 19

2.3.3 Entidade-Relacionamento ……………………………...….……. 20

**3. RESULTADOS E DISCUSSÃO** …………………………………………………. 21

3.1 Identidade Visual ………………………………………...…….……………… 21

3.2 Interfaces Gráficas ………………………………………………………….…. 23

3.2.1 Tela de login ……………………………………………………………. 23

3.2.2 Tela Inicial (*Dashboard*) ……………………………………………….. 24

3.2.3 Cadastro de produto …………………………………………………….. 25

3.2.4 Cadastro de cliente ……………………………………………………… 25

3.2.5 Cadastro de subcategoria ……………………………………………….. 26

3.2.6 Cadastro de marca ………………………………………………………. 26

3.2.7 Cadastro de cidade ………..…………………………………………….. 27

3.2.8 Cadastro de usuário …………………………………………………….. 27

3.2.9 Consulta de produtos …………………………………………………… 28

3.2.10 Consulta de clientes ……..…………………………………………….. 28

3.2.11 Consulta de marcas ……………………………………………………. 29

3.2.12 Consulta de subcategorias ……...…………………..………………….. 29

3.2.13 Consulta de cidades ……………………………………………..…….. 30

3.2.14 Consulta de usuários ……………………………...…………………… 30

3.2.15 Nova venda (carrinho vazio) ……………………………….………….. 31

3.2.16 Nova venda (Itens no carrinho) ………………………..………………. 31

3.2.17 Consulta de vendas ………...………………………………………….. 32

3.2.18 Página de erro 404 ……………………………...…………………….. 32

3.2.19 Página de erro Acesso Negado ………………………………………... 33

3.2.20 E-mail com resumo de vendas ……..………………………………….. 33

**4. CONSIDERAÇÕES FINAIS** …………………………………………………….. 34

**5. REFERÊNCIAS** …………………………………………………….…………….. 55

**1. INTRODUÇÃO**

Ao longo de toda a evolução da informática, ela sempre propôs a criação de novos produtos, serviços e ideias, mas mais do que isso, ela sempre se dispôs a aprimorar o que já existia. Renovar o que já funciona, agora com os recursos da tecnologia, é pensamento fundamental para qualquer estudante ou profissional da área de informática.

Segundo Stallman (2012),

“Pode-se definir  tecnologia da Informação (TI) como o conjunto de todas as [atividades](https://pt.wikipedia.org/wiki/Atividade_(engenharia_de_software)) e soluções providas por recursos de computação que visam a produção, o armazenamento, a transmissão, o acesso, a [segurança](https://pt.wikipedia.org/wiki/Segurança_da_informação) e o uso das [informações](https://pt.wikipedia.org/wiki/Escrita). Na verdade, as aplicações para TI são tantas — e estão ligadas a tantas áreas — que há diversas definições para a expressão e nenhuma delas consegue determiná-la por completo. É a área da [informática](https://pt.wikipedia.org/wiki/Informática) que trata a informação, a [organização](https://pt.wikipedia.org/wiki/Organização) e a classificação de forma a permitir a tomada de decisão em prol de algum objetivo.”

Desta forma, pode-se acrescentar que TI é utilizada como um meio ou instrumento que possibilita a revitalização de antigas práticas, a melhor compreensão do objeto aplicado, a criação de novos indicadores e de tendências potencializadoras, em qualquer que seja a área de aplicação. Por isso, observa-se a tecnologia tão impregnada em tudo. Ela é parte integrante dos serviços, produtos e informações que são geradas.

Um dos segmentos que mais usufrui da tecnologia como um todo é o corporativo, que sempre buscando melhorar seus processos, serviços, produtos e alcançar destaque no cenário de competitividade, não costuma poupar esforços quando trata-se de investimento em sistemas informatizados.

Uma pesquisa anual do PortalERP, realizada em janeiro de 2017 junto a mais de 4.000 empresas constatou que, pelo menos 44% delas, pretendiam fazer investimentos na melhoria de seus processos e sistemas.

Segundo a 28ª edição da Pesquisa do Uso de Tecnologia nas Empresas (2016), da FGV, o aumento do investimento de 1% em tecnologia, tem aumentado os lucros das companhias em 7% dentro de dois anos.

Diante dos constantes avanços tecnológicos, as empresas tendem a perceber a importância de incorporá-los, extraindo o máximo de benefícios que a tecnologia pode gerar para seus resultados, pois elas percebem que a informatização converge com a maximização dos indicadores e melhoria no tratamento das informações.

O software em questão debruça-se sobre um nicho específico de vendas, conhecido como *Business to Business*, B2B, ou Negócio para Negócio, que nada mais é do que a venda para outras empresas, ao invés de direto ao cliente final. Alguns exemplos seriam indústrias automobilísticas, que vendem os carros para agências de automóveis (empresa), e essas então vendem ao consumidor final. Ou a empresa de sorvetes Kibon, que não possui lojas físicas, mas vendedores que se responsabilizam em distribuir os produtos para lanchonetes, supermercados, padarias e outros, não tendo contato direto com quem irá consumi-lo de fato.

E foi pensando nesta dinâmica de negócios que a aplicação foi desenvolvida. Ela toma como exemplo um distribuidor de produtos eletrônicos, que possui equipe de vendedores responsáveis por fechar vendas com lojas especializadas. A ideia surgiu após a observação de negócios, no estilo descrito, ocorrendo sem a utilização de nenhuma tecnologia.

Após uma pesquisa, encontrou-se dificuldade em identificar softwares que cumprissem tais quesitos. Os sistemas de vendas existentes são muito voltados para o tipo de venda mais comum, vendedor-cliente final, deixando carentes os que negociam com outras empresas. Isso ajudou a corroborar com a ideia de desenvolvimento da solução.

**2. OBJETIVOS**

**2.1. Objetivo geral**

Desenvolver um sistema web, capaz de atender às demandas de uma empresa no modelo B2B, no que se refere à parte de vendas e produtos.

A aplicação deve oferecer controle sobre o estoque de produtos, garantindo a fidedignidade dos dados e com isso, permitir operações de vendas pelos vendedores e ações de consultas de informações pertinentes aos usuários.

**2.2. Objetivos Específicos**

* Identificar os requisitos do sistema;
* Selecionar as linguagens e tecnologias que proverão o desenvolvimento da aplicação;
* Garantir que o software poderá ser acessado satisfatoriamente através de dispositivos móveis;
* Modelar o sistema e o banco de dados através de diagramas;

**3. METODOLOGIA**

As tecnologias e linguagens escolhidas foram baseadas em profunda análise do cenário de cada uma das opções e suas relativas integrações.

Após selecionar a linguagem que demandaria todas as outras, o Java, foi feito um levantamento de quais tecnologias integrariam-se de forma interessante, sua relevância na comunidade e suas perspectivas de futuro.

Além disso, foi necessário a observação de alguns requisitos pontuais do sistema, como a necessidade de funcionar bem em dispositivos móveis e de delimitação das permissões de acordo com o tipo de usuário.

Portanto, tecnologias em desuso, com pouca documentação, sem muito respaldo da comunidade de desenvolvedores ou que não atendiam aos requisitos necessários foram descartadas, chegando então ao conjunto de soluções e ferramentas aplicadas no desenvolvimento da aplicação.

**3.1 Tecnologias e Ferramentas utilizadas**

**3.1.1 UML (Unified Modeling Language)**

A UML, ou Linguagem Unificada de Modelagem, é uma linguagem gráfica para visualização, especificação, construção e documentação de artefatos de sistemas complexos de software. (BOOCH, 2005).

Foi utilizado como guia para a criação dos três diagramas contidos neste trabalho: o diagrama de classes, o diagrama de caso de uso e o modelo entidade-relacionamento.

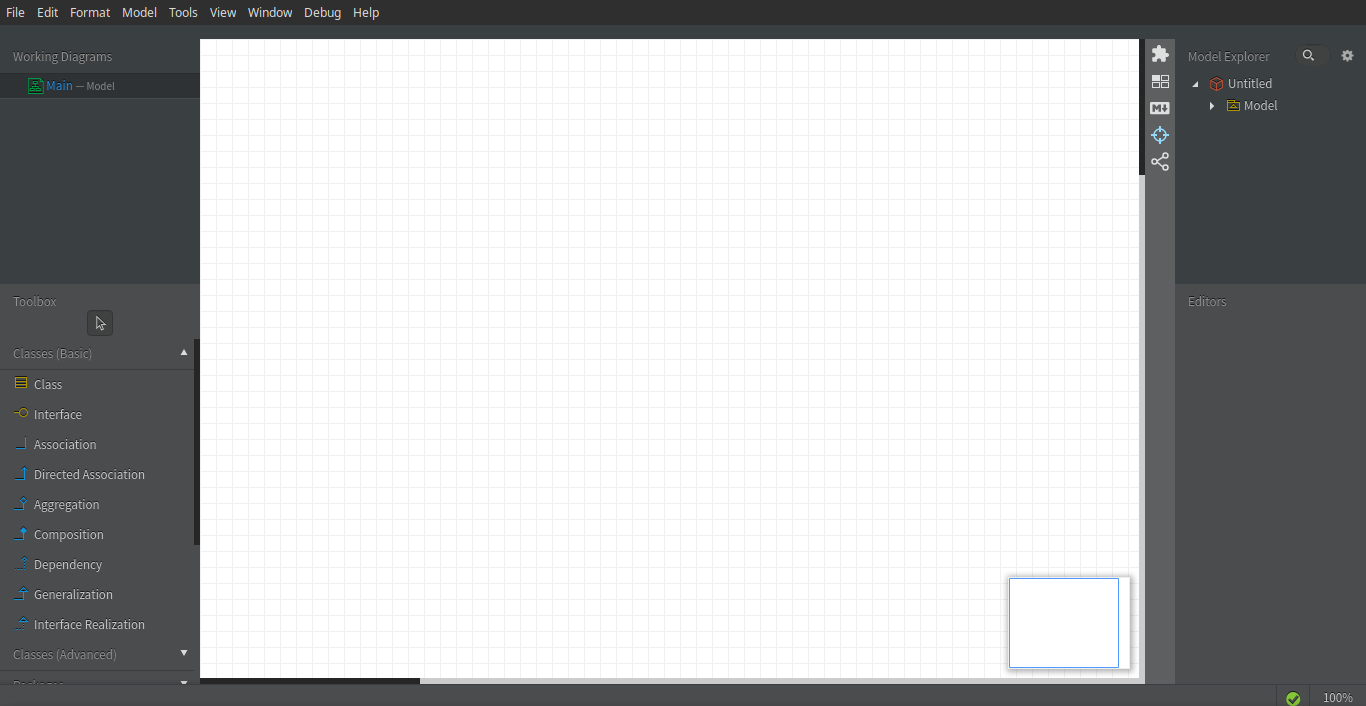
**3.1.2 StarUML**

O StarUML é um software que possibilita a criação de modelagens e diagramas

seguindo a padronização da UML 2.0, a última padronização da linguagem.

Todos os diagramas de software apresentados neste trabalho foram criados nesta aplicação.

Figura 1: Interface do programa StarUML



**3.1.1 Java**

Java é uma linguagem de programação orientada a objetos criada no início dos anos 90. Desde então é extremamente usada e difundida, sendo precursora de preceitos que seriam e são replicados em diversas outras linguagens.

É hoje uma das linguagens mais utilizadas no mundo, sendo base para a criação de aplicativos para aparelhos móveis com sistema Android.

Java é a principal linguagem da Internet,   
mas é mais do que isso. Ela revolucionou a programação, mudando a maneira de pensarmos tanto sobre a forma quanto sobre a função de um programa. Atualmente, ser um programador profissional exige a habilidade de programar em Java, tal é sua importância. (SCHILDT, 2015).

A linguagem foi escolhida por ser bastante robusta, com vasta documentação, suporte da comunidade e ser multiplataforma, permitindo a programação em ambiente web, que pensou-se como ideal para atender a todos os requisitos do software.

**3.1.2 Spring Framework**

O Spring Framework é uma coleção de soluções que ajudam na criação de aplicações Java, principalmente web.

Dentro dela, estão sendo utilizados neste projeto os seguintes módulos:

Spring MVC, que ajuda na construção de aplicações web dentro do padrão Model, View e Controller;

Spring Data, que auxilia na parte de persistência com o banco de dados, principalmente nas operações de CRUD da aplicação;

Spring Security, que provê facilidades para a parte de segurança do sistema;  
  
 O Spring Framework foi escolhido pois reunia uma gama de soluções que seriam necessárias no projeto, favorecendo-a em comparação com as tecnologias concorrentes.

**3.1.4 MySQL**

É um Sistema de Gerenciamento de Banco de Dados (SGBD) dos mais populares no mundo.

Utiliza a linguagem SQL e é extremamente difundida no mundo do desenvolvimento.

Foi escolhido pois apresenta uma solução simples, consolidada, com boa documentação de fácil implementação.

**2.1.5 Wildfly**

O Wildfly foi utilizado para a incrementação gradual do banco de dados e controle versionado dessas evoluções.

**2.1.6 Hibernate**

O Hibernate é o *framework* baseado no conceito ORM (Modelo Entidade-Relacionamento) mais utilizado atualmente. Ele abstrai as operações com banco de dados e disponibiliza uma enorme gama de propriedades e opções para controle do fluxo de informação da aplicação.

**2.1.3 HTML 5**

HTML é a sigla em inglês para *HyperText Markup Language*, que em português significa linguagem para marcação de hipertexto.

Nada mais é que uma linguagem de marcação que estrutura documentos seguindo os Padrões Web.

O número 5 refere-se à última versão adotada pela internet.

**2.1.7 Javascript**

O Javascript faz parte da tríade de desenvolvimento web, além do HTML e do CSS. É ele que define o comportamento da página, no sentido de interatividade.

Apesar de a linguagem ter crescido bastante e ter se tornado bem mais do que só definição de como a página irá se comportar, é essa funcionalidade que será incorporada neste projeto.

**2.1.8 CSS**

O CSS (*Cascading Style Sheets*) é utilizado para definir o estilo e posicionamento dos elementos dentro das páginas HTML. As fontes, cores, margem, altura, largura e inúmeros outros aspectos são controlados através de códigos escritos com esta tecnologia.

**2.1.9 Bootstrap**

O Bootstrap é uma biblioteca para ajudar na criação de interfaces.

O framework, criado pelos desenvolvedores do Twitter, contém uma vasta quantidade de elementos e componentes estilizados e já adequados para adaptação em vários tamanhos de telas.

Foi escolhido pois retira um pouco da responsabilidade da estilização e comportamento das páginas, sobrando mais tempo para que o desenvolvedor foque na solução do problema, e também fornece componentes já prontos para serem bem renderizados em diferentes formatos de telas, a chamada responsividade.

**2.1.10 Thymeleaf**

O Thymeleaf é uma *template engine* que ajuda a dinamizar nossas páginas HTML, tornando possível que elas se comuniquem com o código *back-end*, nesse caso, o Java.

Foi escolhido pois aparentou ser uma solução melhor do que o “concorrente” JSP para a criação deste tipo de aplicação monolítica. E ao longo do projeto realmente mostrou-se mais dinâmica e intuitiva.

**2.1.11 Eclipse IDE**

Como Ambiente de Desenvolvimento Integrado (IDE), foi utilizado o Eclipse em sua versão denominada de Neon.2.

O Software é grátis e de código-aberto.

**2.1.12 Servidor**

Como servidor web, foi escolhido o Tomcat, mais precisamente sua versão 8.5.  
Além de prático e leve, é ideal para este tipo de aplicação mais simples.  
É mantido pela Apache e tem seu código livre e aberto, sendo gratuita a sua utilização.

**2.1.13 Git**

O Git é uma linguagem de versionamento de código gratuita e open-source.  
Ela facilita a organização e o rastreamento do projeto.

**2.1.14 GitHub**

O GitHub é uma plataforma gratuita que permite a hospedagem de código-fonte de softwares através da linguagem de controle de versionamento Git.

**2.2 Requisitos do Sistema**

Após a escolha das tecnologias a serem utilizadas na criação do programa, foi necessário colher os requisitos para a definição das funcionalidades a serem desenvolvidas.

Após breve análise de softwares de vendas disponíveis, mesmo que focados em outros nichos, a observação de vendedores no processo de vendas e a percepção de desenvolvedor para identificar e mapear as demandas, foram pensados os requisitos que serão apresentados a seguir, divididos em Funcionais e Não Funcionais, conforme orientação de arquitetura e engenharia de software.

Segundo Sommerville (2011, página 59), os requisitos funcionais são o descritivo dos serviços, funções, ações e operações que o sistema deve prover.

Já os Requisitos Não Funcionais, segundo Sommerville (2011, página 59), são características não relacionadas aos serviços específicos da aplicação, mas características globais como: robustez, segurança, disponibilidade e usabilidade.

**2.2.1 Requisitos Funcionais**

* O sistema deverá permitir que o administrador cadastre e remova vendedores;
* O administrador poderá atualizar seus dados cadastrais e dos vendedores;
* O sistema deverá permitir que o usuário cadastre produtos, marcas e subcategorias, clientes e cidades;
* O sistema deverá possibilitar que um administrador retire relatórios;
* O usuário poderá, através de vários filtros, pesquisar pelos produtos no banco de dados;
* O sistema deverá permitir que o vendedor realize uma venda;
* O sistema deverá permitir que o vendedor envie o resumo da venda para o e-mail do cliente;
* O sistema deverá possuir um controle de usuários.

**2.2.2 Requisitos Não Funcionais**

* O software deverá funcionar em qualquer browser;
* O sistema deverá ser acessado por qualquer sistema operacional;
* A aplicação deverá funcionar em dispositivos móveis;
* A aplicação só deverá ser acessada após login e senha;
* O software deve manter cores neutras;
* A navegabilidade deve ser simples e intuitiva para qualquer usuário;
* O sistema deverá ter fácil usabilidade, independente do nível de conhecimento e experiência do usuário;
* As respostas do software às ações do usuário devem ser rápidas.

**2.3 Diagramas**

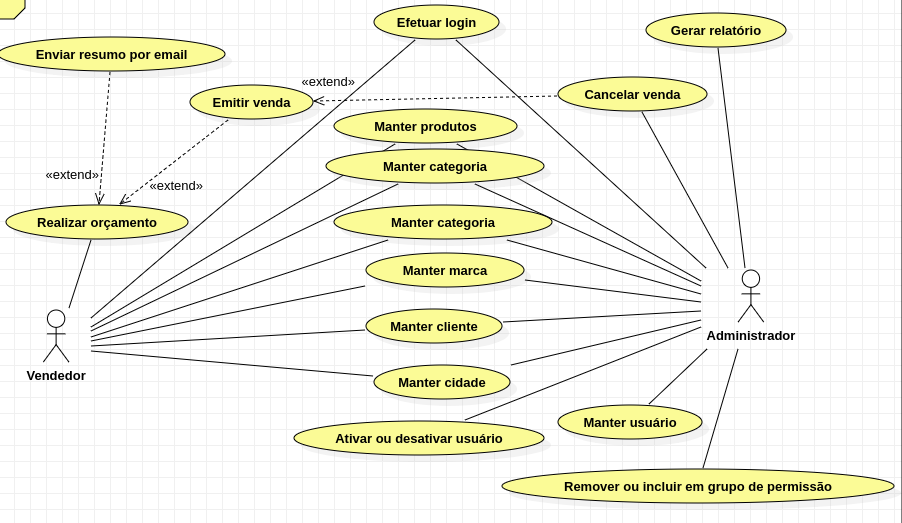
**2.3.1 Caso de Uso**

O diagrama de caso de uso traz as funcionalidades do sistema para o contexto do usuário, sendo então mais simples perceber como elas vão interagir ou reagir de acordo com as ações dos tipos de usuário, identificados no diagrama como atores do sistema.

Cada elipse representa uma funcionalidade. As linhas contínuas que ligam-as aos atores identificam que elas são possíveis somente pelo tipo de usuário a que elas estão ligadas.

Linhas tracejadas identificam que aquela ação somente é possível após a ação a qual ela está interligada.

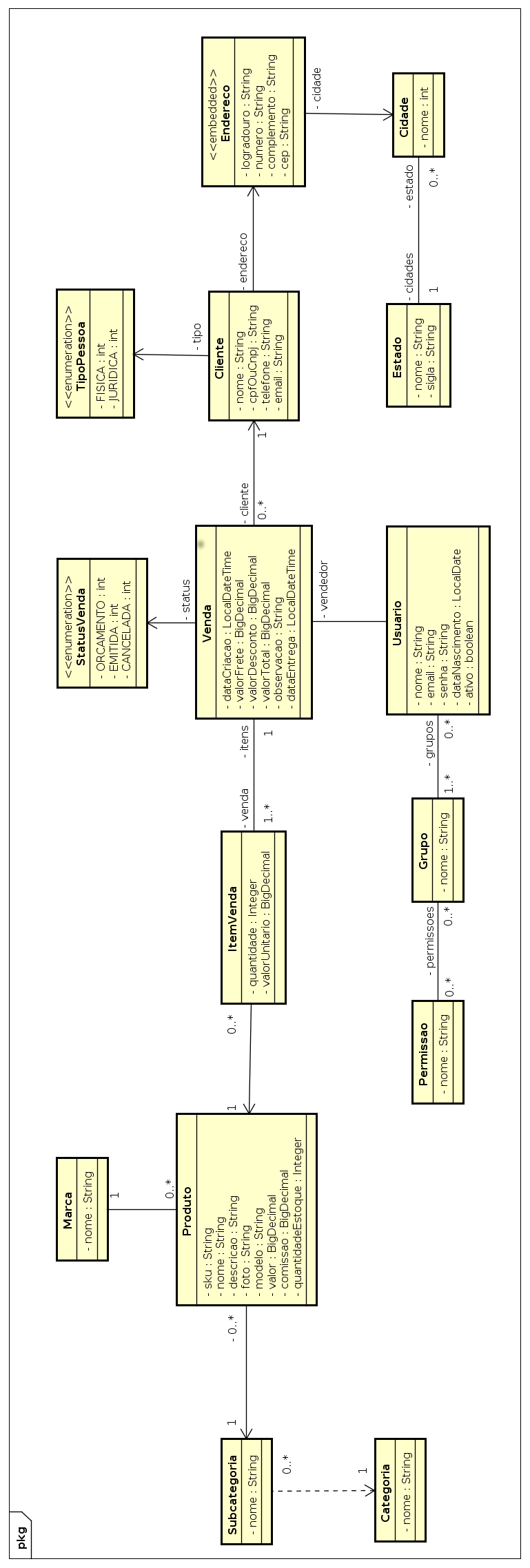
Figura 2 – Diagrama de Caso de Uso

Autoria própria

**2.3.2 Diagrama de Classe**

O diagrama de classes apresenta todas as classes, mais precisamente da camada de modelo da aplicação, e os seus relacionamentos.

Figura 3: Diagrama de Classes



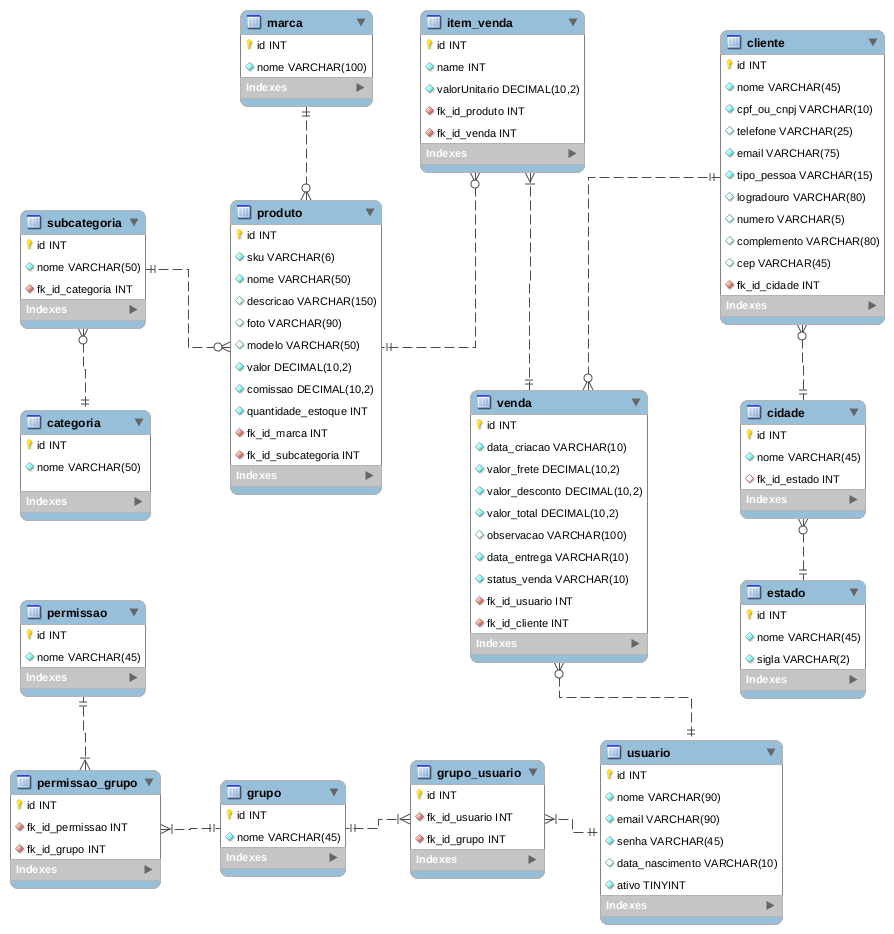
Autoria própria

**2.3.3 Modelo Entidade-Relacionamento**

O modelo entidade-relacionamento apresenta todas as tabelas do banco de dados da aplicação, suas relações e respectivas cardinalidades.

Dentro de cada retângulo representando a tabela, é possível identificar suas colunas, o tipo de dado, e se aquela coluna representa uma chave primária ou estrangeira

Figura 4: Modelo Entidade-Relacionamento



Autoria própria

**3. RESULTADOS E DISCUSSÃO**

**3.1 Identidade Visual**

Apesar do software ser aplicável a qualquer ramo de negócios, dentro da proposta B2B, a aplicação foi pensada para um cliente fictício especialista em produtos eletrônicos. Portanto, foi desenvolvida, em conjunto com a solução sob demanda, uma identidade visual para a empresa.

Definiu-se o cliente como Mago Digital. Uma revendedora de produtos eletrônicos variados.

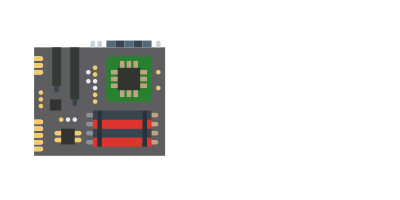
Além de aproximar a experiência de desenvolvimento deste projeto o máximo possível de uma criação de solução no mundo real, as imagens e marca são utilizadas no software para representar a empresa e ligá-lo ainda mais à companhia.

Figura 5 - Identidade Visual da Empresa Fictícia Mago Digital



Autoria própria

Figura 6 - Parte separada da identidade visual



Autoria própria

Figura 7 – Parte textual da imagem visual



Autoria própria

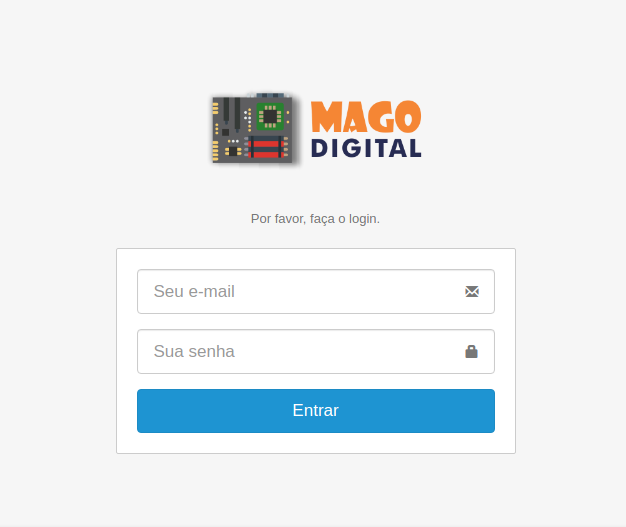
**3.2 Interfaces Gráficas**

Após todas as análises, planejamento, escolha das tecnologias e ferramentas, o software foi construído. Levando em conta todos os requisitos levantados e alguns observados ao longo do desenvolvimento, a aplicação atende a todas as necessidades previstas no início, indo além e incorporando funcionalidades adicionais.

**3.2.1 Tela de login**

Para efetuar login no sistema, é necessário cadastro prévio de um administrador, e para credenciais são utilizados e-mail e senha.

Figura 8 – Tela de login

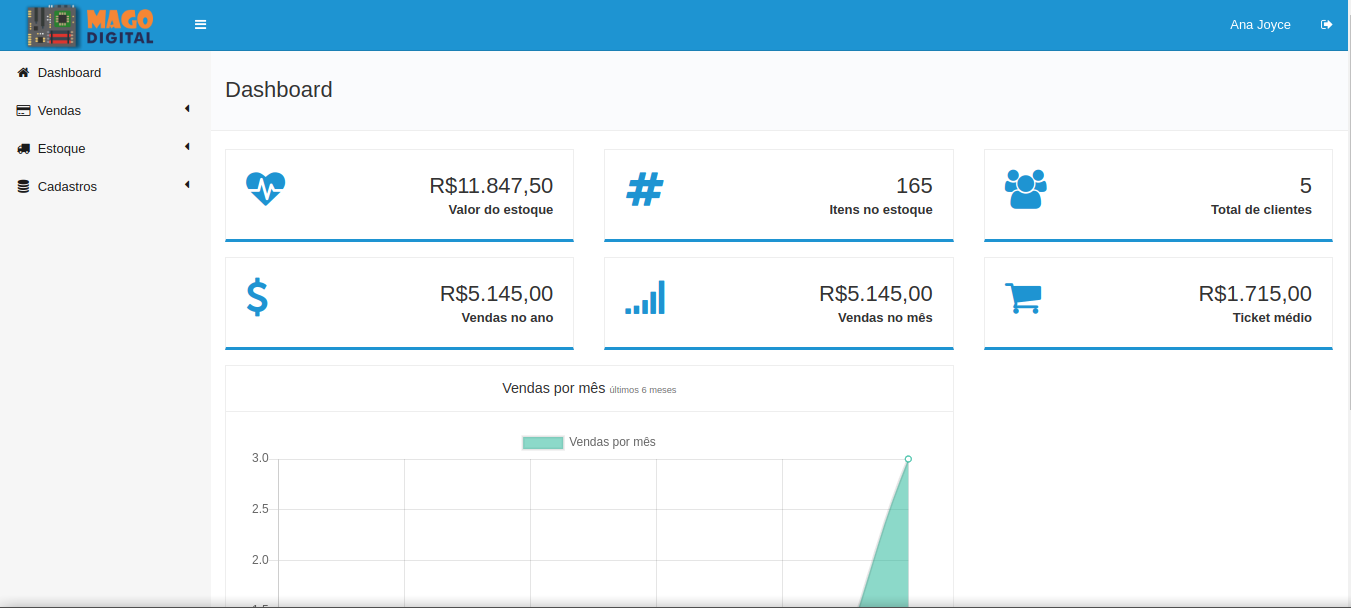


**3.2.2 Dashboard**

Ao efetuar login, o usuário é direcionado para a tela inicial, que apresenta algumas informações gerais de estoque, valores e dados referentes ao negócio e nome do usuário logado.

É apresentado também um gráfico com as vendas por mês dos últimos 6 meses.  
A partir daqui, é possível efetuar logout no canto superior direito, ou navegar no sistema junto ao menu lateral esquerdo.

Figura 9 – Dashboard ou tela inicial



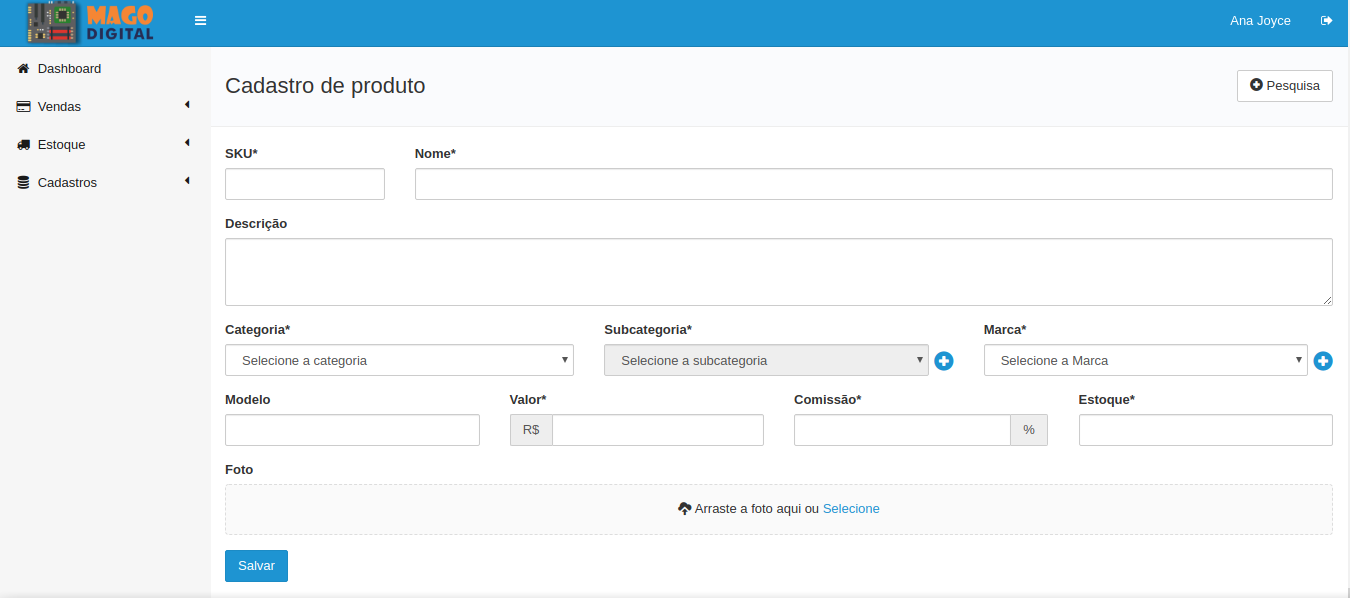
**3.2.3 Cadastro de produto**

Para cadastrar o produto, são necessárias algumas informações básicas.  
O SKU é um identificador único de produto, portanto, não se repete e há uma validação para isso. Os campos com \* são obrigatórios, enquanto os outros são opcionais.

No campo de foto, é possível selecionar a partir do *explorer* ou arrastar a imagem diretamente.

Antes de salvar, todos os campos são validados e então é salvo ou mantido e emitidas as mensagens de erro.

Figura 10 – Cadastro de produto



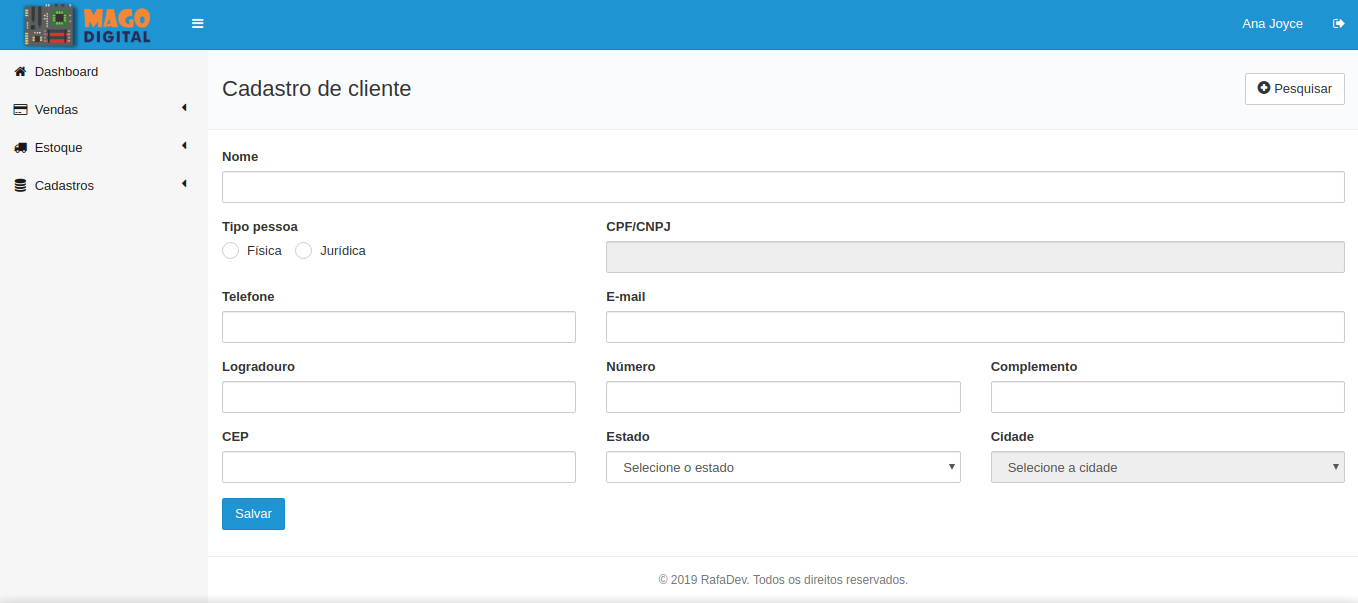
**3.2.4 Cadastro de cliente**

O cadastro de clientes segue o padrão comum de cadastro.

O campo CPF/CNPJ é habilitado após a escolha do Tipo pessoa, e a validação é feita conforme o tipo de dado que será fornecido.

A cidade só é habilitada após a seleção do Estado, que então carrega as cidades que fazem parte daquele Estado em específico.

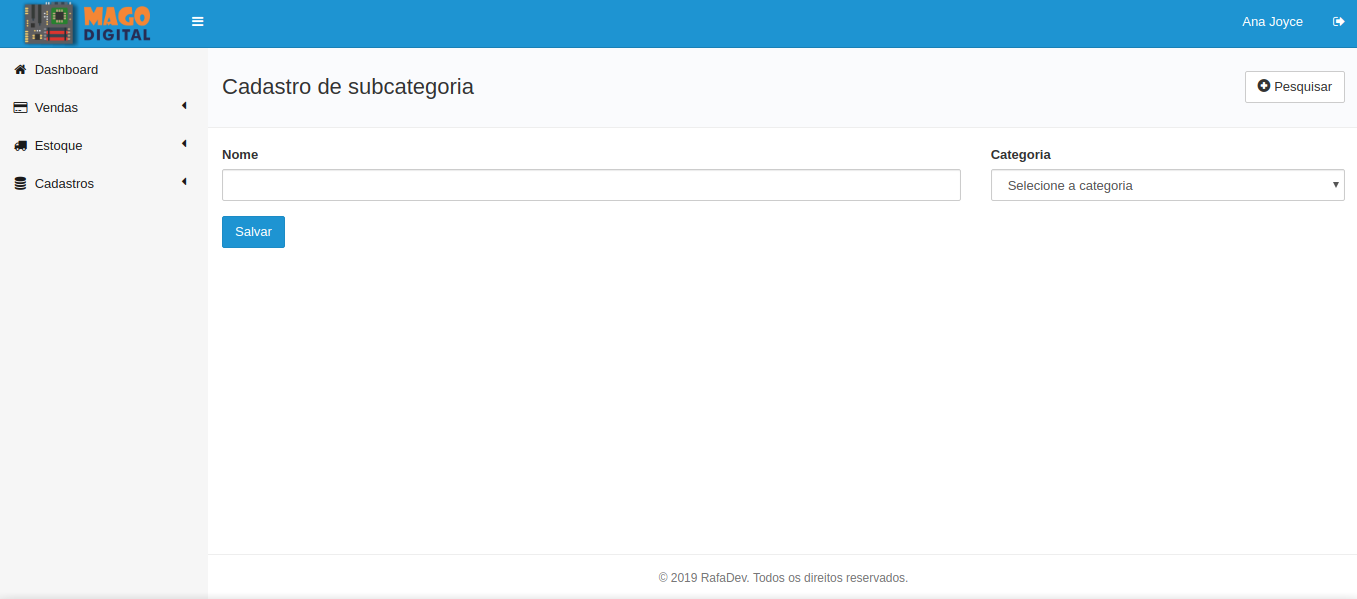
Figura 11 – Cadastro de cliente



**3.2.5 Cadastro de subcategoria**

Os produtos são divididos em categorias, que por sua vez possuem suas subcategorias. As categorias já estão pré-cadastradas no sistema, sendo possível ao usuário o cadastro de suas subcategorias.

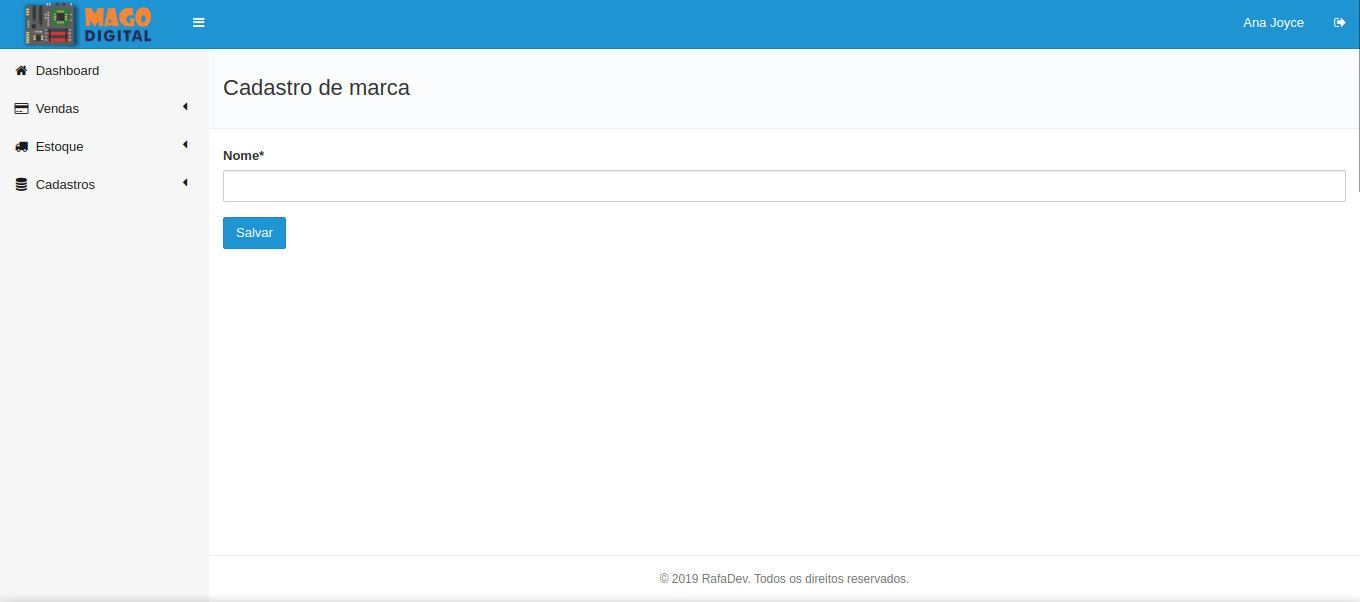
Não é possível salvar uma subcategoria com nome em branco, sem categoria ou com o mesmo nome de uma outra já cadastrada anteriormente.

Figura 12 – Cadastro de subcategoria

**3.2.6 Cadastro de Marca**

Para posterior relacionamento, foi pensado o cadastro de marcas para ser associada ao produto. Nome, que é o único campo, é obrigatório e não pode ser o mesmo que outra marca já cadastrada anteriormente.

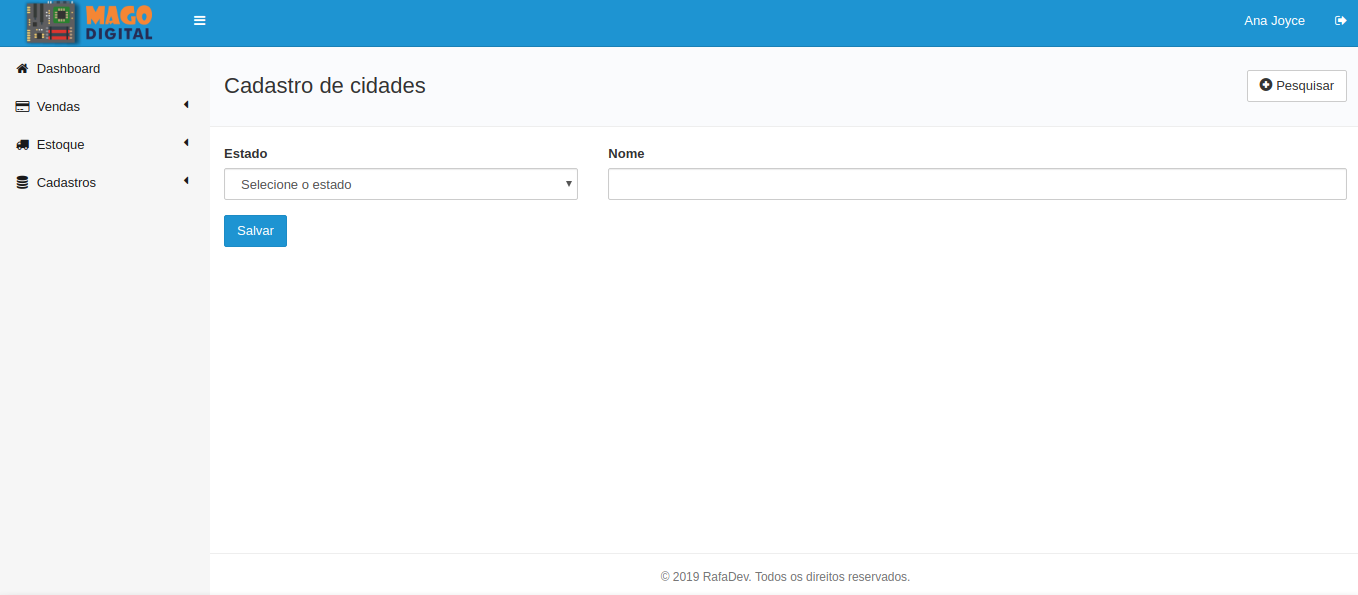
Figura 13 – Cadastro de marca



**3.2.7 Cadastro de cidade**

Para posterior relacionamento, foi pensado o cadastro de marcas para ser associada ao Estado, e consequentemente ao cliente. Nome, que é o único campo, é obrigatório e não pode ser o mesmo que outra cidade já cadastrada anteriormente no mesmo Estado, visto que existem cidades com mesmo nome em Estados diferentes.

Figura 14 – Cadastro de cidade



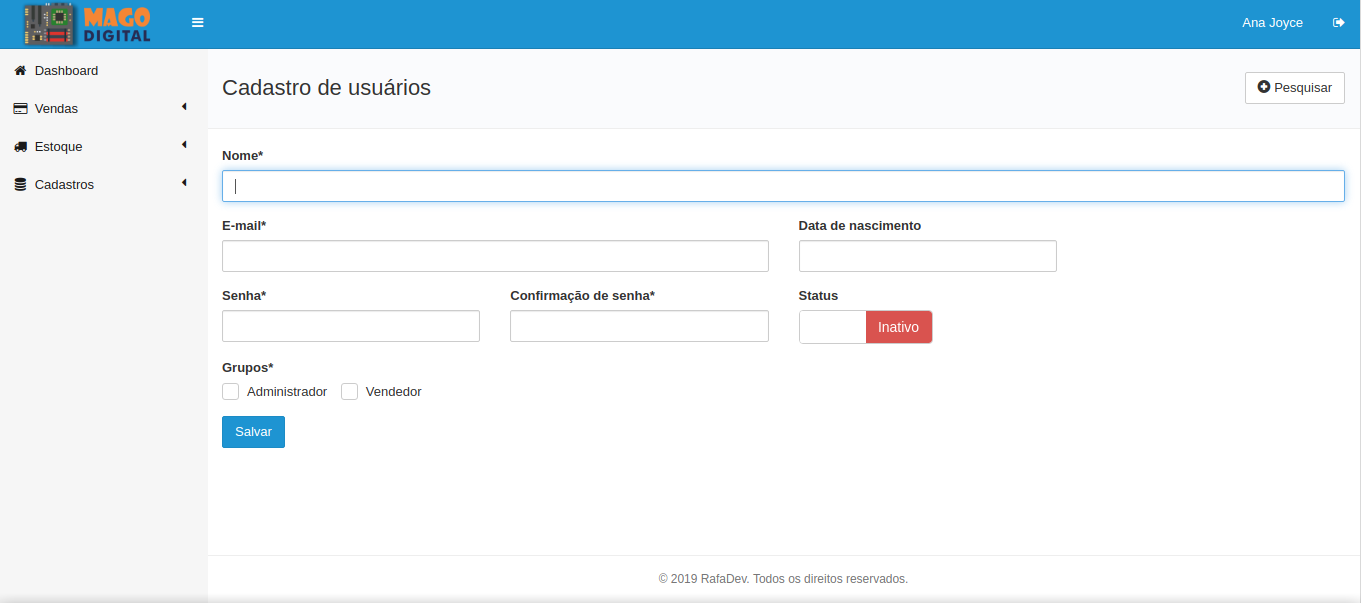
**3.2.8 Cadastro de usuário**

Nesta tela, um administrador pode cadastrar outros usuários para terem acesso ao sistema. Além da data de nascimento, campos são todos obrigatórios.

Somente um usuário com status “Ativo” poderá acessar o sistema.

Há dois grupos de permissão: o administrador, que tem acesso a todas as áreas, e o vendedor, que não pode cancelar vendas de outros vendedores, cadastrar ou pesquisar outros usuários.

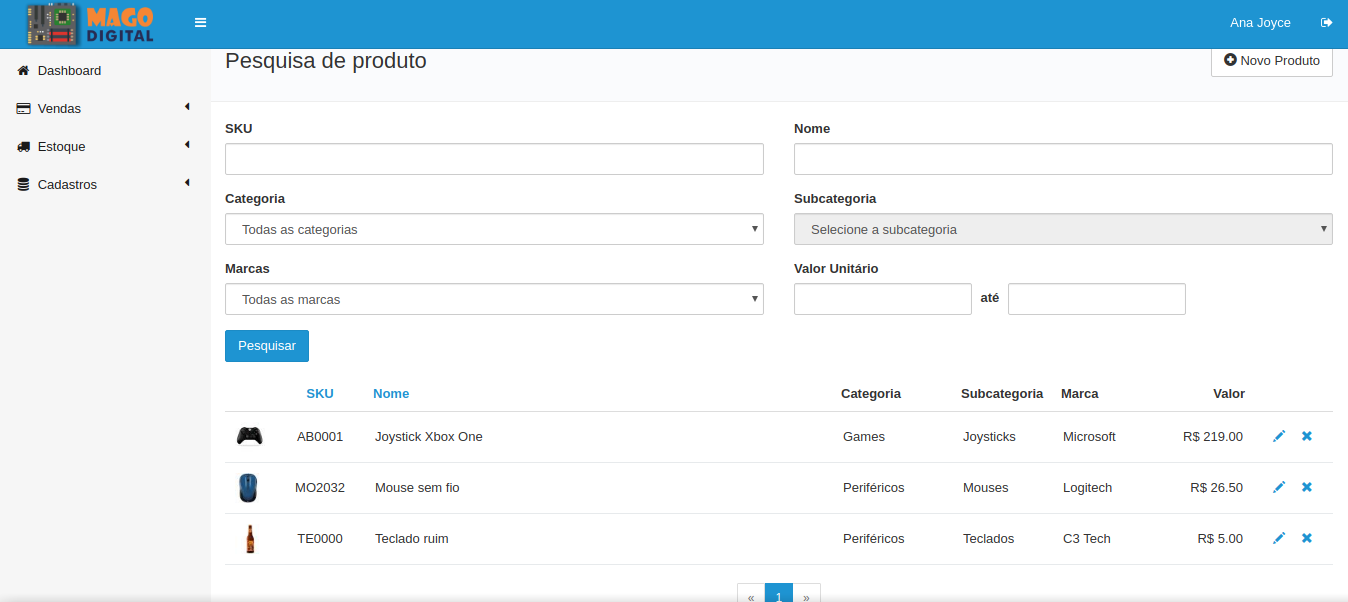
Figura 15 – Cadastro de usuário

****

**3.2.9 Consulta de produtos**

Nesta tela é possível consultar produtos através de filtros, além de acessar as telas de edição e exclusão do produto.

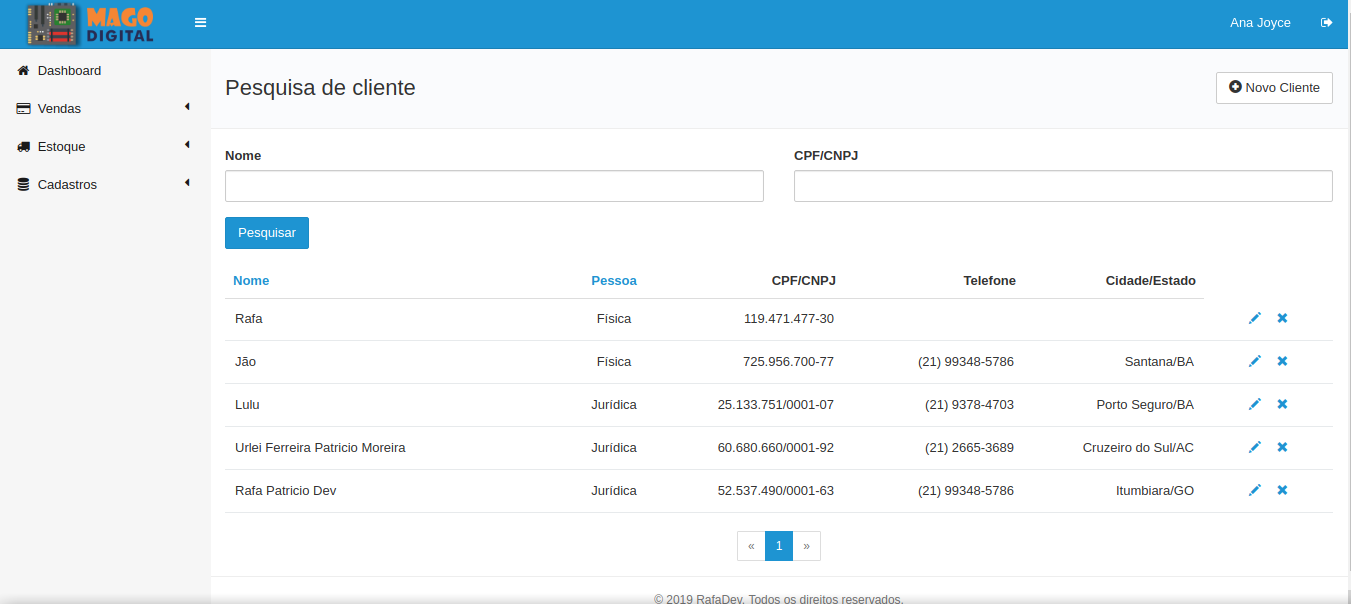
Figura 16 – Consulta de produtos



**3.2.10 Consulta de clientes**

Permite a consulta de clientes com filtro, e na tabela oferece as opções de edição e exclusão, além de ser possível ordenar os dados por nome ou tipo de pessoa.

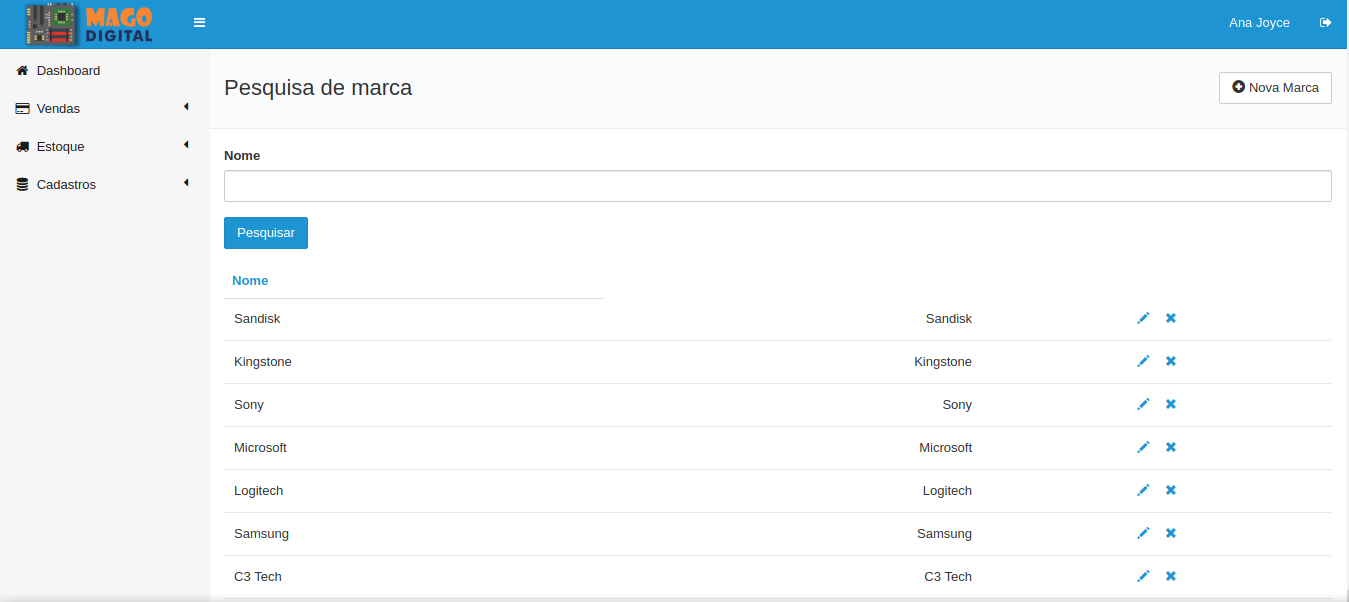
Figura 17 – Consulta de clientes



**3.2.11 Consulta de marcas**

Permite a busca de todas as marcas ou filtrar por nome, além de permitir edição, exclusão e ordenação na tabela.

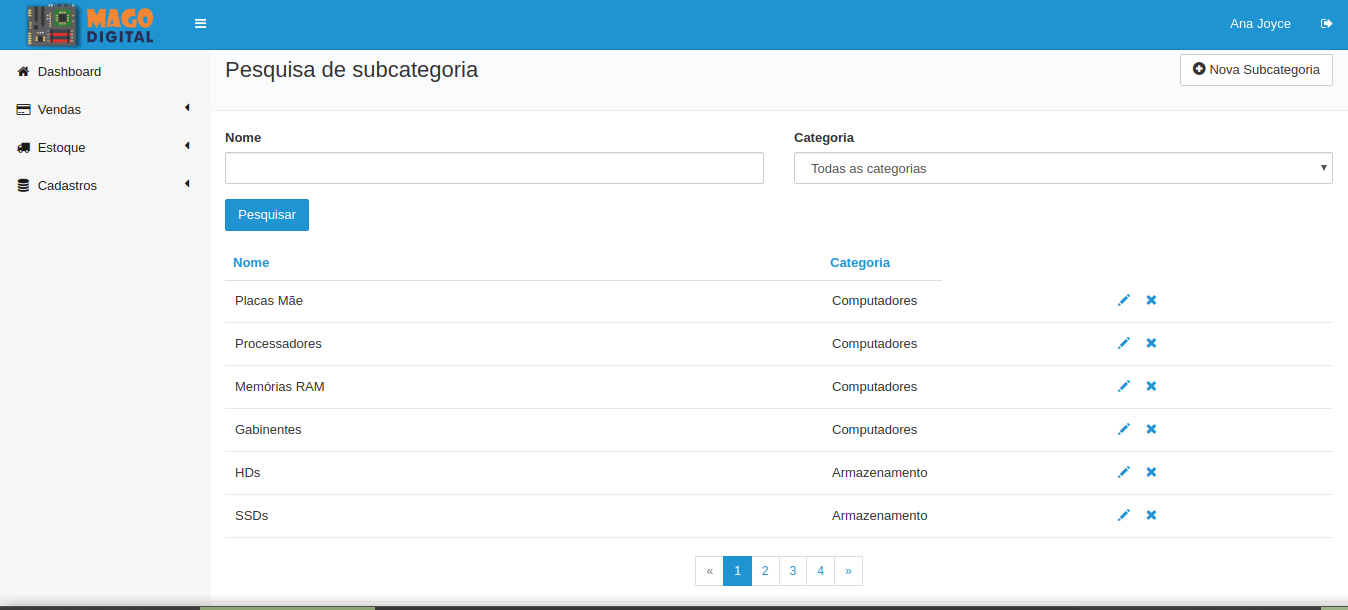
Figura 18 – Consulta de marcas



**3.2.12 Consulta de subcategorias**

Permite a busca de todas as marcas ou filtrar por nome e/ou categoria, além de permitir edição, exclusão e ordenação na tabela.

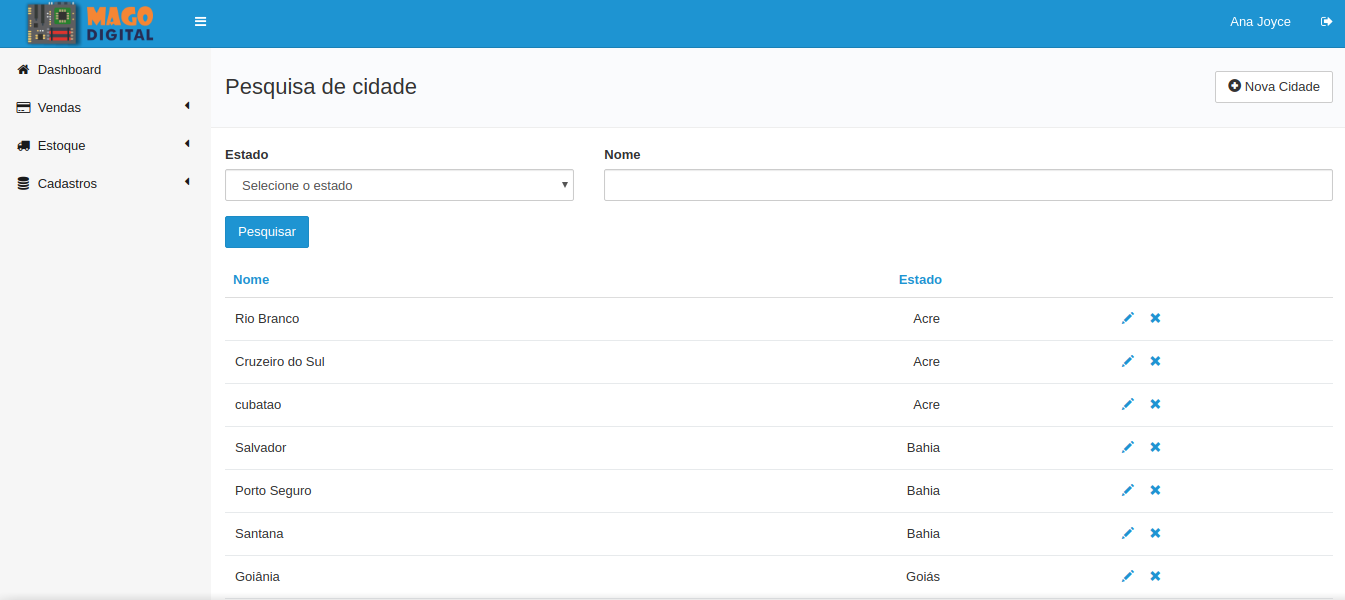
Figura 19 – Consulta de subcategorias



**3.2.13 Consulta de cidades**

Permite a busca de todas as cidades ou filtrar por nome e/ou Estado, além de permitir edição, exclusão e ordenação na tabela.

Figura 20 – Consulta de cidades

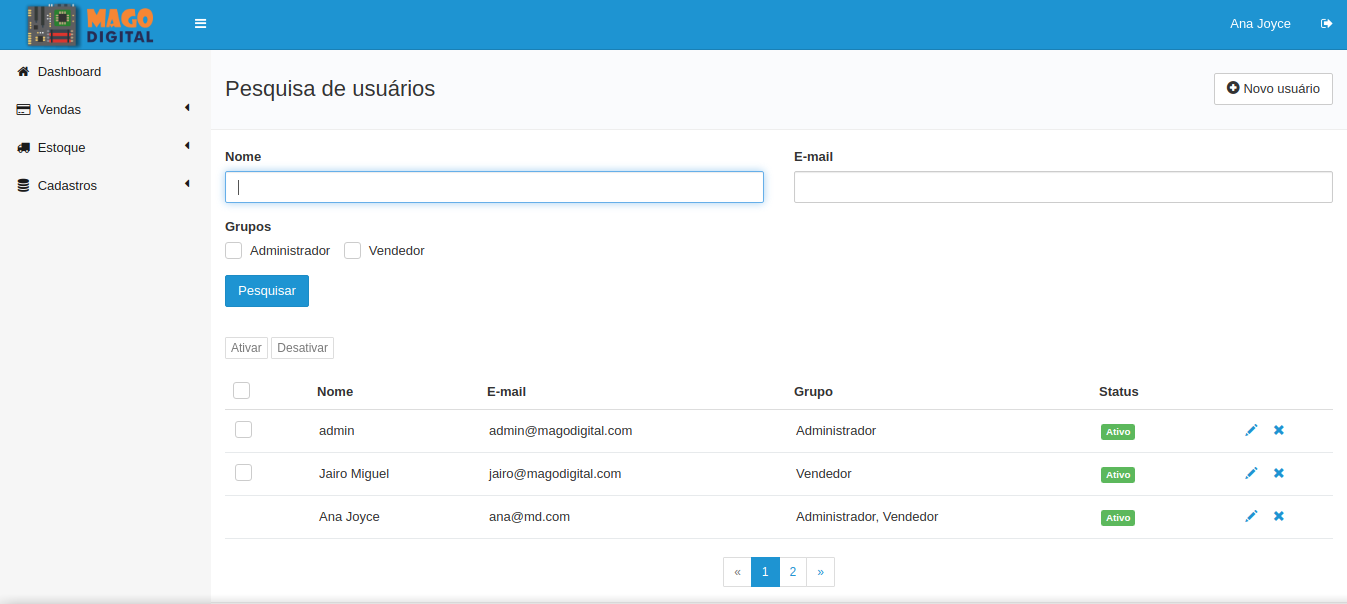


**3.2.14 Consulta de usuário**

Permite a busca de todos os usuários ou filtrar por nome e/ou email, além de permitir edição, exclusão e ordenação na tabela.

Só é acessada por quem tem a permissão de administrador.

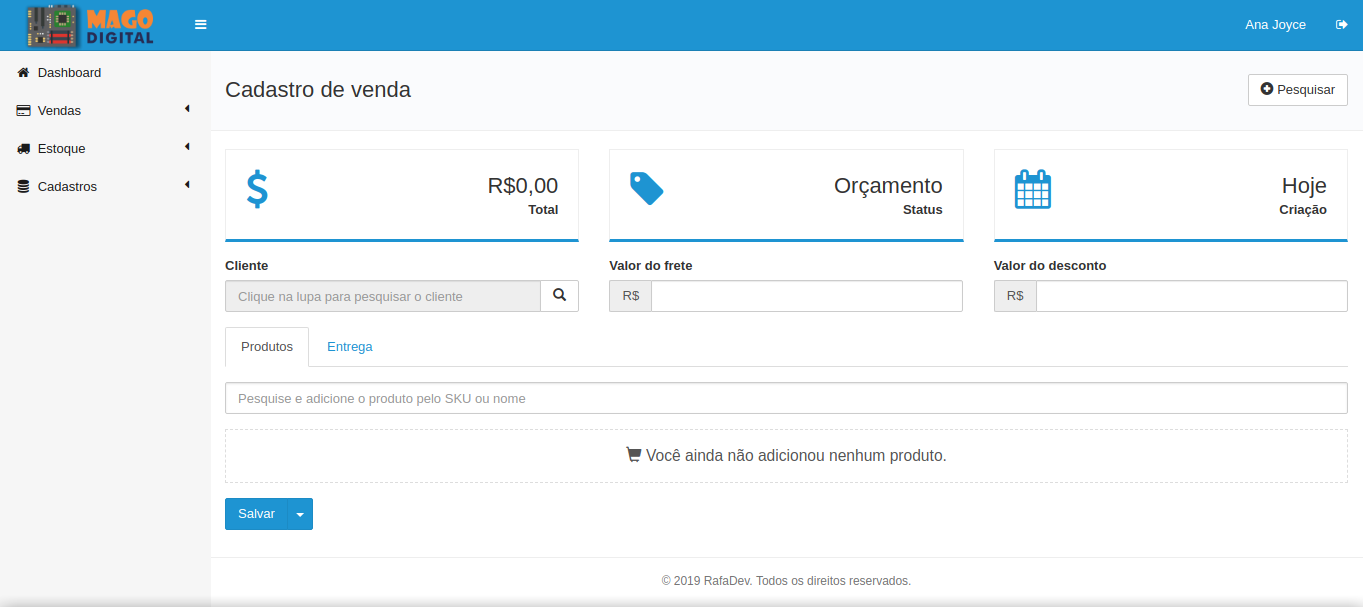
Figura 21 – Consulta de usuários



**3.2.15 Nova Venda (carrinho vazio)**

Nesta tela, é possível efetuar uma venda.

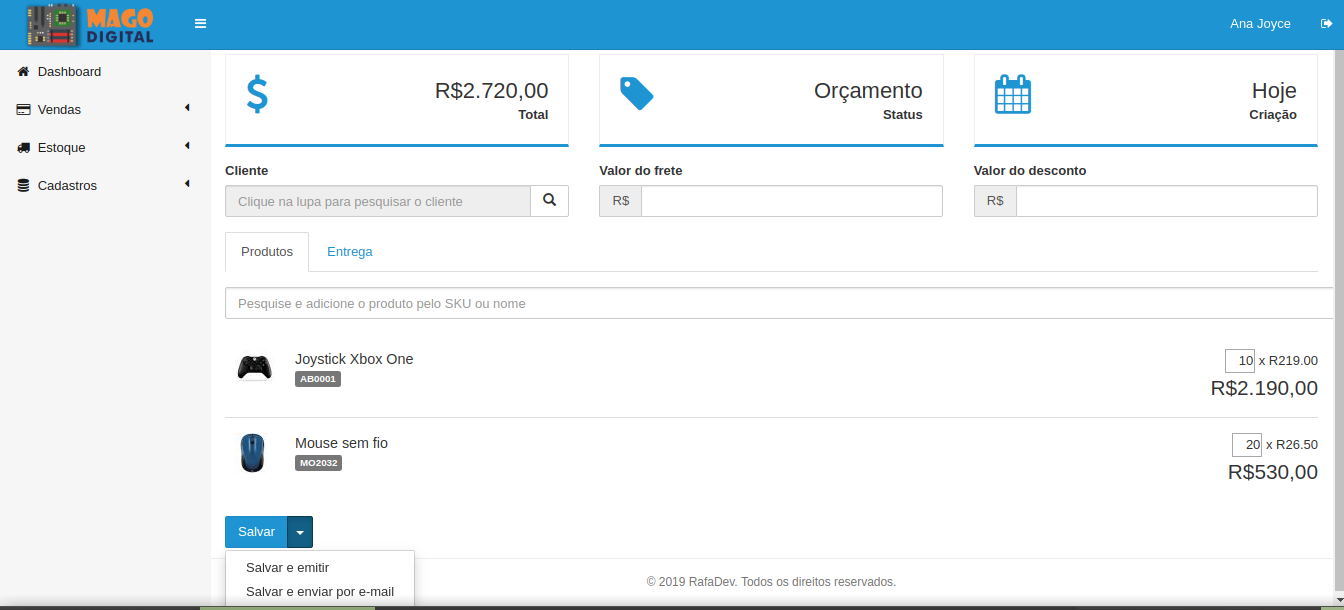
Após informar os dados necessários, como cliente, frete e produtos, é possível salvar a venda como um orçamento, emitir a venda e também enviar o resumo via e-mail.

Figura 22 – Nova venda (carrinho vazio)

**3.2.16 Nova venda (itens no carrinho)**

A tela de vendas após inclusão de itens atualiza as informações junto aos painéis acima. O valor total é calculado, e é possível somente salvar o orçamento, emitir a venda ou emitir e enviar por e-mail.

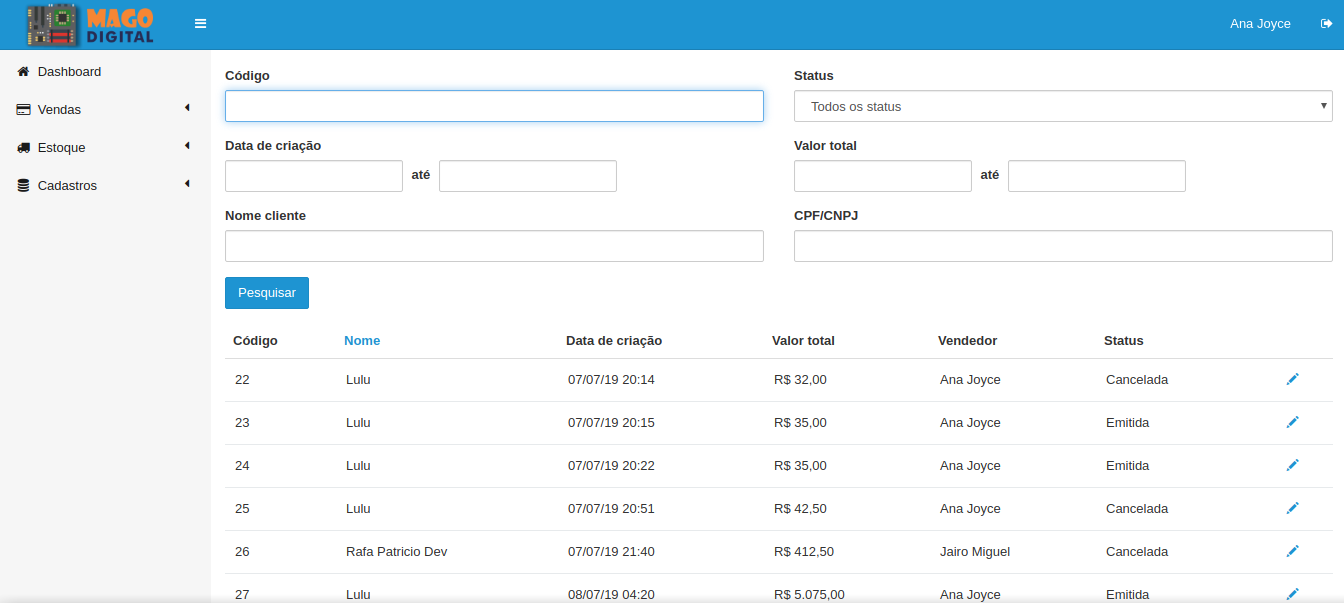
Figura 23 – Nova venda (itens adicionados)



**3.2.17 Pesquisa de vendas**

É possível a pesquisa de vendas através de filtros.

É permitida a edição, mas a exclusão é somente para administradores ou para o vendedor que efetuou a venda/orçamento.

Figura 24 – Consulta de vendas

**3.2.18 Tela de erro 404 (recurso não encontrado)**

Ao acessar url incorreta, o usuário é direcionado para a tela de erros abaixo, informando que a página não foi encontrada e com o botão para retornar à pagina anterior.

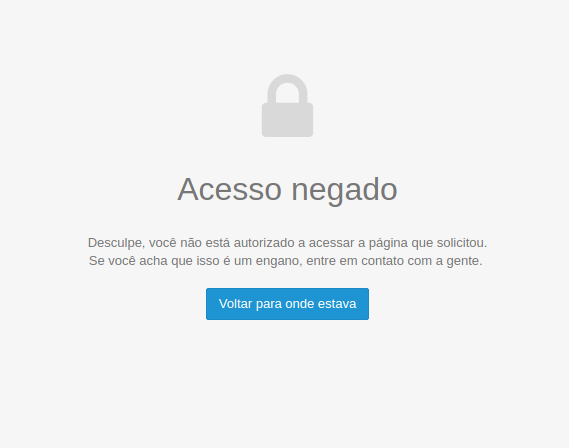
Figura 25 – Página de erro 404



**3.2.19 Tela de erro para acesso negado**

Ao acessar recurso não permitido para o nível de permissão do usuário, é exibida a mensagem abaixo, com o informativo e botão para retornar à navegação.

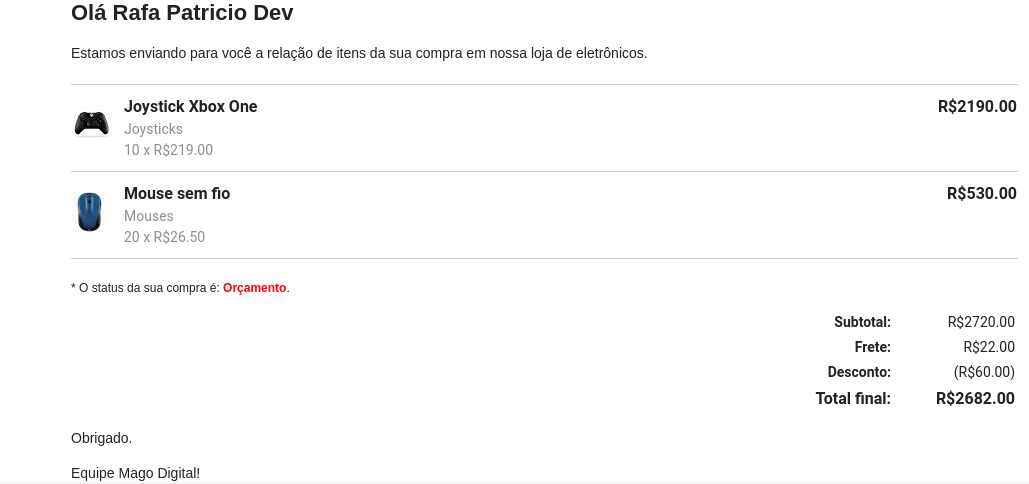
Figura 26 – Página de erro acesso negado

****

**3.2.20 E-mail com resumo de vendas**

Ao emitir a venda, é possível enviar ao cliente através do e-mail cadastrado, o resumo da venda efetuada, que apresenta descritivo dos itens, quantidades e valores.

Figura 27 – E-mail com resumo de vendas



**4. CONSIDERAÇÕES FINAIS**

Desde o início, a ideia sempre foi a criação não só de um sistema funcional, mas que utilizasse das tecnologias mais eficientes, modernas e em evidência dentro do cenário da linguagem escolhida.

Era intenção do autor estudar, aprender e criar uma aplicação com os requisitos observados em vagas de emprego, da forma que grandes empresas e *startups* de sucesso faziam, utilizando muito do que dizem ser as melhores práticas, ferramentas, bibliotecas e funcionalidades disponíveis.

Pode-se dizer que o objetivo foi alcançado.

O software, apesar de simples, possui diversas funções que agregam a experiência de utilizar uma aplicação de qualidade.

O *back-end* foi desenvolvido com as melhores práticas de *Clean Code* e Orientação a Objetos, usufruindo ao máximo do que a linguagem e bibliotecas nos proporcionam.

A parte visual da aplicação, apesar de bem simples, cumpre perfeitamente seu papel, sendo clara, objetiva e agradável.

A usabilidade do software torna simples a experiência de usuário, induzindo e guiando-o intuitivamente através de um layout e organização de páginas lógico e fácil de identificar.

Este produto representa a compreensão das principais ideias apresentadas durante toda a graduação, a condensação dos anos de aprendizado junto à instituição, das cadeiras mais gerais até as mais específicas, das mais teóricas às mais práticas.

Como propostas para futuras melhorias, a inclusão de métodos mais reais de pagamento, como integração com serviços de bancos, empresas de pagamentos, buscar endereços nos serviços dos Correios e também a geração de Nota Fiscal Eletrônica são possíveis e bem vindas.

Além disso, tornar a aplicação um módulo de um sistema maior, como um ERP, também é bem visto.

O software está aberto a críticas, melhorias, inclusões e qualquer tipo de ideia que contribua para a sua melhoria, e caso aplicado, tende a melhorar consideravelmente com o *feedback* de reais usuários.

**5. REFERÊNCIAS**

BOOCH, G.; RUMBAUGH, J.; JACOBSON, I. **UML**. Guia do Usuário. 2a Ed. Rio de Janeiro: Editora Elsevier, 2005.

CALLEGARO, M.; OLIVEIRA, L. **Desenvolvimento de Software**: Sistema Gerenciador de Provas Online e Offline. 2016. 52 páginas. Trabalho de Conclusão de Curso. FAETERJ, Paracambi/RJ.

CARRION, L. **Marketing B2B**: Definição, Tipos e Exemplos de Modelos de Negócios. 2018. Disponível em <https://blog.mailrelay.com/pt/2018/09/20/marketing-b2b>, acessado em 11/06/2019 às 22:43.

CHELI, L. A. **E-Commerce Rede Onix de Farmácias**. 2012. 69 páginas. Trabalho de Conclusão de Curso. Faculdades Integradas do Vale do Ivaí, Ivaiporã/PR.

FLANAGAN, D. **Javascript**. O Guia Definitivo. 6a Ed. São Paulo: Editora Bookman, 2013.

MARINHO, J. J. **A Obscura Diferença entre Requisitos Funcionais e Não Funcionais**. Disponível em https://www.tiespecialistas.com.br/analise-e-levantamento-de-requisitos-em-historias-em-quadrinhos-parte-2-obscura-diferenca-entre-requisitos-funcionais-e-requisitos-nao-funcionais/, acessado em 21/06/2019 às 11:55.

MEIRELLES, F. S. **Pesquisa do Uso de Tecnologia nas Empresas**. Fundação Getúlio Vargas, 2018. Disponível em https://eaesp.fgv.br/sites/eaesp.fgv.br/files/pesti2018gvciappt.pdf, acessado às 12/06/2019 às 09:01.

MULLER, J. **Mydevresume**:Uma Proposta De Um Sistema Para Exposição Da Trajetória Profissional De Desenvolvedores De Software. 2016. 58 páginas. Trabalho de Conclusão de Curso. FAETERJ, Paracambi/RJ.

SILVA, M. S. **HTML 5**. A Linguagem de Marcação que Revolucionou a Web. 2a Ed. São Paulo: Editora Novatec, 2014.

SCHILDT, H. **Java para Iniciantes**. 6a ed. Porto Alegre: Editora Bookman, 2015.

SOMMERVILLE, Ian. Engenharia de Software: 9a ed. Rio de Janeiro: Prentice-Hall, 2011.

STALLMAN, R.; MANCE, E. **Declaração Pessoal de Richard Stallman e Euclides Mence**. Movimento do Software Livre. Disponível em http://stallman.org/solidarity-economy.pt.html. Acessado em 08/05/2019 às 07:44.

Panorama Mercado de ERP no Brasil. Portal ERP, 2017. Disponível em https://portalerp.com/portal-divulga-resultados-pesquisa-20172018, acessado em 11/06/2019 às 19:32.