

Instituto Federal de Educação Ciência e Tecnologia da Paraíba

Curso de Análise e Desenvolvimento de Sistemas

João Marcos Ferreira

*Sudo Loja: Sistema de Gerenciamento de Conteúdo
Voltado ao E-Commerce*

Cajazeiras/PB

2016

João Marcos Ferreira

*Sudo Loja: Sistema de Gerenciamento de Conteúdo
Voltado ao E-Commerce*

Monografia apresentada ao Curso de Análise e Desenvolvimento de Sistemas do IFPB, como requisito para a obtenção do grau de TECNÓLOGO em Análise e Desenvolvimento de Sistemas.

Orientador: Prof. Msc. Francisco Paulo de Freitas Neto

Cajazeiras/PB

2016

João Marcos Ferreira

*Sudo Loja: Sistema de Gerenciamento de Conteúdo
Voltado ao E-Commerce*

Monografia apresentada ao Curso de Análise e Desenvolvimento de Sistemas do IFPB, como requisito para a obtenção do grau de TECNÓLOGO em Análise e Desenvolvimento de Sistemas.

Aprovado em dd/MM/2016

BANCA EXAMINADORA

Francisco Paulo de Freitas Neto

Orientador

Examinador 1

Examinador

Examinador 2

Examinador

Aos meus pais e irmãos.
Aos meus amigos

Agradecimentos

Primeiramente a Deus. O que seria de nós sem Ele?

Aos meus pais que sempre me incentivaram e ajudaram, sem eles não teria chegado até aqui.

Ao meu Orientador Francisco Paulo de Freitas Neto pela paciência, disponibilidade e por prestar toda a orientação e esclarecimentos necessários.

Aos meus tios Isabel e Alonso por me acolherem durante todos esses dias.

À minha namorada por estar sempre ao meu lado me encorajando

Aos meus amigos e todos os meus parentes pelo apoio.

Resumo

Na atualidade muitas pessoas preferem fazer suas compras em casa usando seu computador ou *smartphone*, ao invés de se deslocarem até uma loja ou supermercado. Com um número de pedidos no *e-commerce* aumentando cada vez mais e movimentando quantias cada vez maiores, percebe-se que é um mercado em ascensão. Os comerciantes da região ainda desconhecem ferramentas que facilitam a criação e gestão de uma loja virtual. Este trabalho apresenta uma ferramenta capaz de criar, gerenciar e personalizar uma loja virtual. O uso desta ferramenta trará benefícios a aqueles interessados em vender na Internet, pois poderão com facilidade, sem burocracias, sem custos elevados, e sem precisar ter conhecimentos avançados sobre programação, criar e gerir seu próprio *e-commerce*.

Palavras-chaves: Sistemas de Gerenciamento de Conteúdo. Loja Virtual. Vendas.

Abstract

Nowadays many people prefer to do their shopping at home using your computer or smartphone, instead of moving to a store or supermarket. With a number of requests in e-commerce continually increasing and moving increasingly larger amounts, it is clear that it is a growing market. Traders in the region are still unaware of tools that facilitate the creation and management of a virtual store. This paper presents a tool to create, manage and customize a virtual store. Use of this tool will benefit those interested in selling on the Internet, they may easily, without bureaucracy, without high costs and without having advanced knowledge about programming, create and manage your own e-commerce.

Keywords: Content Management System. Virtual Store. Sales.

*“Pequenas oportunidades são muitas
vezes o começo de grandes
empreendimentos.”*

Demóstenes

Lista de Figuras

Figura 1	Consumidores únicos ativos	13
Figura 2	Total de pedidos realizados no <i>e-commerce</i> no Brasil	14
Figura 3	Total de faturamento do <i>e-commerce</i> no Brasil	14
Figura 4	Satisfação e fidelização de clientes	15
Figura 5	Estimativa de faturamento do <i>e-commerce</i> para 2016	16
Figura 6	Fluxo do <i>easyProcess</i>	22
Figura 7	Modelo em camadas do Sudo Loja	34
Figura 8	Diagrama de Componentes do Sistema	38
Figura 9	Diagrama de classes US03	41
Figura 10	Diagrama de sequência - atualizar informações do Produto	42
Figura 11	Ficha de edição de Produto	43
Figura 12	Página do estoque de uma loja do sistema	43
Figura 13	Visualização de produtos cadastrados	53
Figura 14	Cadastro de produtos	53
Figura 15	Visualização de pedidos	54
Figura 16	Painel de Personalização	54
Figura 17	Controle de estoque	55
Figura 18	Produto e seus relacionamentos	74
Figura 19	Pedido e seus relacionamentos	75
Figura 20	Pedido e seus relacionamentos	75
Figura 21	Endereço e seus relacionamentos	76
Figura 22	Usuário da loja e seus relacionamentos	77
Figura 23	Loja e seus relacionamentos	77

Figura 24	Lojista e seus relacionamentos	78
Figura 25	Produto e seus relacionamentos	78
Figura 26	Categoria de Produto e seus relacionamentos	79
Figura 27	Variação de produto e seus relacionamentos	79
Figura 28	Comentários e seus relacionamentos	80

Lista de Quadros

Quadro 1	Definição de Papéis	27
Quadro 2	<i>User Stories</i> levantadas	28
Quadro 3	Funcionalidades das ferramentas	31
Quadro 4	Alocação de Tarefas - US03	32
Quadro 5	Teste de aceitação - US03	33
Quadro 6	Resumo dos usuários	51
Quadro 7	Resumo das Funcionalidades	52
Quadro 8	Alocação de Tarefas - US01	56
Quadro 9	Teste de aceitação - US01	56
Quadro 10	Alocação de Tarefas - US01	57
Quadro 11	Teste de aceitação - US02	58
Quadro 12	Alocação de Tarefas - US03	58
Quadro 13	Teste de aceitação - US03	59
Quadro 14	Alocação de Tarefas - US04	60
Quadro 15	Teste de aceitação - US04	61
Quadro 16	Alocação de Tarefas - US05	62
Quadro 17	Teste de aceitação - US05	63
Quadro 18	Alocação de Tarefas - US06	63
Quadro 19	Teste de aceitação - US06	64
Quadro 20	Alocação de Tarefas - US07	64
Quadro 21	Teste de aceitação - US07	65
Quadro 22	Alocação de Tarefas - US08	65
Quadro 23	Teste de aceitação - US08	66

Quadro 24	Alocação de Tarefas - US09	67
Quadro 25	Teste de aceitação - US09	68
Quadro 26	Alocação de Tarefas - US10	69
Quadro 27	Teste de aceitação - US10	70
Quadro 28	Alocação de Tarefas - US11	70
Quadro 29	Teste de aceitação - US11	71
Quadro 30	<i>Release</i> 01	73
Quadro 31	<i>Release</i> 02	73
Quadro 32	<i>Release</i> 03	73
Quadro 33	<i>Release</i> 04	73
Quadro 34	<i>Release</i> 05	73

Sumário

1	Introdução	12
1.1	Motivação	16
1.2	Objetivos	18
1.2.1	Objetivo Geral	18
1.2.2	Objetivos Específicos	18
1.3	Organização do Documento	18
2	Fundamentação Teórica	19
2.1	CMS	19
2.2	<i>E-commerce</i>	20
2.3	YP	21
2.4	Bancos de dados NoSQL	23
3	Metodologia	26
3.1	Atividades	26
3.2	Etapas do YP	26
3.2.1	Definição de Papéis	27
3.2.2	Conversa Com o Cliente	27
3.2.3	Inicialização	27
3.2.4	Planejamento de Releases	29
4	Sudo Loja	30
4.1	Ferramentas Relacionadas	30
4.2	Análise	31

4.3	Projeto Arquitetural	33
4.3.1	Camada de Apresentação	34
4.3.2	Camada de Serviços	35
4.3.3	Camada de Persistência	36
4.3.4	Diagrama de Componentes	37
4.3.5	subsection name	38
4.3.6	Tecnologias Utilizadas	38
4.3.7	Implantação	40
4.4	Diagrama de Classes	41
4.5	Implementação	42
5	Considerações Finais	44
5.1	Trabalhos futuros	44
	Referências Bibliográficas	46
	Apêndices	49

1 Introdução

Comprar na Internet traz muitas vantagens para um consumidor. É possível fazer compras sempre que quiser e à hora que desejar, no conforto da sua casa; Não existem filas; As compras são feitas sem a utilização de dinheiro físico, proporcionando mais segurança ao cliente, Além de diversas outras vantagens.

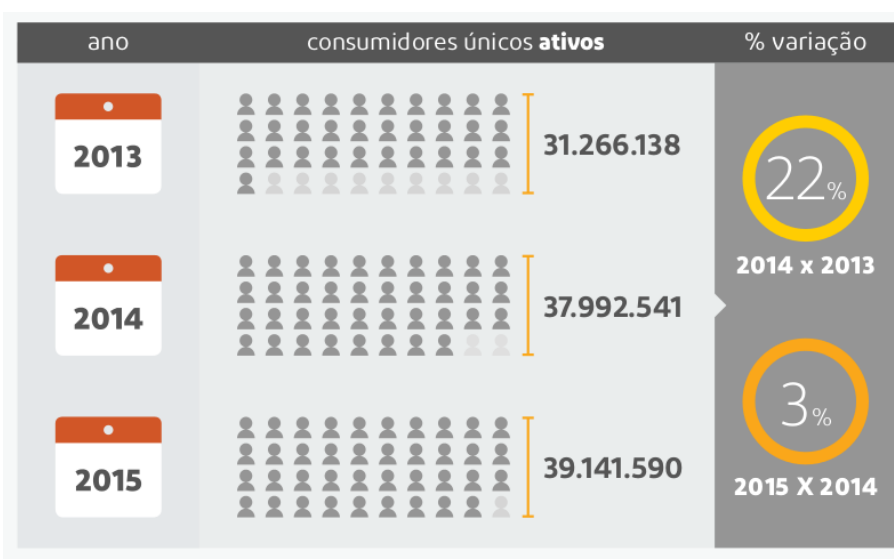
Para os comerciantes as vantagens também são muitas: o produto fica acessível aos clientes 24 horas por dia, todos os dias da semana; pessoas de todos os lugares podem ver o produto; não é necessário um espaço físico para expor os produtos; custo de investimento inicial mínimo; sem risco de assalto, entre outras.

Dados indicam que os consumidores virtuais demonstram altos índices de intenção de compra, ou seja, continuam consumindo apesar do estabelecimento de um cenário de crise, e principalmente na Internet, onde os preços geralmente são mais baratos, se comparados aos do varejo tradicional (E-BIT, 2015).

O *WebShoppers* E-BIT (2015), relatório sólido e respeitado sobre o comércio eletrônico anualmente publicado, no qual são analisadas a evolução do *e-commerce*, tendências, estimativas, mudanças de comportamento e preferências dos e-consumidores, apresentou boas estatísticas sobre o *e-commerce* no Brasil. A seguir serão apresentados dados em forma de figuras e gráficos.

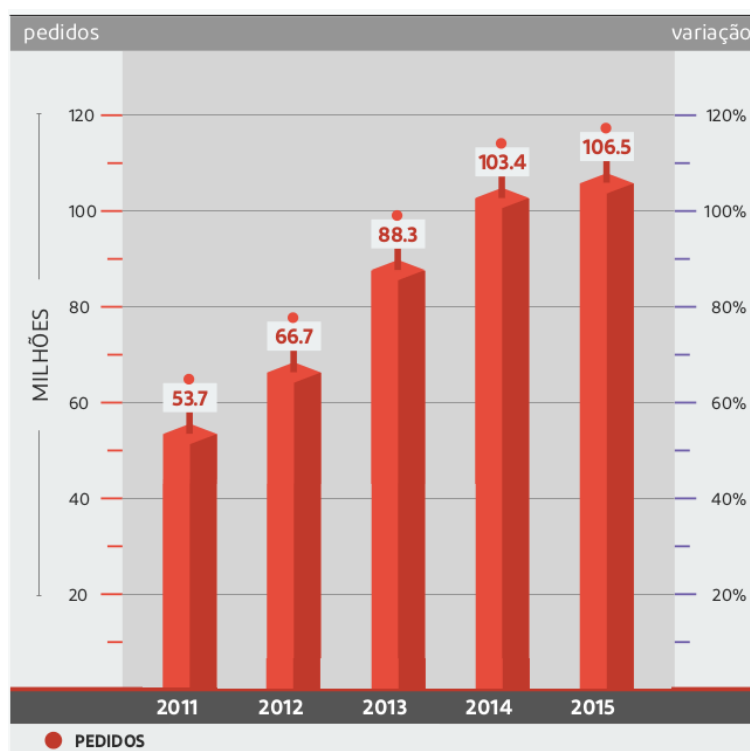
A figura 1 apresenta a quantidade de e-consumidores ativos (pessoas que efetivaram pelo menos uma compra virtual, ao longo de 2015). Percebe-se um crescimento, se comparada com anos anteriores. Em 2015, esse número chegou ao total de 39,1 milhões de consumidores.

Figura 1: Consumidores únicos ativos



Fonte: E-BIT (2015) - Relatório WebShoppers

O número de pedidos e o faturamento do comércio eletrônico também foram notavelmente altos. Como mostra a figura 2, com um total de 106,2 milhões em 2015, o incremento no número de pedidos no mercado brasileiro foi de 3%, em relação a 2014. O faturamento do comércio eletrônico foi de R\$ 41,3 bilhões. O número representa um crescimento nominal de 15,3%, em relação a 2014, quando as vendas somaram um total de R\$ 35,8 bilhões como pode ser visto na figura 3.

Figura 2: Total de pedidos realizados no *e-commerce* no Brasil

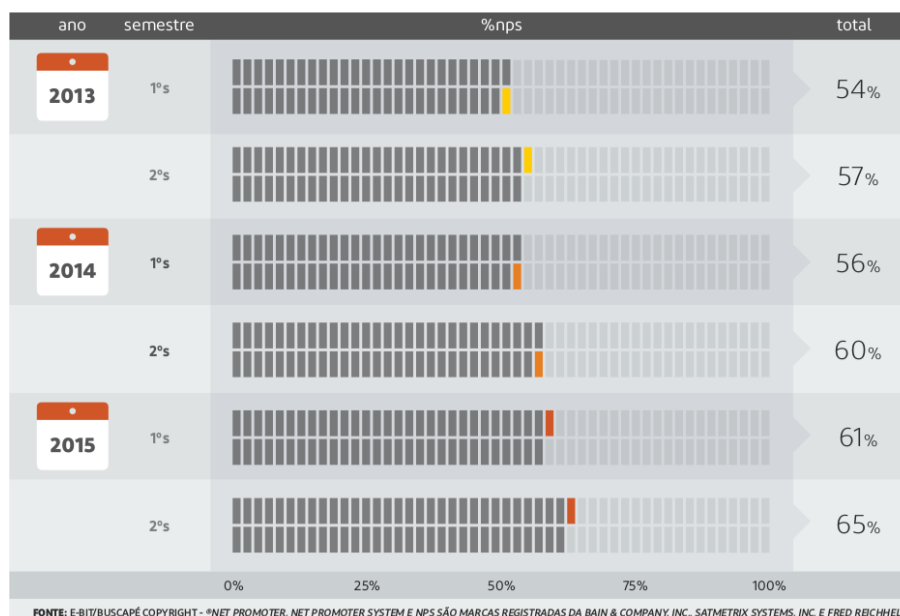
Fonte: E-BIT (2015) - Relatório WebShoppers

Figura 3: Total de faturamento do *e-commerce* no Brasil

Fonte: E-BIT (2015) - Relatório WebShoppers

O relatório também destaca dados sobre a satisfação dos clientes em realizar suas compras na internet. O Net Promoter Score (NPS) é um indicador que mensura a satisfação e a fidelização dos clientes. No balanço geral do ano, o NPS apresentou o melhor resultado. A figura 4 apresenta esse dados.

Figura 4: Satisfação e fidelização de clientes

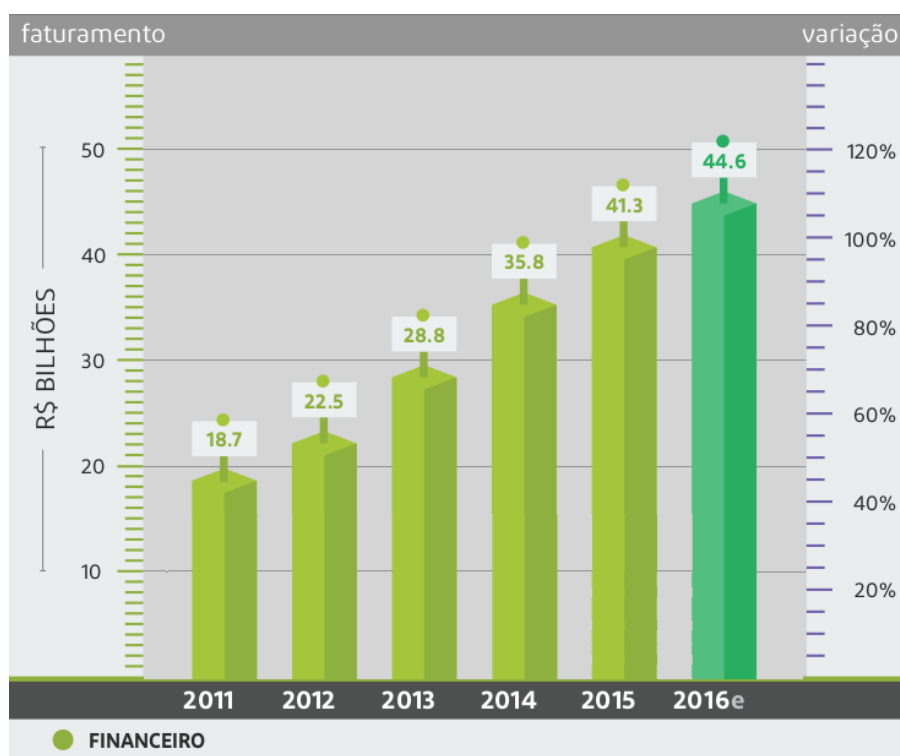


Fonte: E-BIT (2015) - Relatório WebShoppers

Vitor Augusto Meira França, economista da Fecomercio-SP principal entidade sindical paulista dos setores de comércio e serviços. Responsável por administrar, no Estado, o Serviço Social do Comércio (Sesc) e o Serviço Nacional de Aprendizagem Comercial (Senac) afirma o seguinte:

Diante de um quadro de instabilidade política, inflação alta, taxas de juros elevadas, escassez de crédito, aumento do desemprego e consequente conservadorismo dos consumidores, o varejo brasileiro deve repetir o fraco desempenho do ano passado e registrar nova queda das vendas neste ano. por outro lado, o e-commerce deve apresentar crescimento como ocorreu em 2015. (E-BIT, 2015)

A figura 5 apresenta estimativas relacionadas ao faturamento do e-commerce, para o ano de 2016, no Brasil.

Figura 5: Estimativa de faturamento do *e-commerce* para 2016

Fonte: E-BIT (2015) - Relatório WebShoppers

Diante esses dados sobre o *e-commerce* no Brasil, percebe-se que o comércio eletrônico é um mercado em ascensão.

Entre os empresários que se aventuram nesse ambiente, um dos anseios é ter uma loja que tenha a identidade visual da sua marca, que possibilite formas práticas e seguras de compra pelos clientes e, claro, que seja de fácil manuseio e administração para ele mesmo. Com ferramentas como o Mercado Livre¹, por exemplo, é possível disponibilizar produtos para a venda, mas em um ambiente diferente de uma loja virtual particular.

1.1 Motivação

Existem muitas ferramentas que facilitam a criação e gestão de lojas virtuais, porém algumas delas não são populares e as vezes desconhecidas, tais ferramentas poderiam ajudar a expandir um negócio ou criar uma oportunidade para aqueles que desejam empreender através de vendas na Internet.

¹<https://www.mercadolivre.com.br>

Segundo Felipini (2012), *Um e-commerce pode ser implantado aos poucos e testado*. Diferentemente de uma empresa tradicional, em que o início das operações ocorre somente com o empreendimento totalmente estruturado, um negócio na Internet pode ser implantado em etapas, o que dilui o investimento e facilita a correção de erros. Por exemplo, um estabelecimento de vendas só receberá seu primeiro cliente após a loja estiver totalmente pronta. Na Internet, você pode montar um site de conteúdo, com ou sem sua marca definitiva, testar a aceitabilidade de seu modelo de negócio e produtos, avaliar a visita e, somente depois, começar a vender. Dessa forma, mesmo aqueles que ainda não possuem um negócio, poderão investir em um *e-commerce*.

Os comerciantes já fixados no mercado e interessados em colocar seu negócio na rede, procuram por empresas especializadas em desenvolvimento de softwares para desenvolver seu site de vendas. O problema nisso é que o custo pode ser alto. Jalote (2012) afirma: *O software é caro porque torna se uma atividade difícil e trabalhosa de ser realizado pelo engenheiro de software*. Outro problema relacionado ao desenvolvimento de software no geral, isso inclui uma loja virtual, é o tempo. O que gera insatisfação dos clientes, pela demora no cumprimento dos prazos (PRESSMAN, 2009).

Além desses, um outro problema é a manutenção do site. Segundo Sommerville (2007) manutenção de software é a modificação de um programa após ter sido colocado em uso. Mudanças por exemplo, em alterar o *layout*² geram custos e podem também ser demoradas.

A Sudo Loja, visa resolver esses problemas oferecendo um CMS (do inglês *Content Management System*), capaz de criar, gerenciar e tornar possível a personalização de uma loja virtual. O CMS proposto busca facilitar a criação e gestão de um *e-commerce*, promovendo uma entrada rápida ao mercado virtual à pequenas e médias empresas. Podem ser vistos vários sistemas que funcionam dessa maneira, como por exemplo: a Tray commerce³ e Loja integrada⁴. Na Seção 4.1 será apresentado um comparativo entre estas e outras ferramentas.

²Aspecto visual das páginas do site

³<https://www.tray.com.br/>

⁴<https://lojaintegrada.com.br/>

1.2 Objetivos

Esta seção apresenta os objetivos que direcionarão a construção do sistema.

1.2.1 Objetivo Geral

Esse trabalho tem como objetivo principal o desenvolvimento de uma ferramenta para criação de lojas virtuais, onde um usuário cadastrado poderá criar um ou mais lojas, personalizar seu *layout* e disponibilizar produtos para a venda.

1.2.2 Objetivos Específicos

- Prover uma alternativa de *e-commerce* para a região.
- Incentivar o uso de novas soluções em TI nas pequenas e médias empresas comerciais da região.
- Facilitar a entrada de novos comerciantes no mercado virtual.
- Desenvolver um painel administrativo para que os usuários possam acompanhar e gerenciar sua loja.
- Tornar possível a criação de mais de uma loja com a mesma conta de usuário.
- Disponibilizar um módulo para personalização das lojas.

1.3 Organização do Documento

A Seção 2 apresenta a fundamentação teórica para o desenvolvimento deste trabalho. Na Seção 3 é descrita a metodologia adotada para execução do projeto. A Seção 4 descreve a solução proposta no trabalho, um comparativo entre ferramentas do gênero e também as etapas de análise, projeto, implementação e validação do sistema. A Seção 5 apresenta as considerações finais do trabalho, assim como discussões sobre trabalhos futuros. Por fim, os artefatos gerados no decorrer do desenvolvimento do trabalho podem ser vistos nos Apêndices.

2 Fundamentação Teórica

Essa seção discutirá sobre alguns assuntos necessários para obter um melhor entendimento do trabalho.

2.1 CMS

Sistemas de Gerenciamento de Conteúdo (SGC) ou Content Management System (CMS), segundo Mercer (2008) são softwares que facilitam a criação, organização, manipulação e remoção de dados em forma de imagens, documentos, scripts, textos, etc. Já Bárcia (2008), diz que um CMS, é uma plataforma de gestão de conteúdos, ou seja, um sistema que integra ferramentas que permitem criar, editar e publicar conteúdo em tempo real, onde os utilizadores manipulam uma interface sem terem a necessidade de saber programar. Bárcia (2008) ainda ressalta que os gerenciadores também dispensam o uso de programação, facilitando dessa forma a gestão dos dados e o acesso às funcionalidades da ferramenta.

O uso de um CMS pode facilitar o trabalho dos administradores de aplicações, que envolvem muitas atualizações no seu conteúdo, como por exemplo, blogs, portais corporativos¹. *Os benefícios ligados a adoção de um CMS incluem desde a redução do custo de atualização dos conteúdos nos web sites até o aumento da eficiência das equipes de TI* (PEREIRA; BAX, 2010).

Em linhas gerais, um CMS permitiria administrar conteúdos em meio digital. E para o caso particular desse trabalho, um CMS permitiria gerenciar os conteúdos de uma loja virtual.

Em outras palavras, um CMS é uma ferramenta que permite a um editor criar e publicar qualquer tipo de informação em uma página web. Geralmente, um CMS trabalha manipulando um banco de dados, de modo que o editor simplesmente atualiza este banco, incluindo nova informação ou editando a existente.

¹ Instrumento de gestão de informação e de conhecimento

Contudo, o mais interessante, é que essa ferramenta é feita de tal maneira que mesmo aqueles que nunca ouviram falar de JAVA, PHP, MySQL, Javascript ou qualquer outra linguagem voltada para web poderá utilizá-la, inclusive esta é sua principal função, tornar acessível a todos a sua presença na Internet através de um site, facilitando a inserção de textos, de comentários e dezenas de outras funcionalidades, de acordo com as características de cada aplicação. Alguns Exemplos de CMS podem ser vistos a seguir.

- *Drupal* - É uma plataforma de gerenciamento de conteúdo de código aberto, usado em milhares de web sites e aplicações. Ele é desenvolvido, usado, e apoiado por uma comunidade ativa e diversificada de pessoas ao redor do mundo (DRUPAL, 2015).
- *OpenText* 2015 - O primeiro sistema CMS comercial que apareceu no mercado. A OpenText é líder em Gestão de Informação Corporativa. Seus produtos de gerenciamento de conteúdo permitem uma coleta mais eficiente de todos os tipos de informação - estruturada e não estruturada - e fornecem essa informação em contexto por qualquer aplicativo, plataforma ou processo (OPENTEXT, 2015).
- *Wordpress* - Sistema muito popular e bastante usado pelos *bloggers*. WordPress é um software web que você pode usar para criar um site, blog, ou app (WORDPRESS, 2015).

2.2 E-commerce

Segundo Kalakota e Whinston (1996), o Comércio Eletrônico (CE) pode ser definido como sendo a compra e a venda de informações, produtos e serviços através de redes de computadores. Bloch, Pigneur e Segev (1996) estenderam esta definição incluindo que CE (Comércio Eletrônico) é o suporte para qualquer tipo de transações de negócio sobre uma infraestrutura digital.

Kalakota e Whinston (1996 apud ALBERTIN, 1999) há muito tempo, consideravam que o ambiente tradicional de negócio estava mudando rapidamente, com os consumidores e negócios procurando flexibilidade para mudar os parceiros

de negócio, plataformas, carreiras e redes. O *e-commerce* começava a se disseminar.

O comércio eletrônico ou *e-commerce*, é aplicável a qualquer tipo de transação comercial que pode envolver compra, venda, transferência ou troca de produtos, serviços ou informações por meio de redes de computadores, incluindo a Internet (TURBAN, 2008).

Existem diferenças entre *e-commerce* e *e-business*. *E-business* é definido como o uso das tecnologias da informação para executar funções de negócios. *E-business* é, portanto, um termo mais amplo que inclui *e-commerce* (GORDON, 2011).

Dentre os modelos existentes de *e-business*, Asfoura, Jamous e Salem (2009) destacam três tipos principais que incluem o e-commerce em sua execução:

- **Empresa para Empresa (B2B – Business to Business):** Engloba as negociações de bens ou serviços que acontecem entre empresas;
- **Empresa para Consumidor (B2C – Business to Consumer):** Tipo de comércio mais conhecido, onde a empresa faz o negócio diretamente com os consumidores finais;
- **Consumidor para Consumidor (C2C – Consumer to Consumer):** Engloba todas as transações que acontecem entre consumidores finais, geralmente intermediadas por uma terceira entidade;

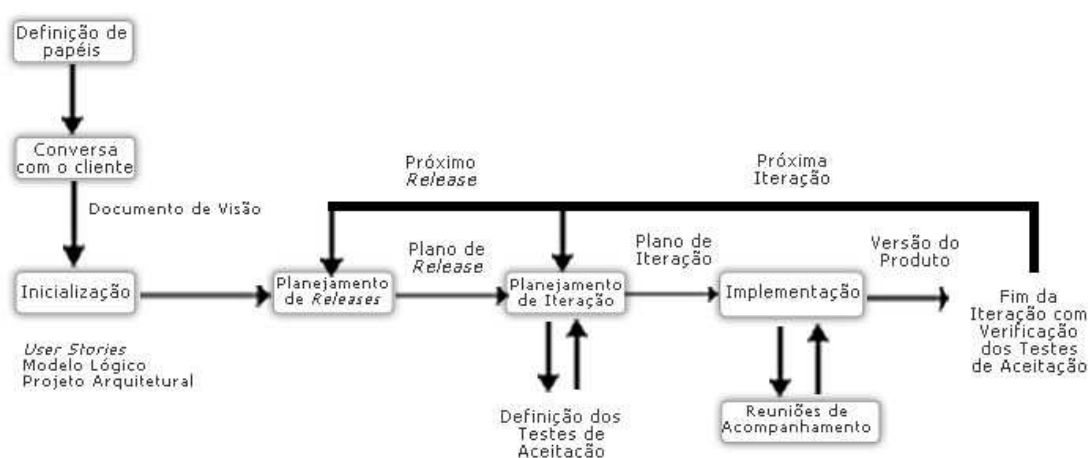
O sistema que será apresentado se encaixa no modelo de negócio B2C (*Business to Consumer*), onde o lojista fará o negócio diretamente com os consumidores finais.

2.3 YP

Quando se trabalha na elaboração de um produto os sistema, é importante seguir uma série de passos previsíveis - um roteiro que ajude a criar um resultado de alta qualidade e dentro de prazo estabelecido. O roteiro é denominado “Processo de Software” (PRESSMAN, 2009).

O *easyProcess*, comumente chamado de YP. É um processo de software criado com o objetivo de sanar as dificuldades dos alunos em se adaptar aos processos já existentes para o uso na academia. Voltado a projetos de pequeno escopo, o YP foca-se na aprendizagem do processo com alguns elementos qualificadores: uma boa produtividade, bom uso do ferramental de apoio e geração mínima de artefatos, além de se buscar uma consolidação do entendimento das práticas e conceitos da Engenharia de Software (GARCIA et al., 2004). O fluxo básico do YP está ilustrado na figura 6.

Figura 6: Fluxo do *easyProcess*



Fonte: (GARCIA et al., 2004)

A primeira etapa do processo consiste na **Definição de papéis**. O YP sugere os seguintes papéis: cliente, usuário, testador, desenvolvedor e gerente, podendo uma mesma pessoa desempenhar mais de um papel dentro do processo, principalmente quando se tratam de equipes de desenvolvimento pequenas (GARCIA et al., 2004).

Em seguida deve ser realizada uma **Conversa com o cliente**. Aqui é onde as informações sobre o escopo do problema são adquiridas. A partir de então, a equipe encontra-se apta a gerar o documento de visão, que após ser validado pelo cliente, funciona como um acordo de trabalho entre cliente e equipe de desenvolvimento (GARCIA et al., 2004).

Na fase de **Inicialização** o cliente define as *User Stories*² e são elaborados o projeto arquitetural e o modelo lógico de dados, este último apenas se

²Funções que o sistema deve desempenhar e que são definidas pelo cliente e pelo desenvolvedor

necessário. O cliente deve priorizar as User Stories e a equipe deve fazer uma estimativa inicial do tempo para implementação de cada uma delas. Baseado nessa estimativa pode-se então verificar a viabilidade de desenvolvimento do projeto no escopo e tempo definidos (GARCIA et al., 2004).

Parte-se então para o **Planejamento** fase composta por dois planos, o de release e o de iteração. Ambos possuem tempo fixo com variação de escopo permitida. Tratando-se do ambiente acadêmico são sugeridos três releases, cada um com duas iterações de duas semanas, por semestre letivo (GARCIA et al., 2004).

Para a **Implementação**, o processo prega o uso de algumas práticas, tais como: *Design* Simples, Padrões de Codificação, Padrões de Projeto, Refatoramento e Propriedade Coletiva de Código, a fim de produzir um código com mais qualidade (GARCIA et al., 2004).

O andamento do processo deve ser coordenado pelo gerente através da **Reunião de Acompanhamento** semanal que visa recolher e analisar métricas.

2.4 Bancos de dados NoSQL

Os bancos de dados NOSQL surgiram como uma solução para a questão da escalabilidade no armazenamento e processamento de grandes volumes de dados na Web 2.0. No início, grandes empresas enfrentando esse tipo de problema criaram suas próprias soluções e publicaram alguns artigos, mas sem usar ainda o nome NOSQL. O nome só surgiu alguns anos depois, em 2009, quando algumas novas empresas da Web 2.0 e a comunidade de software livre e código aberto começaram a desenvolver novas opções de bancos de dados, inspiradas nas ideias que apareceram no artigos (DIANA; GEROSA, 2010).

A web é composta por uma grande quantidade de dados semi estruturados e crús, como as páginas web (cuja estrutura descrita no documento HTML expressa muito pouco sobre o significado do conteúdo do documento) e conteúdo multimídia (imagens, sons e vídeos). Ao reconhecer a natureza particular dos dados é possível criar soluções otimizadas para eles, ao invés de se tentar estruturá-los (DIANA; GEROSA, 2010).

A constatação de que os dados na web não são estruturados é um dos fatores que favoreceram o surgimento de tecnologias de gerenciamento de dados diferentes das tradicionais. Os tipos mais comuns de bancos de dados NOSQL são:

- **Bancos de dados orientados a documentos** - Os documentos de bancos de dados orientados a documentos são coleções de atributos e valores, onde um atributo pode ser multi-valorado. Em geral, os bancos de dados orientados a documento não possuem esquema, ou seja, os documentos armazenados não precisam possuir estrutura em comum. É geralmente usado em CMSs, plataformas de blog e análises web.
- **Chave-valor** - Sistemas distribuídos nessa categoria, também conhecidos como tabelas de hash distribuídas, armazenam objetos indexados por chaves, e possibilitam a busca por esses objetos a partir de suas chaves. Tem um modelo de dados simples um mapa/dicionário, permitindo que os clientes recuperem valores por chave (STRAUCH; SITES; KRIHA, 2011). Neto (2015), sugere o uso bancos de dados chave valor para o armazenamento de informações de sessão, perfis de usuários e preferências.
- **Bancos de dados de famílias de colunas** - Otimizado para leitura de dados estruturados, bancos de dados de famílias de colunas são mais interessantes, pois eles guardam os dados contiguamente por coluna em vez de linhas como em bancos relacionais (STONEBRAKER et al., 2005).
- **Bancos de dados de grafos** Diferentemente de outros tipos de bancos de dados NOSQL, esse está diretamente relacionado a um modelo de dados estabelecido, o modelo de grafos. A ideia desse modelo é representar os dados e / ou o esquema dos dados como grafos dirigidos, ou como estruturas que generalizem a noção de grafos (ANGLES; GUTIERREZ, 2008). Muito usado em redes sociais, roteamento baseado em localização e mecanismos de recomendação.

Em relação a SGBD tradicionais, a distribuição dos dados de forma elástica é inviabilizado pois o modelo de garantia de consistência é fortemente baseado no controle transacional ACID (Atomicity, Consistency, Isolation e Durability). Esse tipo de controle transacional é praticamente inviável quando os dados e o

processamento são distribuídos em vários nós. O teorema CAP (Consistency, Availability e Partition tolerance) mostra que somente duas dessas 3 propriedades podem ser garantidas simultaneamente em um ambiente de processamento distribuído de grande porte. A partir desse teorema, os produtos NoSQL utilizam o paradigma BASE (Basically Available, Soft-state, Eventually consistency) para o controle de consistência, o que consequentemente traz uma sensível diminuição no custo computacional para a garantia de consistência dos dados em relação a SGBD tradicionais (VIEIRA et al., 2012).

3 Metodologia

Esta seção aborda a metodologia usada para o desenvolvimento desse trabalho

3.1 Atividades

Esta seção apresenta as atividades seguidas para o desenvolvimento deste trabalho, foram elas:

- **Determinação do tema** - Nesta fase, foi verificada o tema foco do trabalho, o qual foi posteriormente avaliado por membros do corpo docente da instituição.
- **Levantamento bibliográfico** - Com o tema estabelecido, foi efetuado um estudo bibliográfico, afim de fundamentar o tema.
- **Leitura e Documentação** - Foi de competência desta fase, a filtragem e entendimento do material encontrado conforme a relevância da publicação.
- **Análise** - Nesta fase as primeiras reuniões foram feitas afim de obter um maior conhecimento das funcionalidades do sistema que seria desenvolvido.
- **Projeto e desenvolvimento** - Fase prática da exploração das referências bibliográficas, para esta etapa o uso do processo de desenvolvimento YP foi adotado e é descrito em detalhes na seção 3.2.
- **Escrita de documentos** - O desenvolvimento do texto foi organizado em capítulos, cada qual abordando diferentes ênfases.

3.2 Etapas do YP

Nesta seção serão abordadas cada etapa realizada durante o processo de desenvolvimento do sistema, seguindo as diretrizes do *easYProcess*, estas são:

Definição de papéis, na Subseção 3.2.1; Conversa com o Cliente, na Subseção 3.2.2; Inicialização, na Subseção 3.2.3; E Planejamento de *Releases*, na Subseção 3.2.4.

3.2.1 Definição de Papéis

Após a etapa de definição de Papéis chega-se ao resultado visto no quadro 1:

Quadro 1: Definição de Papéis

Papel	Stakeholder
Cliente	Francisco Paulo de Freitas Neto
Usuário	Francisco Paulo de Freitas Neto
Testador	João Marcos Ferreira
Desenvolvedor	João Marcos Ferreira
Gerente	João Marcos Ferreira

3.2.2 Conversa Com o Cliente

Nessa etapa foi gerado o Documento de Visão. Pretende-se que ele sirva como ferramenta de auxílio, a evitar problemas mais custosos. Ele apresenta os problemas a serem solucionados, as necessidades dos principais envolvidos, o alcance do projeto e as funcionalidades esperadas do sistema.

Segundo Zadrozny (), o objetivo desse documento é expor as necessidades e funcionalidades gerais do sistema, definindo os requisitos de alto nível em termos de necessidades dos usuários finais.

O Documento de visão produzido durante esta etapa pode ser visto no Apêndice A.

3.2.3 Inicialização

Como sugerido pelo YP, nesta fase foram definidas as *Users Stories* e suas respectivas estimativas de tempo para implementação. As *Users Stories* capturadas podem ser vistas a seguir no quadro 2. Nas seções seguintes uma delas será

apresentada com detalhes e as demais poderão ser vistas no Apêndice B *User Stories* e Testes de Aceitação.

Quadro 2: *User Stories* levantadas

Identificação	Descrição	Estimativa
US01	O sistema deve realizar o gerenciamento dos perfis de clientes logistas.	1 semana.
US02	O sistema deve realizar o gerenciamento dos dados das lojas virtuais.	1 semana.
US03	O sistema deve realizar o gerenciamento do estoque da loja virtual.	2 semanas.
US04	O sistema deve disponibilizar uma página de gerenciamento para os clientes logistas.	2 semanas.
US05	O sistema deve permitir a personalização do Layout da página da sua loja virtual.	3 semanas.
US06	O sistema deve realizar o gerenciamento dos perfis de clientes da loja virtual.	1 semana.
US07	O sistema deve disponibilizar uma página de acompanhamento para os clientes.	1 semana.
US08	Nas páginas dos produtos deve ser possível realizar perguntas ao vendedor.	1 semana.
US09	O sistema deve calcular o frete dos produtos automaticamente.	3 dias.
US10	As páginas dos produtos deve apresentar uma galeria imagens por produto.	3 dias.
US11	O sistema deve concretizar vendas através de diferentes meios de pagamento.	1 semana.

De acordo com Heuser (2009), um modelo lógico de um BD (base de dados) relacional, deve definir quais as tabelas que o banco contém e, para cada tabela, quais os nomes das colunas. O modelo lógico para o DB em questão pode ser visto no apêndice D.

3.2.4 Planejamento de Releases

Durante esta etapa foram definidas as *Releases*, juntamente com a data estimada para o termino de seu desenvolvimento. Esses dados podem ser vistos no apêndice C Plano de Iteração.

4 Sudo Loja

A Sudo Loja é um CMS voltado a construção de lojas virtuais. A ferramenta possibilita o gerenciamento e personalização das páginas, assim como outras capacidades.

Esta seção irá discutir as etapas de análise, projeto e implementação d Sudo Loja.

4.1 Ferramentas Relacionadas

Foram analisadas algumas ferramentas em busca de compreender as funcionalidades do sistema e procurar um diferencial dentre as soluções existentes. A seguir, o quadro 3 mostra um comparativo com as principais ferramentas usadas na criação de lojas virtuais e suas respectivas funcionalidades, comparando com o sistema desenvolvido.

- 1. Magento (www.magento.com)
- 2. Tray Commerce (www.tray.com.br)
- 3. Loja UolHost (www.uolhost.com.br)
- 4. Wix Commerce (pt.wix.com/ecommerce/website)
- 5. Loja Integrada (www.lojaintegrada.com.br)

Quadro 3: Funcionalidades das ferramentas

	1	2	3	4	5	Sudo Loja
Pago	não	sim	sim	sim	sim	não
Gerenciamento de produtos, clientes e pedidos	sim	sim	sim	sim	sim	sim
Personalização do layout	sim	sim	sim	sim	sim	sim
Múltiplas imagens por produto	sim	sim	sim	sim	sim	sim
Site apropriado para dispositivos moveis	sim	não	não	não	não	sim
Controle de estoque	sim	sim	sim	sim	sim	sim
Perguntas na página do produto	não	não	não	não	não	sim

Como o quadro 3 mostra, a Sudo Loja contará com as funcionalidades encontradas nas ferramentas listadas acima e ainda com um diferencial, que será a possibilidade de fazer perguntas ao vendedor sobre determinado produto.

4.2 Análise

A seguir serão apresentados os artefatos produzidos na fase de análise, inerentes a algumas *User Stories* específicas. Para isso foi escolhida a *User Story* US03 (O sistema deve realizar o gerenciamento do estoque da loja virtual).

As *User Stories* alocadas são quebradas em tarefas, e o cliente deve definir os testes de aceitação para cada *User Story*. Para auxílio na gerência foi feito uso da Tabela de Alocação de Tarefas (TAT), na qual especifica-se as *User Stories* envolvidas, tarefas, responsáveis, estimativas de tempo, tempo real consumido e status da tarefa. Nos quadros 4 e 5 é possível ver, respectivamente, a tabela de alocação de tarefas e os testes de aceitação geradas pra a US03. As demais *User Stories* são apresentadas com detalhes no Apêndice B.

Quadro 4: Alocação de Tarefas - US03

User Story 03					
Tarefa	Descrição	Responsável	Estimativa de tempo (horas)	Tempo real (horas)	Status
T1	Implementar Script para salvar um Produto	João Marcos	2	1	Finalizado
T2	Implementar Script para validar informações do produto	João Marcos	2	1	Finalizado
T3	Criar Formulário para cadastro de produto	João Marcos	3	2	Finalizado
T4	Criar formulário para edição de produto	João Marcos	2	1	Finalizado
T5	Criar página para o estoque, exibindo as informações básicas de cada produto com sua respectiva quantidade e link para ficha do produto	João Marcos	4	3	Finalizado
T6	Criar botão para exclusão de um produto na página de estoque	João Marcos	2	1	Finalizado

Quadro 5: Teste de aceitação - US03

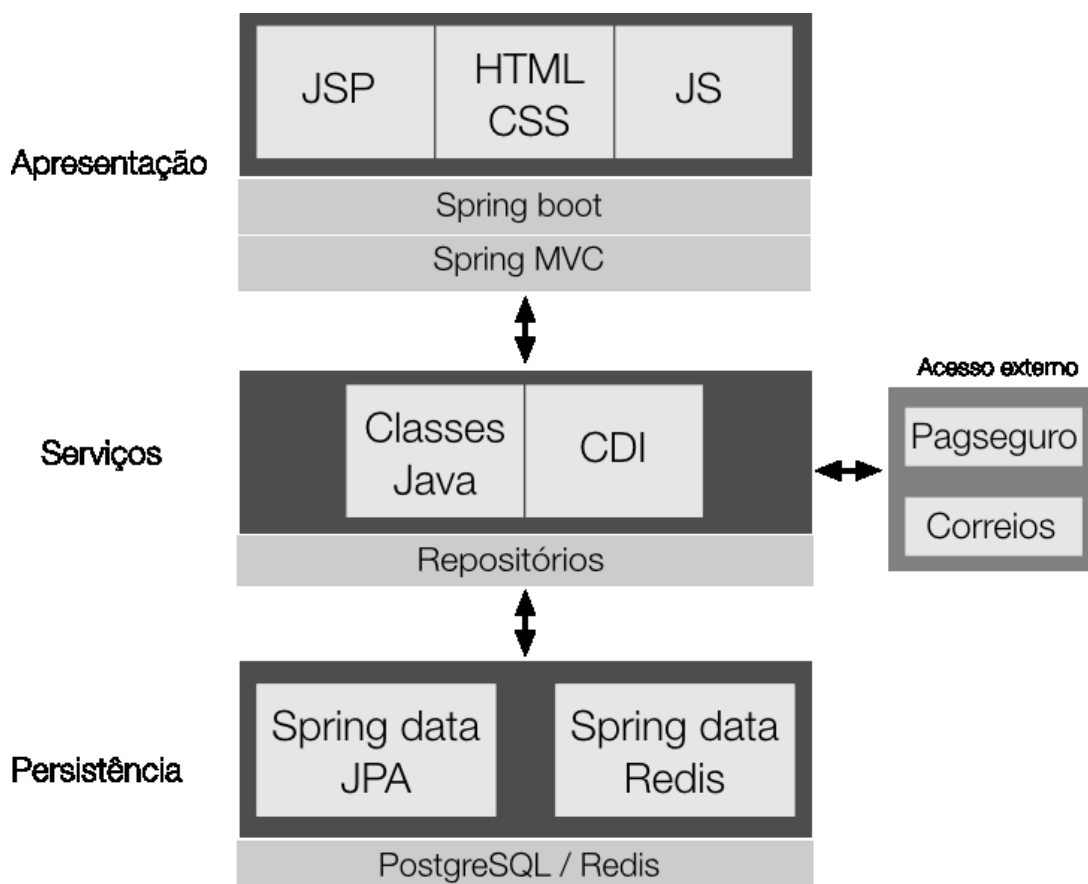
User Story 03		
Testes de aceitação		Status
TA1	Dado que estou na página de cadastro de produto, quando eu preencher o formulário com os dados obrigatórios de forma correta , o produto deve ser salvo e informado uma mensagem de sucesso.	Finalizado
TA2	Dado que estou na página de cadastro de produto, quando eu preencher o formulário com os dados obrigatórios de forma incorreta , o produto não deve ser salvo e uma mensagem de erro deve ser exibida.	Finalizado
TA3	Dado que estou acessando a ficha de edição de produto, quando eu alterar alguma informação do formulário com os dados válidos, o produto deve ser atualizado e uma mensagem de sucesso deve ser exibida.	Finalizado
TA4	Dado que estou acessando a ficha de edição de produto, quando eu alterar alguma informação do formulário com os dados inválidos, tais informações não serão salvas e uma mensagem de erro deve ser exibida.	Finalizado
TA5	Dado que estou na página de estoque, quando eu selecionar a opção excluir este produto, um diálogo de confirmação deve ser exibido. Caso confirme o produto deve ser excluído	Finalizado
TA6	Dado que estou na página de estoque, quando eu selecionar a opção excluir este produto, um diálogo de confirmação deve ser exibido. Caso não confirme o produto não deve ser excluído	Finalizado

4.3 Projeto Arquitetural

O sistema será organizado em 3 camadas, são elas: Apresentação, Serviços e Persistência. As próximas seções abordarão individualmente cada uma delas. Segundo Sommerville et al. (2008), o modelo em camadas de uma arquitetura organiza o sistema em camadas, cada uma das quais fornecendo um conjunto de

serviços, essa abordagem apoia o desenvolvimento incremental de sistemas. À medida que uma camada é desenvolvida, alguns serviços disponibilizados por essa camada podem ser disponibilizados para os usuários. Na Figura 7, pode ser vista uma representação visual de como se dará a organização das camadas, onde cada camada só acessa diretamente suas camadas adjacentes.

Figura 7: Modelo em camadas do Sudo Loja



4.3.1 Camada de Apresentação

A partir desta camada o usuário poderá interagir diretamente com o sistema. Será composta por páginas JSP (*Java Sever Pages*) e HTML(*HyperText Markup Language*, que significa Linguagem de Marcação de Hipertexto), estas que serão estilizadas com o uso de CSS3. Para agilizar o processo de desenvolvimento dessas páginas, será usado o *framework Bootstrap* na versão 3. Com isso, a velocidade na criação das páginas aumentará devido a existência de dezenas de componentes já estilizados e estruturalmente organizados, que poderão ser reaproveitados. Alguns efeitos visuais da página serão implementados com o uso de *JavaScript*. Todo

conteúdo dinâmico dessas páginas serão implementados usando JSP em sua versão 2.1 e para o conteúdo estático será usado HTML.

Esta camada poderá interagir diretamente (e apenas) com sua camada adjacente, a camada de Serviços que será discutida na seção 4.3.2. O estabelecimento desta comunicação será feita utilizando o Spring MVC, a través de controladores. Toda a configuração necessária para a inicialização desta camada é feita através do módulo Spring Boot.

Segundo Souza (1998), um Framework pode ser visto como um projeto genérico em um domínio que pode ser adaptado a aplicações específicas, servindo como um molde para a construção de aplicações.

O Spring Boot, o Spring MVC, assim como o Spring Data fazem parte do Spring Framework, trata-se de um poderoso *framework* de código aberto desenvolvido em Java, para suporte a injeção de dependências, gerenciamento de transações, aplicações web, manipulação de dados, troca de mensagens, testes e muito mais (PIVOTAL, 2015b).

O Spring Boot é usado responsável por toda a configuração da aplicação, foi escolhido por ser ideal para construir aplicações Spring prontos para produção, tornando a tarefa de configuração a mais simples possível.

O Spring MVC é o framework MVC nativo do Spring framework e usado para desenvolver aplicações web em Java, seguindo o padrão MVC de arquitetura.

A junção do Spring Boot e Spring MVC em um projeto torna o desenvolvimento de uma aplicação web em Java bem mais fácil, pois não há mais a necessidade do desenvolvedor ficar se preocupando com arquivos de configuração seja Java ou XML.

4.3.2 Camada de Serviços

Esta camada fornecerá todos os serviços necessários para o funcionamento do sistema. Terá acesso direto às suas camadas adjacentes: Camada de Apresentação e Camada de Persistência. Para isso, fará uso da linguagem de programação Java e uso da especificação CDI (Injeção de Dependência e Contextos, em português) para facilitar o uso de injeção de dependências. É composta por um conjunto de classes

Java, cada qual com suas responsabilidades. Estas classes validarão e manipularão os dados vindo da camada de apresentação. Essas classes que fornecerão os serviços, irão acessar diretamente a camada de persistência, esta que por sua vez irá retornar algum resultado, para que o serviço realize o processamento e retorne para a camada de apresentação.

Para Foldoc (2011), Interface de Programação de Aplicação (português brasileiro), cujo acrônimo API provém do Inglês Application Programming Interface, é um conjunto de rotinas e padrões estabelecidos por um software para a utilização das suas funcionalidades por aplicativos que não pretendem envolver-se em detalhes da implementação do software, mas apenas usar seus serviços.

Esta camada tem acesso a alguns serviços externos e usadas em forma de APIs. O PagSeguro, uma solução de comércio eletrônico para transações comerciais através de pagamentos online ou móveis (UOL, 2016). E a API de cálculo de frete dos correios.

4.3.3 Camada de Persistência

Esta camada será responsável por manter e disponibilizar todos os dados do sistema. Para isto terá auxílio dos módulos Spring Data JPA e Spring Data Redis. Estes módulos automatizarão o processo criação e manipulação dos dados e auxiliarão nas operações CRUD (acrônimo de *Create*, *Read*, *Update* e *Delete* na língua Inglesa) para as quatro operações básicas utilizadas em bases de dados relacionais (RDBMS) criação, consulta, atualização e exclusão de dados.

Segundo KAUFELD (1996), o modelo de banco de dados relacional possui a capacidade de lidar com grandes volumes de informações, eliminando dados redundantes.

No modelo relacional existe a possibilidade de elaboração de um relacionamento lógico entre as informações referentes à espécie e as referentes ao indivíduo, evitando-se a necessidade da repetição de informações e agilizando as consultas feitas às duas fontes de dados (FILHO et al., 2002).

O PostgreSQL (2015) foi escolhido como SGBDR (Sistema Gerenciador de Banco de Dados Relacional) por ter sido o mais abordado durante a período

acadêmico. E o Redis (2016) foi escolhido para a manipulação de preferências de usuário por ser um banco extremamente rápido e simples de usar e open source. *O Redis fornece operações de relâmpago em conjuntos de dados em memória, e também torna mais fácil para persistir no disco* (CARLSON, 2013).

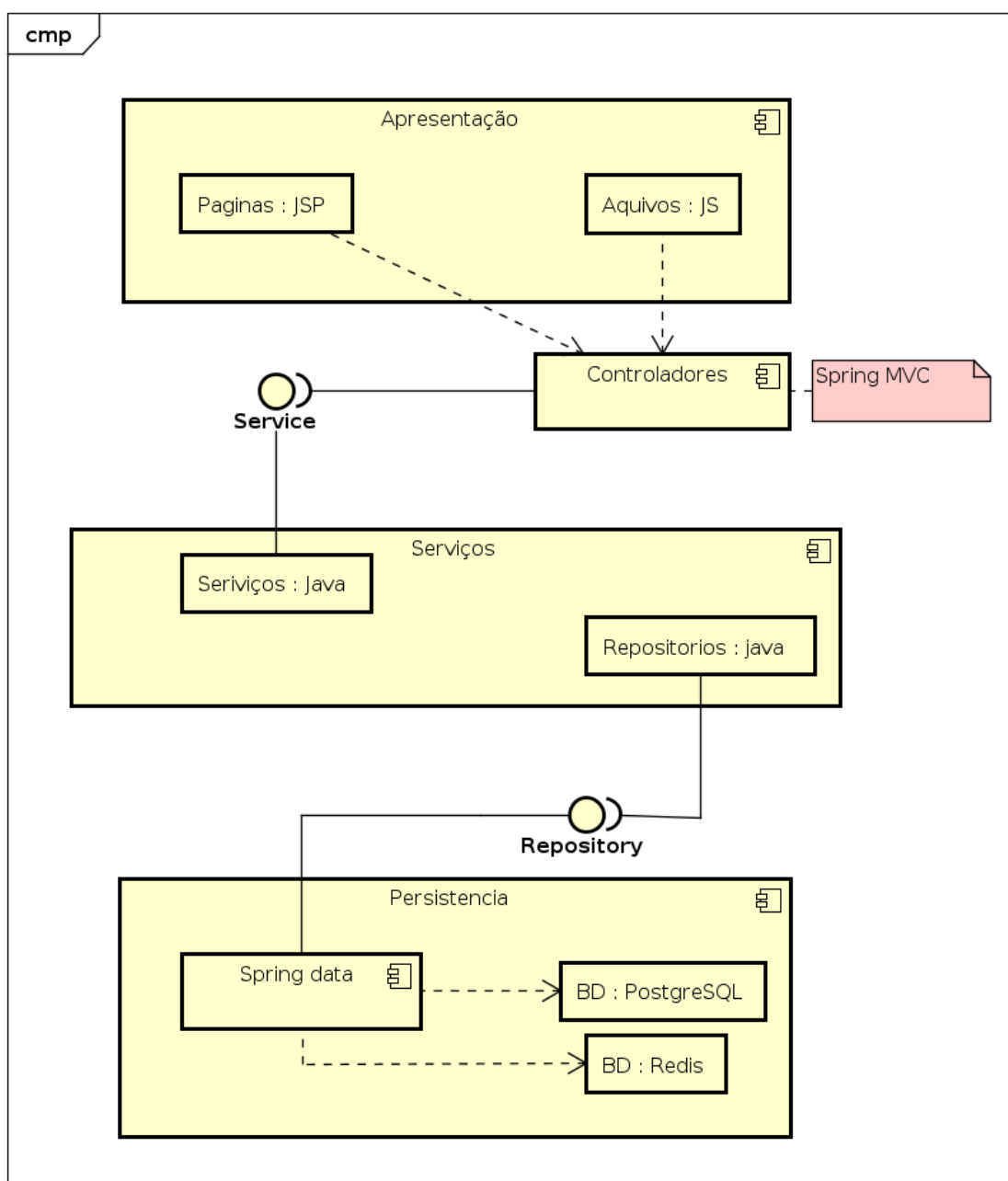
4.3.4 Diagrama de Componentes

Os diagramas estruturais da Linguagem de Modelagem Unificada (do inglês, UML - Unified Modeling Language), existem para visualizar, especificar, construir e documentar os aspectos estáticos do sistema (BOOCH; RUMBAUGH; JACOBSON, 2006).

Booch, Rumbaugh e Jacobson (2006) afirmam que esses aspectos podem ser considerados uma representação de seu esqueleto e estrutura relativamente estáveis. O diagrama de componentes por ser um diagrama estrutural, é útil para representar a arquitetura de um sistema.

A figura 8 apresenta o diagrama de componentes do sistema, e possibilitará uma melhor visualização de sua arquitetura, mostrando como cada camada interage com as outras.

Figura 8: Diagrama de Componentes do Sistema



4.3.5 subsection name

4.3.6 Tecnologias Utilizadas

Está seção apresentará com detalhes todas as tecnologias que serão utilizadas no sistema.

- **HTML** - No HTML (Hypertext Markup Language) ou Linguagem de Marcação

de Hipertexto são definidos símbolos de marcação que informam ao navegador World Wide Web como exibir textos, imagens e uma variedade de recursos na página que é renderizada e apresentada ao usuário (ROUSE, 2005).

- **JSP e Servlets** - *JavaServer Pages* (JSP) e Servlets são tecnologias complementares para a produção páginas Web dinâmicas via Java. Servlets são a base para o *server side* Java. JSP complementa Servlets simplificando o desenvolvimento. Uma página JSP básica consiste em texto simples e de marcação e pode, opcionalmente, ter scripts incorporados e outras funcionalidades para a criação dinâmica de conteúdo (JSP, 2007).
- **CSS** - Cascading Style Sheets (CSS) é uma “folha de estilo” composta por “camadas” e utilizada para definir a apresentação (aparência) em páginas da internet que adotam para o seu desenvolvimento linguagens de marcação (como XML, HTML e XHTML). O CSS define como serão exibidos os elementos contidos no código de uma página da internet e sua maior vantagem é efetuar a separação entre o formato e o conteúdo de um documento (PEREIRA, 2009a).
- **JavaScript** - A linguagem JavaScript (GOODMAN, 2010). É uma linguagem popular e hoje é empregada em diversas plataformas, tal como cliente, servidor, smartphones e desktops (STEFANOV, 2013).
- **Bootstrap** - Originalmente criado por um designer e um desenvolvedor do Twitter 2015, o *Bootstrap* é um framework front-end popular e é um projeto de código aberto. O *Bootstrap* em sua terceira versão se preocupa primeiramente com acesso por dispositivos móveis, focando-se no design responsivo (BOOTSTRAP, 2014).
- **Java** - Originalmente desenvolvida por uma equipe de desenvolvedores liderada por James Gosling na *Sun Microsystems* (atualmente de propriedade da Oracle) e lançada em 1995, o Java é uma linguagem de programação orientada a objetos que atualmente faz parte do núcleo da Plataforma Java (PEREIRA, 2009b).
- **JPA** - JPA é um framework leve, baseado em POJOS (Plain Old Java Objects) para persistir objetos Java. A Java Persistence API, diferente do que muitos imaginam, não é apenas um framework para Mapeamento Objeto-Relacional (ORM - Object-Relational Mapping), ela também oferece diversas

funcionalidades essenciais em qualquer aplicação corporativa (MEDEIROS, 2009).

- **PostgreSQL** - PostgreSQL é um sistema poderoso banco de dados objeto-relacional open source. Ele tem mais de 15 anos de desenvolvimento ativo e uma arquitetura comprovada que ela ganhou uma forte reputação de confiabilidade, integridade de dados e correção. Ele é executado em todos os principais sistemas operacionais, incluindo Linux, UNIX (AIX, BSD, HP-UX, SGI IRIX, Mac OS X, Solaris, Tru64), e Windows. É totalmente compatível com ACID, tem suporte completo para chaves estrangeiras, junções views, triggers e procedimentos armazenados (em várias línguas). Ele inclui mais 2008 tipos de dados SQL, incluindo INTEGER, NUMERIC, BOOLEAN, CHAR, VARCHAR, DATE, INTERVAL e TIMESTAMP. Ele também suporta o armazenamento de grandes objetos binários, incluindo imagens, sons ou vídeo. Ele tem interfaces de programação nativas para C / C ++, Java, .Net, Perl, Python, Ruby, Tcl, ODBC, entre outros, e documentação excepcional (POSTGRESQL, 2015).
- **Spring data** - O Spring data É um módulo de Spring *Framework*. Sua missão é prover um modelo de programação baseado em Spring, consistente e familiar, para ser usado no acesso a dados em bancos relacionais e não relacionais (PIVOTAL, 2015a).
- **Redis** - O Redis é um armazenador de estrutura de dados em memória, usado como banco de dados, cache e *message broker*. Suporta estrutura de dados como *strings*, *hashes*, *listas*, *conjuntos* e várias outras (REDIS, 2016).

4.3.7 Implantação

Como o sistema será web e utilizará Java como linguagem *server side*, o sistema deverá funcionar em um servidor para sistemas Java. O servidor escolhida para realizar o teste de implantação foi o Heroku ¹.

Quanto ao processo de instalação, basta que o arquivo .war gerado no *deploy* da aplicação seja executado com o comando `java -jar "arquivo".war`.

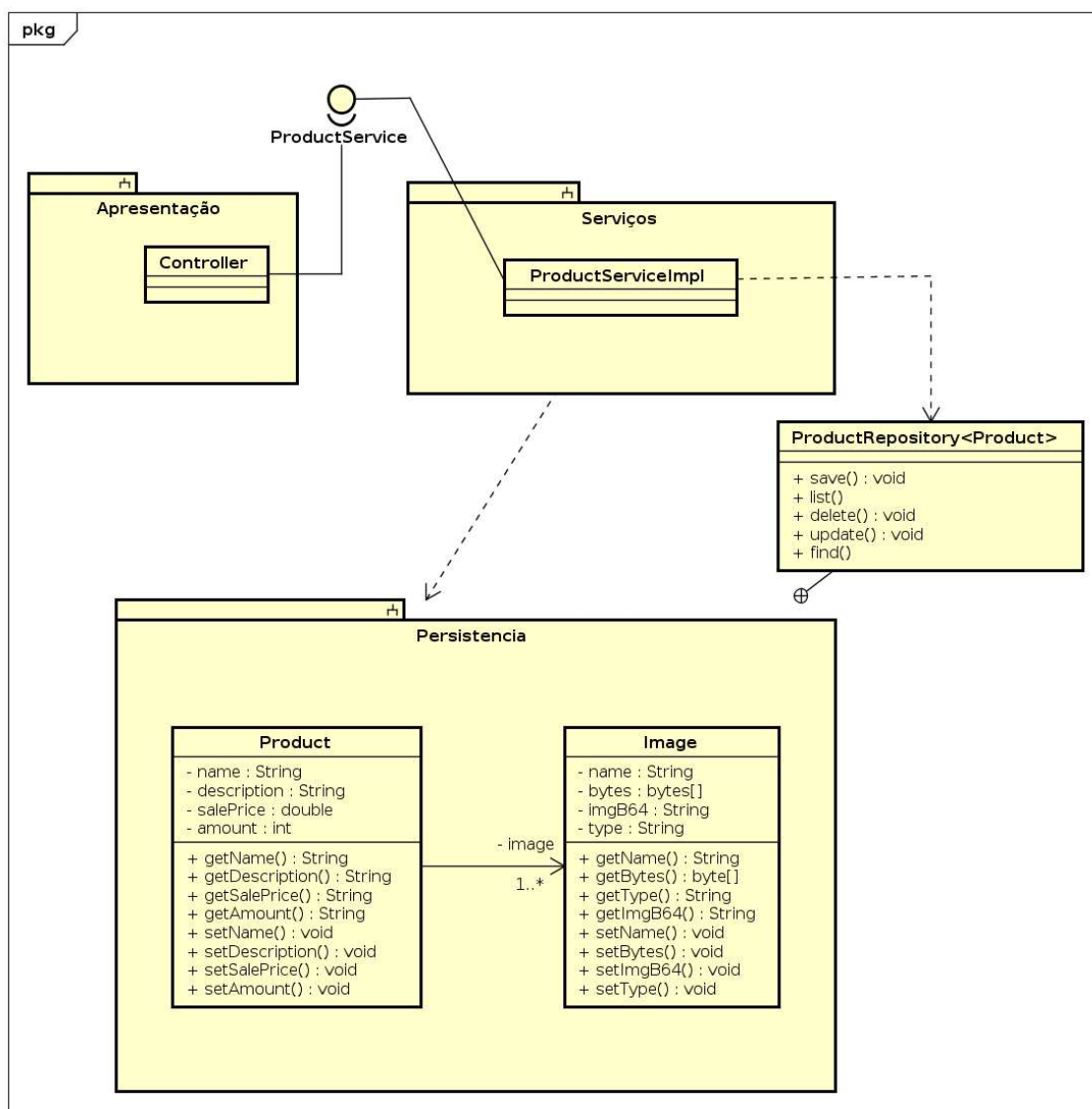
¹<https://www.heroku.com/>

O sistema se encontra *online* e pode ser acessado a partir da URL (<https://sudoloja.herokuapp.com>).

4.4 Diagrama de Classes

Na Figura 9, será apresentado um diagrama de classes que ilustra uma visão mais baixo nível de como as classes necessárias para realizar as operações da US03, deverão se comunicar.

Figura 9: Diagrama de classes US03



powered by Astah

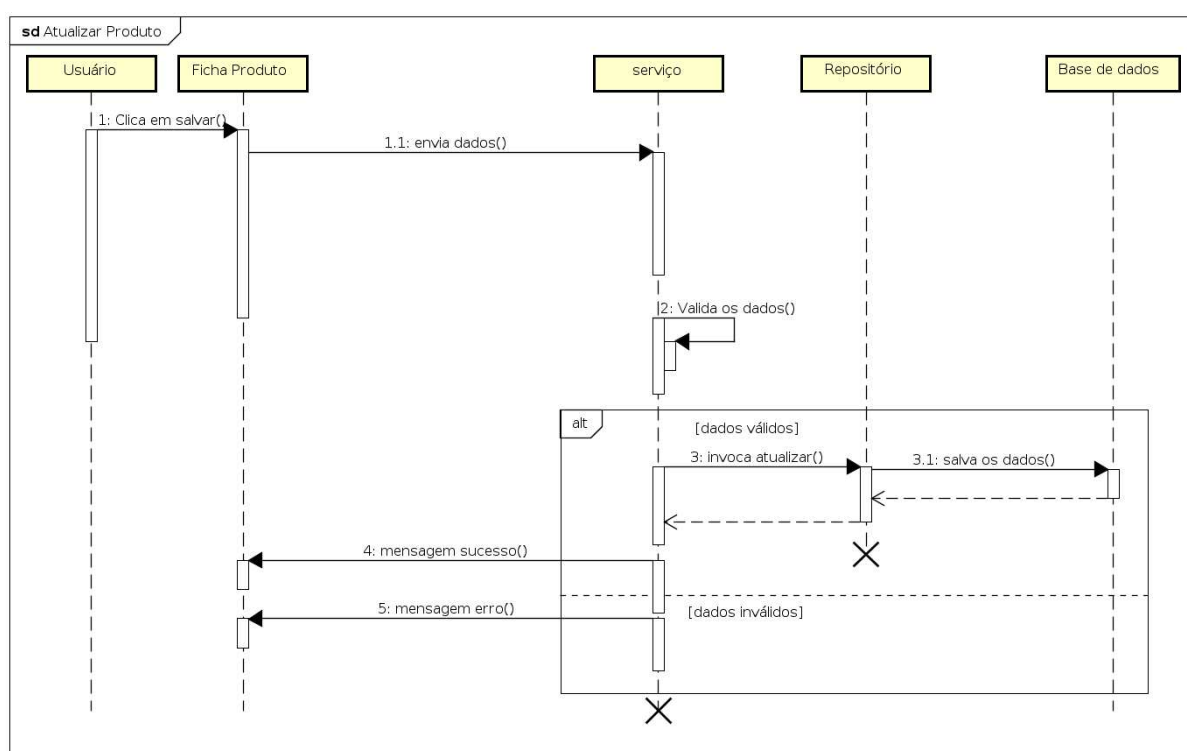
Um *Controller* ainda na camada de apresentação invocará um serviço chamado *ProductServiceImpl*, que fará uso da Classe *ProductRepository*, que por

sua vez retornará para o serviço, o resultado que será repassado para a apresentação.

4.5 Implementação

Na Figura 10, será apresentado um diagrama de sequência que ilustra como se dá a atualização de um produto do estoque, tarefa esta que faz parte da US03.

Figura 10: Diagrama de sequência - atualizar informações do Produto



powered by Astah

Suponhamos que o usuário já esteja devidamente cadastrado e logado no sistema e agora está na página de edição de um produto do estoque, como representa a figura 11. O usuário altera as informações que julgar necessário e clica no botão salvar, isto gera uma requisição para o controlador, que por sua vez irá invocar o serviço de atualizar produto. Antes de salvar as alterações o serviço irá validar os dados, sendo válidos o repositório de produtos é invocado e as alterações são salvas na base de dados. Por fim, em caso de sucesso, será exibida uma mensagem positiva para o usuário confirmando a alteração, caso contrário será apresentada uma mensagem de erro.

Figura 11: Ficha de edição de Produto

Destacar esse produto na página inicial? ☐ Cobrar frete? ☐ peso

Editar Camisa Flamengo

Variações

Cor Tamanho

Categorias

Camisas

Nome

Camisa Flamengo

Especificações

100% Alogodão

Preço de venda

30.00

Quantidade

5

Salvar Alterações

Variações para esse produto

Descrição

Exemplo: Cor

Opções

Exemplo: Azul, Vermelho, Prata, Amarelo

Adicionar variação

Fotos

Adicione fotos para este produto

O estoque de uma loja do sistema é ilustrado na Figura 12. Nessa página o usuário pode adicionar um novo produto, clicando no botão **Cadastrara novo produto**, onde será exibido um formulário para inserção dos dados do produto; Poderá também editar um produto, clicando sobre ele na tabela, sua ficha será exibida, onde o usuário poderá ver com detalhes todas as informações inerentes ao produto, podendo atualizá-las.

Figura 12: Página do estoque de uma loja do sistema

SudoLoja João Marcos

João Marcos Online

Configurar

Início

Conta

Configurar nova Loja

Suas Lojas

Demo

Configurações

Produtos

Pedidos

Usuários

Comentários

Personalização

Demo Configurações

Produtos

10 resultados por página

Pesquisar

Nome	Quantidade	Preço Normal	Código
Camisa Flamengo	5	30.00	183
Ipad	1000	3000.00	188

Mostrando de 1 até 2 de 2 registros

Anterior 1 Próximo

Copyright © 2015 Sudo Loja. Todos os direitos reservados

Version 0.9 Beta

5 Considerações Finais

Para Suh, Ellis e Thiemecke (2002), um CMS busca automatizar o processo de criação, publicação e atualização de conteúdo em um web site. A Sudo Loja como sendo um CMS, automatiza uma série de tarefas quando se trata de criar uma loja virtual. O sistema torna a tarefa de criar, manter e atualizar o conteúdo de um *e-commerce* mais fácil, e não apenas para usuários com algum conhecimento técnico. Aqueles cujo conhecimento seja considerado mínimo, poderá facilmente criar e introduzir conteúdo no seu site de vendas.

Tudo isso feito a partir de um editor front-end, por onde o usuário conseguirá abastecer sua loja com dados; um sistema *back-end* para armazenar e processar o conteúdo; e um mecanismo que obtém o conteúdo e o renderiza, com base num modelo predefinido de site. Este modelo pode ser personalizado pelo próprio usuário, da mesma forma que acontece com o conteúdo.

Algumas tecnologias tiveram que ser abordadas com mais profundidade, para que o sistema pudesse ser finalizado e concluído como planejado. O Spring *Framework*, por exemplo, tornou o desenvolvimento muito mais eficiente e rápido. O projeto foi iniciado sem o uso do Spring. Após sua integração notou-se maior velocidade e qualidade na confecção do código fonte.

5.1 Trabalhos futuros

O sistema desenvolvido trouxe um resultado satisfatório, porém alguns pontos ainda podem ser melhorados, e novas funcionalidades poderiam ser acopladas.

- O sistema utilizou apenas uma estratégia para a realização de pagamentos. O uso do PagSeguro. Outras plataformas poderiam ser integradas ao sistema.
- A personalização foi feita a partir de um template pré definido. O sistema poderia disponibilizar a escolha dentre diversos templates.
- O sistema poderia possibilitar o uso de cupons de desconto em compras.

- O sistema poderia disponibilizar diferentes tipos de relatórios, como por exemplo, de vendas, ou produtos menos vendidos, entre outros.

Referências Bibliográficas

- ALBERTIN, A. L. Comércio eletrônico: um estudo no setor bancário. *Revista de Administração Contemporânea*, SciELO Brasil, v. 3, n. 1, p. 47–70, 1999.
- ANGLES, R.; GUTIERREZ, C. Survey of graph database models. *ACM Computing Surveys (CSUR)*, ACM, v. 40, n. 1, p. 1, 2008.
- ASFOURA, E. S.; JAMOUS, N.; SALEM, W. The economic classification of e-learning business models. *IEEE Technology and Engineering Education (ITEE)*, v. 4, n. 1, p. 8–12, 2009.
- BÁRCIA, L. M. R. C. *A utilização da plataforma Joomla na escola*. Tese (Doutorado), 2008.
- BLOCH, M.; PIGNEUR, Y.; SEGEV, A. On the road of electronic commerce—a business value framework, gaining competitive advantage and some research issues. *Lausanne: Institut D'Informatique et Organization, Ecole des Hautes Etudes Commerciales, Université de Lausanne*, 1996.
- BOOCH, G.; RUMBAUGH, J.; JACOBSON, I. *UML: guia do usuário*. [S.l.]: Elsevier Brasil, 2006.
- BOOTSTRAP. *About Bootstrap*. 2014. Acessada Out. 2015. Disponível em: <http://getbootstrap.com/about/>.
- CARLSON, J. L. *Redis in Action*. Greenwich, CT, USA: Manning Publications Co., 2013. ISBN 1617290858, 9781617290855.
- DIANA, M. D.; GEROSA, M. A. Nosql na web 2.0: Um estudo comparativo de bancos não-relacionais para armazenamento de dados na web 2.0. 2010.
- DRUPAL. *Drupal*. 2015. Acessada Out. 2015. Disponível em: <https://www.drupal.org/>.
- E-BIT. *Relatório Webshoppers*. [S.l.]: Buscapé Company, 2015.
- FELIPINI, D. *Empreendedorismo na internet*. [S.l.]: Lebooks, 2012.
- FILHO, D. F. da S. et al. Banco de dados relacional para cadastro, avaliação e manejo da arborização em vias públicas¹. SciELO Brasil, 2002.
- FOLDDOC. *Application Program Interface*. 2011. Acessada Abr. 2016. Disponível em: <http://foldoc.org/Application+Program+Interface>.
- GARCIA, F. P. et al. easyprocess: Um processo de desenvolvimento para uso no ambiente acadêmico. In: *XII WEI-Workshop de Educação em Computação, XXIV Congresso da Sociedade Brasileira de Computação*. [S.l.: s.n.], 2004.
- GOODMAN, D. et al. *Javascript Bible*. Indianapolis: Wiley Publishing, 2010.

GORDON, S. R. *Sistemas de Informação: uma abordagem gerencial*. Rio de Janeiro: LTC, 2011.

HEUSER, C. A. *Projeto de banco de dados: Volume 4 da Série Livros didáticos informática UFRGS*. [S.l.]: Bookman Editora, 2009.

INTEGRADA, L. *Loja Integrada*. 2015. Acessada Out. 2015. Disponível em: <http://lojaintegrada.com.br>.

JALOTE, P. *An integrated approach to software engineering*. [S.l.]: Springer Science & Business Media, 2012.

JSP, J. S. P. *Java Server Pages*. 2007.

KALAKOTA, R.; WHINSTON, A. B. *Frontiers of electronic commerce*. [S.l.]: Addison Wesley Longman Publishing Co., Inc., 1996.

KAUFELD, J. Access 95 para windows para leigos: Um manual para novos usuários. LUDEMIR, J. *São Paulo: Berkeley Brasil*, 1996.

MAGENTO. *Magento E-Commerce*. 2015. Acessada Out. 2015. Disponível em: <http://magento.com>.

MEDEIROS, H. *Introdução à JPA - Java Persistence API*. 2009. Acessada Out. 2015. Disponível em: <http://www.devmedia.com.br/introducao-a-jpa-java-persistence-api/28173>.

MERCER, D. *Building powerful and robust websites with Drupal 6*. [S.l.]: Packt Publishing Ltd, 2008.

NETO, F. P. de F. Nosql - banco de dados chave-valor. Apresentação em Slides. 2015.

OPENTEXT. *OpenText Web Content Management Systems*. 2015. Acessada Out. 2015. Disponível em: <http://www.vignette.com>.

PEREIRA, A. P. *O que é CSS?* 2009. Acessada Out. 2015. Disponível em: <http://www.tecmundo.com.br/programacao/2705-o-que-e-css-.htm>.

PEREIRA, A. P. *O que é Java?* 2009. Acessada Out. 2015. Disponível em: <http://www.tecmundo.com.br/programacao/2710-o-que-e-java-.htm>.

PEREIRA, J. C.; BAX, M. P. Introdução à gestão de conteúdos. *Revista Gestão & Tecnologia*, v. 1, n. 1, 2010.

PIVOTAL. *Spring Data*. 2015. Acessada Abr. 2016. Disponível em: <http://projects.spring.io/spring-data/>.

PIVOTAL. *Spring Framework*. 2015. Acessada Abr. 2016. Disponível em: <https://projects.spring.io/spring-framework/>.

POSTGRESQL. *About PostgreSQL*. 2015. Acessada Out. 2015. Disponível em: <http://www.postgresql.org/about/>.

PRESSMAN, R. S. *Engenharia de software*. [S.l.]: AMGH Editora, 2009.

REDIS. *Redis*. 2016. Acessada Abr. 2016. Disponível em: <http://redis.io/>.

ROUSE, M. *HTML (Hypertext Markup Language)*. 2005. Acessada Out. 2015. Disponível em: <http://searchsoa.techtarget.com/definition/HTML>.

SOMMERVILLE, I. Engenharia de software, 8ª edição, tradução: Selma shin shimizu mel-nikoff, reginaldo arakaki, edilson de andrade barbosa. *São Paulo: Pearson Addison-Wesley*, v. 22, p. 103, 2007.

SOMMERVILLE, I. et al. *Engenharia de software*. São Paulo: ADDISON WESLEY BRA, 2008. 552 p. ISBN 9788588639287.

SOUZA, C. R. Um framework para editores de diagramas cooperativos baseados em anotações. *Campinas: Unicamp*, 1998.

STEFANOV, S. *JavaScript for PHP Developers*. [S.l.]: O'reilly Media, 2013.

STONEBRAKER, M. et al. C-store: a column-oriented dbms. In: VLDB ENDOWMENT. *Proceedings of the 31st international conference on Very large data bases*. [S.l.], 2005. p. 553–564.

STRAUCH, C.; SITES, U.-L. S.; KRIHA, W. Nosql databases. *Lecture Notes, Stuttgart Media University*, 2011.

SUH, P.; ELLIS, J.; THIEMECKE, D. *Content management systems*. [S.l.]: Peer Information, 2002.

TRAY. *Tray Commerce*. 2015. Acessada Out. 2015. Disponível em: <http://tray.com.br>.

TURBAN, E. *Administração da Tecnologia da Informação*. [S.l.]: Campus, 2008.

TWITTER. *About Twitter*. 2015. Acessada Out. 2015. Disponível em: <https://twitter.com>.

UOL, U. O. *PagSeguro: Sua solução completa para pagamentos*. 2016. Acessada Abr. 2016. Disponível em: https://pagseguro.uol.com.br/sobre_o_pagseguro.jhtml.

UOLHOST. *UOLHost Loja Virtual*. 2012. Acessada Out. 2015. Disponível em: <http://uolhost.com.br>.

VIEIRA, M. R. et al. Bancos de dados nosql: conceitos, ferramentas, linguagens e estudos de casos no contexto de big data. *Simpósio Brasileiro de Bancos de Dados*, 2012.

WIX. *Loja Virtual Wix*. 2012. Acessada Out. 2015. Disponível em: <http://pt.wix.com/ecommerce/website>.

WORDPRESS. *WordPress.org*. 2015. Acessada Out. 2015. Disponível em: <https://wordpress.org/>.

ZADROZNY, B. Documento de visão. Trabalho Prático - UFF. Instituto de Computação – Engenharia de Software II.

A Documento de Visão

A.1 Introdução

Este documento apresenta uma solução de software para ser apresentado como TCC, solicitado pelos professores. Como cliente Francisco Paulo de Freitas Neto (Orientador). Apresenta também os problemas a serem solucionados, as necessidades dos principais envolvidos, o alcance do projeto e as funcionalidades esperadas do sistema.

A.1.1 Objetivos

Como objetivo principal, esse projeto visa facilitar o ingresso de um comerciante ao mercado virtual, fornecendo uma ferramenta capaz de criar uma loja virtual onde um usuário, sem ajuda especializada, poderá gerenciar e customizar sua loja de maneira simples e intuitiva.

A.1.2 Abrangência do Projeto

Esse projeto é voltado ao âmbito comercial, contemplando todos aqueles que desejam criar uma loja virtual e disponibilizar produtos à venda. O produto final será uma plataforma onde qualquer usuário cadastrado poderá criar, gerenciar e personalizar uma loja virtual.

A.1.3 Definições, Acrônimos e Abreviações

CMS – Content Management System (Sistema de Gerenciamento de Conteúdo)

A.1.4 Organização do Documento

As próximas seções trataram sobre a descrição do problema, as partes envolvidas e tipos de usuários que utilizaram o sistema, a solução proposta para os problemas elencados, logo após, as funcionalidades oferecidas por esta solução, seguidas pelas restrições do sistema. Também poderão ser vistos alguns protótipos na Seção A.7.

A.2 Descrição do Problema

- Dificuldades no gerenciamento de uma loja virtual - Aqueles que desejam entrar no mercado virtual geralmente recorrem a empresas de desenvolvimento de software para criar sua loja virtual, e geralmente, qualquer tipo de modificação a ser feita, seja no conteúdo ou no *layout* e estilo do site, necessita da intervenção direta da equipe de desenvolvimento.
- Afeta - O lojista.
- O impacto deste problema é - Um grande impacto que isso causa para o lojista é a demora, tanto no desenvolvimento quanto na modificação de alguma parte da loja online, por exemplo, digamos que por algum motivo o lojista queira modificar o estilo do sua loja online. A equipe então, levará um certo tempo (dependendo da complexidade de modificação) até entender completamente o que o cliente deseja para então realizar a modificação e deixar a loja do jeito que o cliente pediu, e é claro, na maioria dos casos com um custo considerável. Então temos dois problemas principais: demora e custo.
- Uma solução ideal permitiria - Fornecer um sistema que possibilitasse a criação, personalização e gerenciamento de uma loja online sem intervenção de pessoal especializado, com um baixo custo.

A.3 Partes Envolvidas

Os principais envolvidos serão descritos nas seções A.3.1 e A.3.2.

A.3.1 Resumo dos Envolvidos

Os envolvidos que se interessam em empenhar-se nesse projeto, são os desenvolvedores, os comerciantes, e os consumidores.

A.3.2 Resumo dos Usuários

Uma lista resumida de todos os usuários identificados pode ser vista no Quadro 6.

Quadro 6: Resumo dos usuários

Nome	Descrição	Responsabilidades	Envolvido
Lojista	Administrador da loja virtual	Criação da loja, Personalização da loja, Inclusão de dados no sistema, Gerenciamento de pedidos	Dono do negócio ou alguém que represente seus interesses
Cliente	Consumidor dos produtos oferecidos na loja	Efetuar cadastro para realização de uma compra	Qualquer consumidor em potencial que deseje realizar uma compra (de um ou mais) produtos oferecidos pela loja

A.4 Descrição da Solução Proposta

A solução proposta é um sistema computacional web para criação de lojas virtuais (um CMS voltado para criação de lojas, onde um usuário poderá criar, personalizar as suas lojas virtuais e assim, disponibilizar os seus produtos para venda) e uma área para perguntas ao vendedor sobre um produto desejado. tudo isso de forma simples

sem conhecimento especializado, sem burocracias.

A.5 Funcionalidades

No Quadro 7 podemos ver uma breve descrição das funcionalidades do produto, com um nível de detalhamento genérico para uma melhor compreensão.

Quadro 7: Resumo das Funcionalidades

Funcionalidade	Descrição
Gerenciamento de produtos	Inclusão, remoção e alteração de produtos, cadastro de múltiplas imagens para cada produto
Gerenciamento de pedidos	Controle de todos os pedidos realizados
Personalização da loja	A loja poderá ser personalizada totalmente (<i>layout</i> , cores, fontes, edição de CSS e HTML)
Gerenciamento de clientes	Cadastro, alteração de dados cadastrais, remoção da conta.
Calculo de frete automático quando houver	Caso haja frete sobre um produto da loja, em tempo real o cliente poderá ver o valor do frete
Controle de estoque	O estoque poderá ser consultado e gerenciado, contendo opções como desabilitar/habilitar produtos, visualizar quantidade de cada produto, filtrar produtos por nome
Acompanhamento de pedidos para os clientes	Os clientes poderão visualizar o status do pedido a qualquer momento,

A.6 Restrições do Projeto

O sistema será uma aplicação WEB com design totalmente responsivo, o que significa que poderá ser utilizado tanto pelo navegador de um computador desktop, quanto pelo navegador do celular. Não possuirá restrições quanto ao sistema operacional utilizado pelo consumidor.

A.7 Protótipos

Figura 13: Visualização de produtos cadastrados

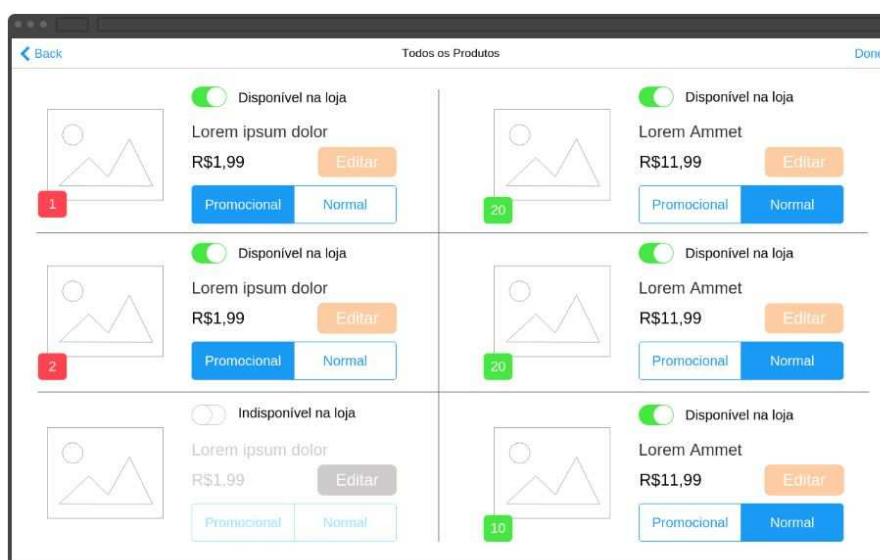


Figura 14: Cadastro de produtos

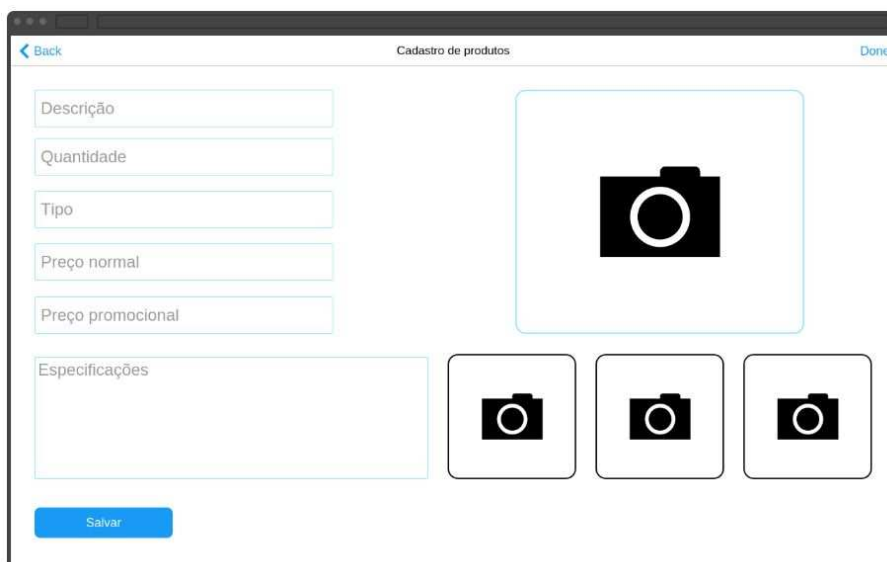
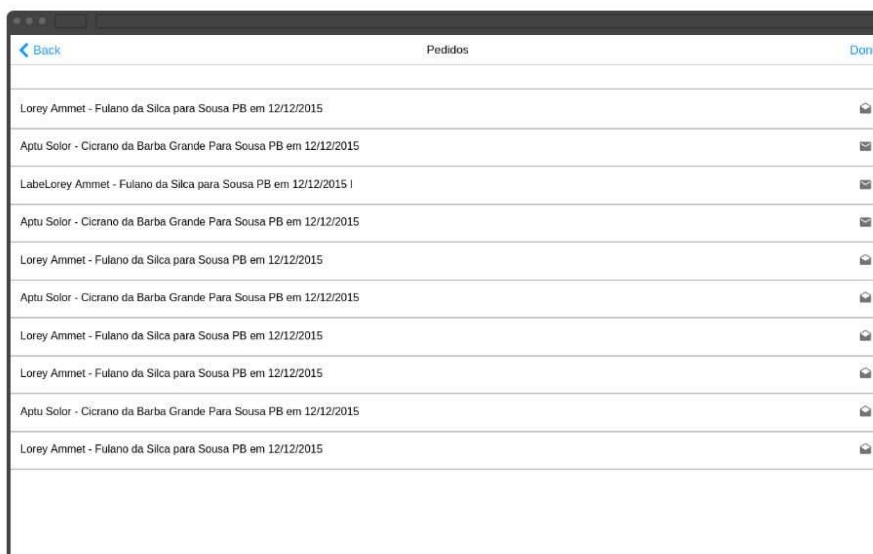


Figura 15: Visualização de pedidos



	Pedidos	Done
Lorey Ammet - Fulano da Silca para Sousa PB em 12/12/2015		
Aptu Solor - Cicrano da Barba Grande Para Sousa PB em 12/12/2015		
Labelorey Ammet - Fulano da Silca para Sousa PB em 12/12/2015		
Aptu Solor - Cicrano da Barba Grande Para Sousa PB em 12/12/2015		
Lorey Ammet - Fulano da Silca para Sousa PB em 12/12/2015		
Aptu Solor - Cicrano da Barba Grande Para Sousa PB em 12/12/2015		
Lorey Ammet - Fulano da Silca para Sousa PB em 12/12/2015		
Lorey Ammet - Fulano da Silca para Sousa PB em 12/12/2015		
Aptu Solor - Cicrano da Barba Grande Para Sousa PB em 12/12/2015		
Lorey Ammet - Fulano da Silca para Sousa PB em 12/12/2015		

Figura 16: Painel de Personalização

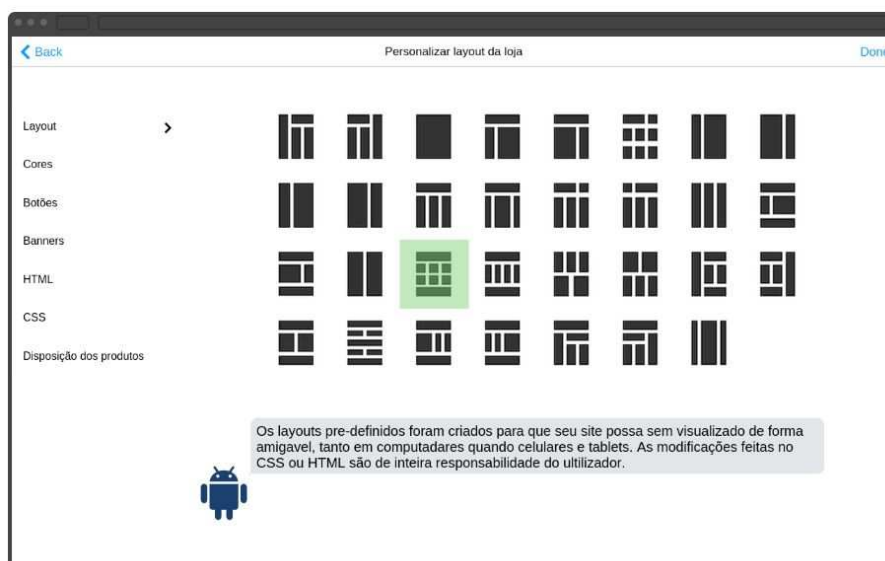
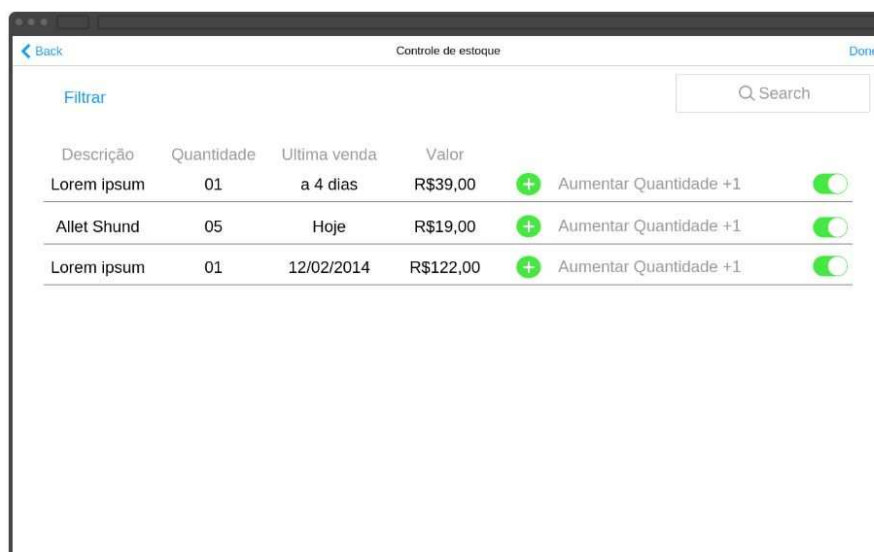


Figura 17: Controle de estoque



The image shows a mobile application interface for inventory control. At the top, there is a header bar with a back arrow and the text 'Back' on the left, 'Controle de estoque' in the center, and 'Done' on the right. Below the header, there is a search bar with a magnifying glass icon and the text 'Search'. To the left of the search bar is a button labeled 'Filtrar'. Below the search bar is a table with three columns: 'Descrição', 'Quantidade', and 'Última venda'. The table has three rows of data. Each row has a green plus icon in a circle to the right of the 'Última venda' column, followed by the text 'Aumentar Quantidade +1', and a green toggle switch to the right of that text. The table is styled with a light gray background and white text. The rows are separated by thin white lines.

Descrição	Quantidade	Última venda	Valor			
Lorem ipsum	01	a 4 dias	R\$39,00	+	Aumentar Quantidade +1	<input checked="" type="checkbox"/>
Allet Shund	05	Hoje	R\$19,00	+	Aumentar Quantidade +1	<input checked="" type="checkbox"/>
Lorem ipsum	01	12/02/2014	R\$122,00	+	Aumentar Quantidade +1	<input checked="" type="checkbox"/>

B User Stories e Testes de Aceitação

Quadro 8: Alocação de Tarefas - US01

User Story 01					
Tarefa	Descrição	Responsável	Estimativa de tempo (horas)	Tempo real (horas)	Status
T1	Implementar Script para salvar um Usuário Lojista	João Marcos	2	1	Finalizado
T2	Implementar Script para validar informações do usuário Lojista	João Marcos	2	1	Finalizado
T3	Criar Formulário para cadastro usuários lojistas	João Marcos	3	2	Finalizado
T5	Criar página editar as informações do usuário Lojista	João Marcos	4	3	Finalizado

Quadro 9: Teste de aceitação - US01

User Story 01		
Testes de aceitação		Status
TA1	Dado que estou na página de registro de novo usuário, quando eu preencher o formulário com os dados obrigatórios de forma correta , o usuário deve ser salvo.	Finalizado

TA2	Dado que estou na página de registro de novo usuário, quando eu preencher o formulário com os dados obrigatórios de forma incorreta , o usuário não deve ser salvo e uma mensagem de erro deve ser exibida.	Finalizado
TA3	Dado que estou acessando a ficha de edição de informações sobre o Lojista, quando eu alterar alguma informação do formulário com os dados válidos, o usuário deve ser atualizado e uma mensagem de sucesso deve ser exibida.	Finalizado
TA4	Dado que estou acessando a ficha de edição de informações sobre o Lojista, quando eu alterar alguma informação do formulário com os dados inválidos, tais informações não serão salvas e uma mensagem de erro deve ser exibida.	Finalizado

Quadro 10: Alocação de Tarefas - US01

User Story 02					
Tarefa	Descrição	Responsável	Estimativa de tempo (horas)	Tempo real (horas)	Status
T1	Implementar Script para salvar uma Loja	João Marcos	2	1	Finalizado
T2	Implementar Script para validar informações do de uma Loja	João Marcos	2	1	Finalizado
T3	Criar Formulário para cadastro de uma Loja	João Marcos	3	2	Finalizado
T5	Criar página editar as informações Sobre uma Loja	João Marcos	4	3	Finalizado

Quadro 11: Teste de aceitação - US02

User Story 02		
Testes de aceitação		Status
TA1	Dado que estou na página de criação de uma Loja, quando eu preencher o formulário com os dados obrigatórios de forma correta , a loja ser salva.	Finalizado
TA2	Dado que estou na página criação de uma Loja, quando eu preencher o formulário com os dados obrigatórios de forma incorreta , a Loja não deve ser salva e uma mensagem de erro deve ser exibida.	Finalizado
TA3	Dado que estou acessando a página de edição de informações sobre uma Loja, quando eu alterar alguma informação do formulário com os dados válidos, a Loja deve ser atualizada e uma mensagem de sucesso deve ser exibida.	Finalizado
TA4	Dado que estou acessando a página de edição de informações sobre uma Loja, quando eu alterar alguma informação do formulário com os dados inválidos, tais informações não serão salvas e uma mensagem de erro deve ser exibida.	Finalizado
TA5	Permitir que eu possa cadastrar mais de uma Loja no sistema.	Finalizado

Quadro 12: Alocação de Tarefas - US03

User Story 03					
Tarefa	Descrição	Responsável	Estimativa de tempo (horas)	Tempo real (horas)	Status
T1	Implementar Script para salvar um Produto	João Marcos	2	1	Finalizado
T2	Implementar Script para validar informações do produto	João Marcos	2	1	Finalizado

T3	Criar Formulário para cadastro de produto	João Marcos	3	2	Finalizado
T4	Criar formulário para edição de produto	João Marcos	2	1	Finalizado
T5	Criar página para o estoque, exibindo as informações básicas de cada produto com sua respectiva quantidade e link para ficha do produto	João Marcos	4	3	Finalizado
T6	Criar botão para exclusão de um produto na página de estoque	João Marcos	2	1	Finalizado

Quadro 13: Teste de aceitação - US03

User Story 03		
Testes de aceitação		Status
TA1	Dado que estou na página de cadastro de produto, quando eu preencher o formulário com os dados obrigatórios de forma correta , o produto deve ser salvo e informado uma mensagem de sucesso.	Finalizado
TA2	Dado que estou na página de cadastro de produto, quando eu preencher o formulário com os dados obrigatórios de forma incorreta , o produto não deve ser salvo e uma mensagem de erro deve ser exibida.	Finalizado

TA3	Dado que estou acessando a ficha de edição de produto, quando eu alterar alguma informação do formulário com os dados válidos, o produto deve ser atualizado e uma mensagem de sucesso deve ser exibida.	Finalizado
TA4	Dado que estou acessando a ficha de edição de produto, quando eu alterar alguma informação do formulário com os dados inválidos, tais informações não serão salvas e uma mensagem de erro deve ser exibida.	Finalizado
TA5	Dado que estou na página de estoque, quando eu selecionar a opção excluir este produto, um diálogo de confirmação deve ser exibido. Caso confirme o produto deve ser excluído	Finalizado
TA6	Dado que estou na página de estoque, quando eu selecionar a opção excluir este produto, um diálogo de confirmação deve ser exibido. Caso não confirme o produto não deve ser excluído	Finalizado

Quadro 14: Alocação de Tarefas - US04

User Story 04					
Tarefa	Descrição	Responsável	Estimativa de tempo (horas)	Tempo real (horas)	Status
T1	Implementar Script para validar credenciais de um lojista	João Marcos	2	1	Finalizado
T2	Criar página inicial contendo um menu de navegação onde o usuário poderá navegar com facilidade no sistema	João Marcos	4	3	Finalizado

T3	Criar botão para sair do Sistema	João Marcos	2	1	Finalizado
T4	Criar página de seleção de lojas, onde o usuário poderá escolher a loja que deseja gerenciar no momento	João Marcos	5	4	Finalizado
T5	Criar página de gerenciamento específica para lojas, que será exibida ao selecionar uma Loja na T4	João Marcos	4	2	Finalizado
T6	Criar menu lateral na página de gerenciamento da loja, permitindo a navegação dentre as funcionalidades inerentes ao gerenciamento da loja	João Marcos	2	1	Finalizado

Quadro 15: Teste de aceitação - US04

User Story 04		
Testes de aceitação		Status
TA1	Dado que estou na página de login, quando eu preencher o formulário com os dados obrigatórios de forma correta , ser redirecionado para a página inicial do sistema.	Finalizado

TA2	Dado que estou na página de login, quando eu preencher o formulário com os dados obrigatórios de forma incorreta , não posso ser redirecionado para o sistema e uma mensagem de erro deve ser exibida.	Finalizado
TA3	Dado que estou logado, quando clicar no botão sair, devo ser desconectado do sistema, onde terei que fazer login novamente caso queira entrar no sistema novamente.	Finalizado
TA4	Dado que estou na página de seleção de lojas, quando selecionar uma loja, devo ser redirecionado para página inicial de gerenciamento da Loja selecionada.	Finalizado
TA5	Dado que estou logado, quando eu selecionar uma opção do menu lateral, devo ser redirecionado para página especificada	Finalizado

Quadro 16: Alocação de Tarefas - US05

User Story 05					
Tarefa	Descrição	Responsável	Estimativa de tempo (horas)	Tempo real (horas)	Status
T1	Implementar Script para salvar preferência	João Marcos	2	1	Finalizado
T2	Classificar alterações com base na seção da página e as expor em um menu lateral	João Marcos	2	1	Finalizado
T3	Criar formulário para cada tipo de modificação na aparência do site	João Marcos	5	3	Finalizado

T4	Criar alerta ao salvar um alteração	João Marcos	2	1	Finalizado
T4	Criar botão para reverter configurações para padrões	João Marcos	2	1	Finalizado

Quadro 17: Teste de aceitação - US05

User Story 05		
Testes de aceitação		Status
TA1	Dado que estou logado no sistema, quando eu selecionar uma loja, o menu de personalização referente a essa loja é carregado	Finalizado
TA2	Dado que selecionei uma seção no menu de personalização, um conjunto de formulário inerentes a esta é carregado e exibido ao utilizador.	Finalizado
TA3	Dado que alterei o estilo de algo usando o formulário, ao clicar em salvar, as alterações devem ser salvas.	Finalizado
TA4	Dado que retornei a uma seção de menu de personalizações já alterada, a mesma deve carregar as preferencias anteriormente salvas.	Finalizado
TA5	Dado que cliquei no botão “reverter alterações” as modificações devem ser ignoradas e as configurações padrões aplicadas	Finalizado

Quadro 18: Alocação de Tarefas - US06

User Story 06					
Tarefa	Descrição	Responsável	Estimativa de tempo (horas)	Tempo real (horas)	Status

T1	Implementar Script para salvar usuário da loja	João Marcos	2	1	Finalizado
T2	Criar página para cadastro/login de cliente de uma loja	João Marcos	3	3	Finalizado
T3	Criar página para visualização de usuários para cada loja	João Marcos	2	1	Finalizado

Quadro 19: Teste de aceitação - US06

User Story 06		
Testes de aceitação		Status
TA1	Dado que estou na página de cadastro, ao informar e-mail e senha corretos devo ser redirecionado para a página principal da loja	Finalizado
TA2	Quando clicar no botão “sair” minha seção seja encerrada e devo ser redirecionado para a página inicial	Finalizado
TA3	Dado que estou no painel de controle, ao selecionar a opção “usuários” a lista de usuários cadastrado em minha loja deve ser exibida.	Finalizado

Quadro 20: Alocação de Tarefas - US07

User Story 07					
Tarefa	Descrição	Responsável	Estimativa de tempo (horas)	Tempo real (horas)	Status

T1	Implementar Script para carregamento de pedidos de cliente	João Marcos	2	2	Finalizado
T2	Criar página para visualização de pedidos e informações básicas do cliente	João Marcos	5	3	Finalizado
T3	Criar quadros expansíveis para cada pedido	João Marcos	2	1	Finalizado
T4	Sinalizar status de cada pedido	João Marcos	2	1	Finalizado

Quadro 21: Teste de aceitação - US07

User Story 07		
Testes de aceitação		Status
TA1	Dado que estou na página de acompanhamento, devo visualizar minhas informações e informações sobre pedidos feitos por mim.	
TA2	Dado que estou na página de acompanhamento, quando eu clicar em um pedido, o mesmo deve ser expandido e exibir seus detalhes e seu andamento, juntamente com um código de referência	Finalizado
TA3	Dado que expandi um pedido, ao clicar na imagem de um produto, devo ser levado até a sua página para que o mesmo seja exibido em detalhes.	Finalizado

Quadro 22: Alocação de Tarefas - US08

User Story 08

Tarefa	Descrição	Responsável	Estimativa de tempo (horas)	Tempo real (horas)	Status
T1	Implementar Script para salvar pergunta sobre determinado produto	João Marcos	5	2	Finalizado
T2	Criar página para visualização perguntas e respostas, classificadas por data e hora	João Marcos	5	3	Finalizado
T3	Criar formulário para submeter pergunta	João Marcos	2	1	Finalizado
T4	Alertar lojista sobre novas perguntas	João Marcos	2	1	Finalizado

Quadro 23: Teste de aceitação - US08

User Story 08		
Testes de aceitação		Status
TA1	Dado que estou logado como usuário de loja, devo poder fazer quantas perguntas quiser sobre determinado produto em sua página	Finalizado
TA2	Dado que submeti um pergunta, a mesma deve ser salva e alertada ao lojista	Finalizado
TA3	Dado que sou um lojista logado no sistema, ao perceber uma nova pergunta poderei respondê-la e a resposta deve aparecer junto a pergunta feita pelo usuário na página do produto	Finalizado

Quadro 24: Alocação de Tarefas - US09

User Story 09					
Tarefa	Descrição	Responsável	Estimativa de tempo (horas)	Tempo real (horas)	Status
T1	Alertar lojista sobre endereço não cadastrado	João Marcos	1	1	Finalizado
T2	Definir endereço padrão de origem dos produto	João Marcos	2	1	Finalizado
T3	Implementar script para consulta frete SEDEX e PAC	João Marcos	3	3	Finalizado
T4	Adicionar campo peso ao produto, e botão para definir se este é passível ao cálculo de frete ou não	João Marcos	2	1	Finalizado
T5	Implementar script para simular valor de frete a partir de um CEP informado	João Marcos	2	1	Finalizado
T6	Criar formulário para simulação de frete	João Marcos	2	1	Finalizado

T7	Listar e selecionar automaticamente o ultimo endereço informado pelo cliente em uma compra e calcular o frete com base nele	João Marcos	2	1	Finalizado
T8	Página para alternar entre endereços já usados pelo usuário	João Marcos	2	1	Finalizado
T9	Opção de escolha para frete, entre SEDEX e PAC	João Marcos	2	1	Finalizado

Quadro 25: Teste de aceitação - US09

User Story 09		
Testes de aceitação		Status
TA1	Dado que estou na página do carrinho da loja, e este contenha algum item, um formulário de simulação de frete deve ser exibido	Finalizado
TA2	Ao informar um CEP válido para simulação o resultado deve ser apresentado junto ao valor total do carrinho	Finalizado
TA3	Ao informar um CEP inválido para simulação o usuário deve ser alertado sobre isso	Finalizado
TA4	Dado que submeti um CEP para simulação do valor de entrega, e este seja válido, quando os produtos não possuírem frete, a palavra “Grátis” é exibida no local onde seria exibido o valor do frete	Finalizado

TA5	Dado que estou na página de checkout, o frete deve ser automaticamente calculado com base no ultimo endereço informado pelo usuário	Finalizado
TA6	Dado que conclui um pedido, as informações de frete devem ser exibidas na página de acompanhamento de pedidos	Finalizado
TA7	Ao tentar calcular o frete, sinalizar de alguma forma o andamento do processamento	Finalizado

Quadro 26: Alocação de Tarefas - US10

User Story 10					
Tarefa	Descrição	Responsável	Estimativa de tempo (horas)	Tempo real (horas)	Status
T1	Implementar script para salvar imagens	João Marcos	2	1	Finalizado
T2	Implementar script para atualizar imagens	João Marcos	2	1	Finalizado
T3	Criar formulário para inserção e modificação das imagens de um produto	João Marcos	3	1	Finalizado
T4	Criar uma galeria na página do produto, com as imagens cadastradas para o mesmo	João Marcos	4	3	Finalizado

T5	Alternar imagem em destaque na galeria através de clique	João Marcos	2	1	Finalizado
----	--	-------------	---	---	------------

Quadro 27: Teste de aceitação - US10

User Story 10		
Testes de aceitação		Status
TA1	Dado que estou na página do produto, a galeria deve ser exibida com suas imagens	Finalizado
TA2	Ao clicar em uma imagem da galeria a mesma deve ficar em destaque	Finalizado
TA3	O número de imagens para cada produto não pode passar de 5 (cinco)	Finalizado
TA4	O formato e o tamanho das imagens devem ser filtrados antes de sua inserção	Finalizado
TA5	Caso cadastre uma imagem não válida deve ser informado disso através de um alerta	Finalizado

Quadro 28: Alocação de Tarefas - US11

User Story 11					
Tarefa	Descrição	Responsável	Estimativa de tempo (horas)	Tempo real (horas)	Status
T1	Implementar script realizar pagamento (integração PagSeguro)	João Marcos	5	4	Finalizado

T2	Implementar script para salvar dados de pagamento (PagSeguro)	João Marcos	1	1	Finalizado
T3	Criar formulário para inserção de dados de pagamento no painel administrativo	João Marcos	2	1	Finalizado
T4	Criar página para visualização de pedidos para lojistas e clientes	João Marcos	5	4	Finalizado
T5	Criar formulário para alteração de status de pedido no painel administrativo	João Marcos	2	2	Finalizado

Quadro 29: Teste de aceitação - US11

User Story 11		
Testes de aceitação		Status
TA1	Dado que estou na página de pagamento, ao clicar em pagar as opções de pagamento devem ser exibidas	Finalizado
TA2	Dado que sou um lojista, se as informações prestadas por mim sobre o pagamento, não forem válidas, o pagamento não pode ser concluído e o usuário deve ser informado	Finalizado
TA3	Caso conclua o pagamento, o lojista deve ser alertado sobre	Finalizado
TA4	Dado que estou na pagina de gerenciamento de pedidos e sou um lojista, quando o pagamento for aprovado, poderei informar ao cliente que o pedido foi aprovado e está a caminho	Finalizado

TA5	Dado que estou na pagina de gerenciamento de pedidos e sou um lojista, quando um pedido for extraviado, poderei informar ao cliente	Finalizado
-----	---	------------

C Plano de Iteração

Quadro 30: *Release 01*

Release 01		Gerente - João Marcos
Iteração	User Stories	Período
Iteração 1	US01	04/06/2015 - 11/06/2015
Iteração 2	US02	12/06/2015 - 18/06/2015

Quadro 31: *Release 02*

Release 02		Gerente - João Marcos
Iteração	User Stories	Período
Iteração 3	US03	19/06/2015 - 02/07/2015
Iteração 4	US04	03/07/2015 - 16/07/2015

Quadro 32: *Release 03*

Release 02		Gerente - João Marcos
Iteração	User Stories	Período
Iteração 5	US06, US08	02/12/2015 - 14/12/2015
Iteração 6	US09, US10	15/12/2015 - 21/12/2015

Quadro 33: *Release 04*

Release 04		Gerente - João Marcos
Iteração	User Stories	Período
Iteração 7	US07	10/02/2016 - 15/02/2016
Iteração 8	US11	16/02/2016 - 28/02/2016

Quadro 34: *Release 05*

Release 04		Gerente - João Marcos
Iteração	User Stories	Período
Iteração 9	US05	12/03/2016 - 02/04/2016

D Modelo lógico de dados

Este documento apresenta o modelo lógico de dados da Sudo Loja. Para melhor entendimento o diagrama foi decomposto em partes menores, divididos em grau de relacionamento com cada entidade.

Figura 18: Produto e seus relacionamentos

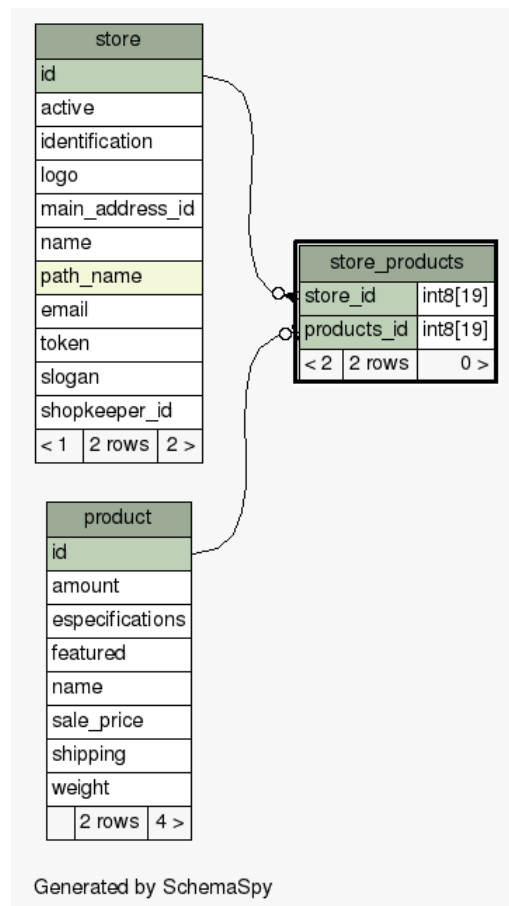


Figura 19: Pedido e seus relacionamentos

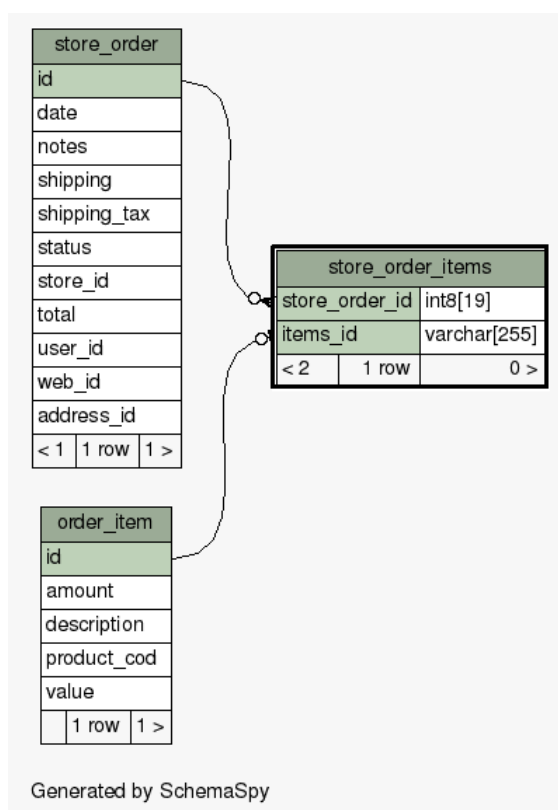


Figura 20: Pedido e seus relacionamentos

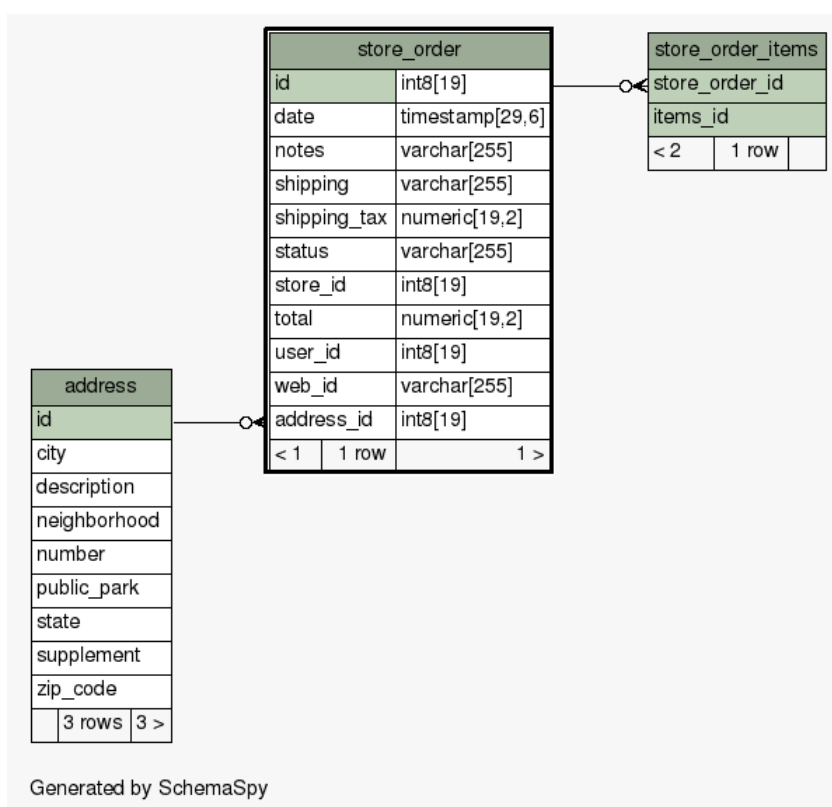


Figura 21: Endereço e seus relacionamentos

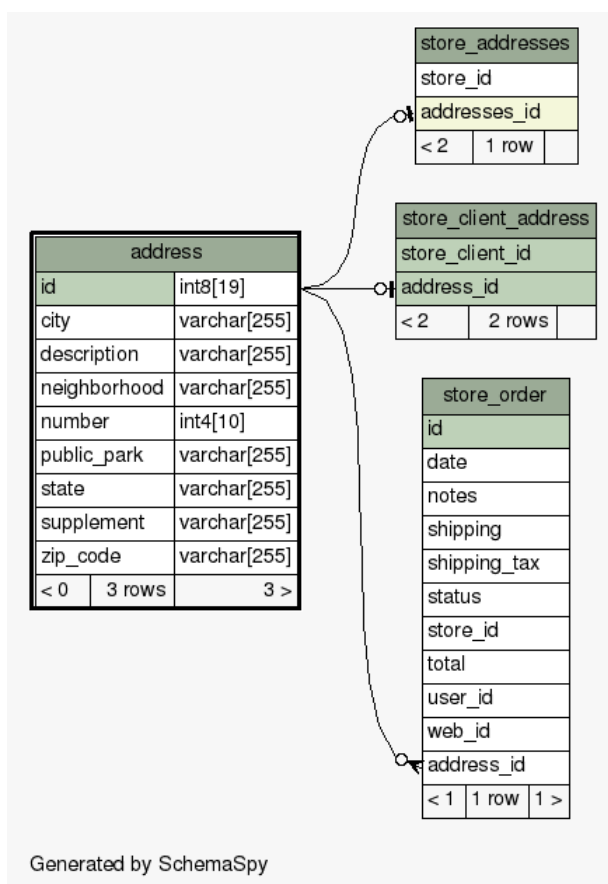


Figura 22: Usuário da loja e seus relacionamentos

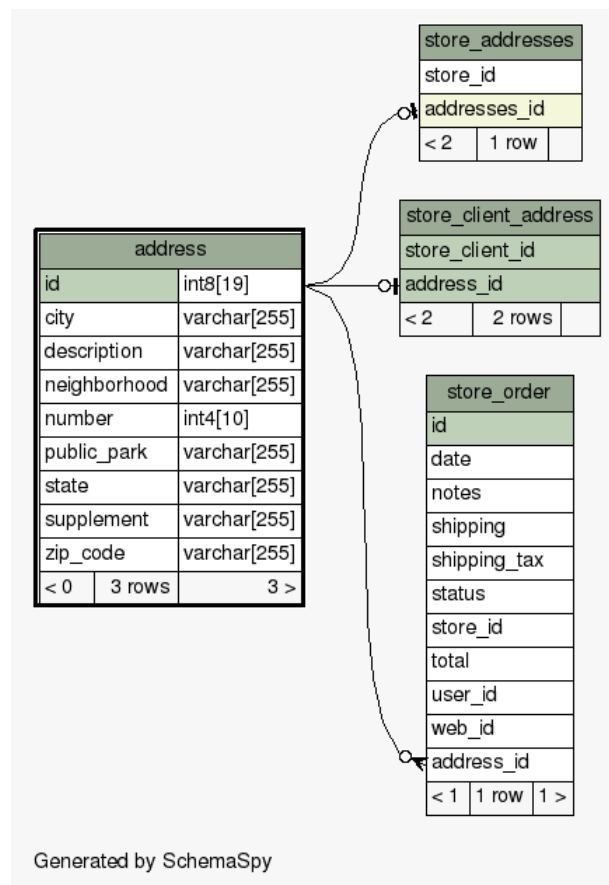


Figura 23: Loja e seus relacionamentos

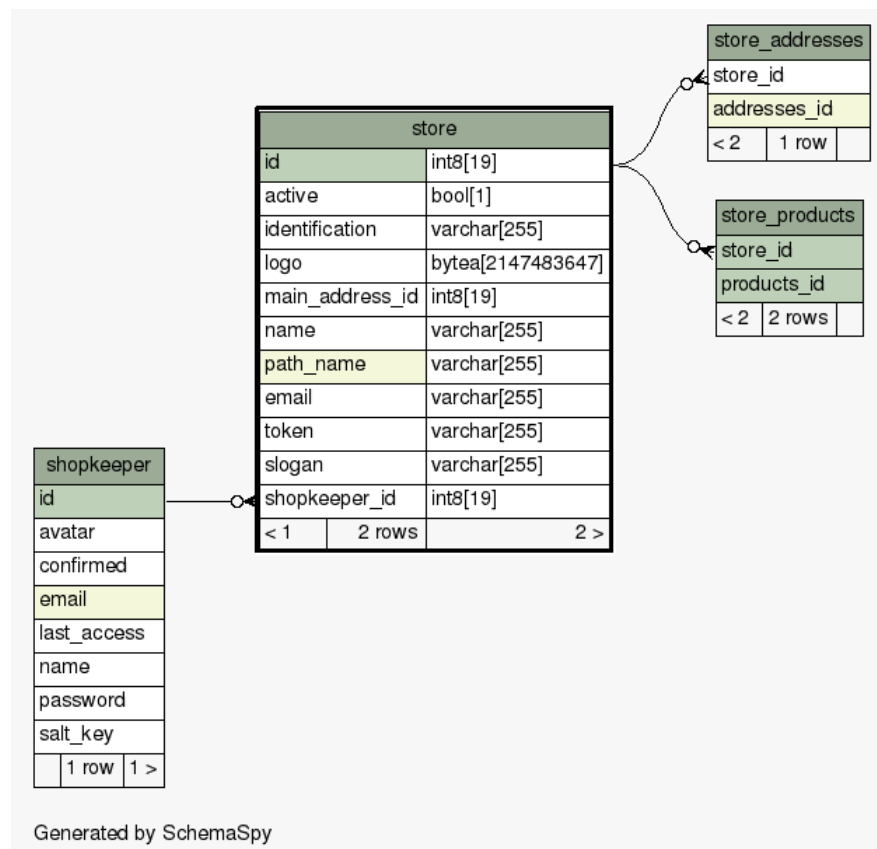


Figura 24: Lojista e seus relacionamentos

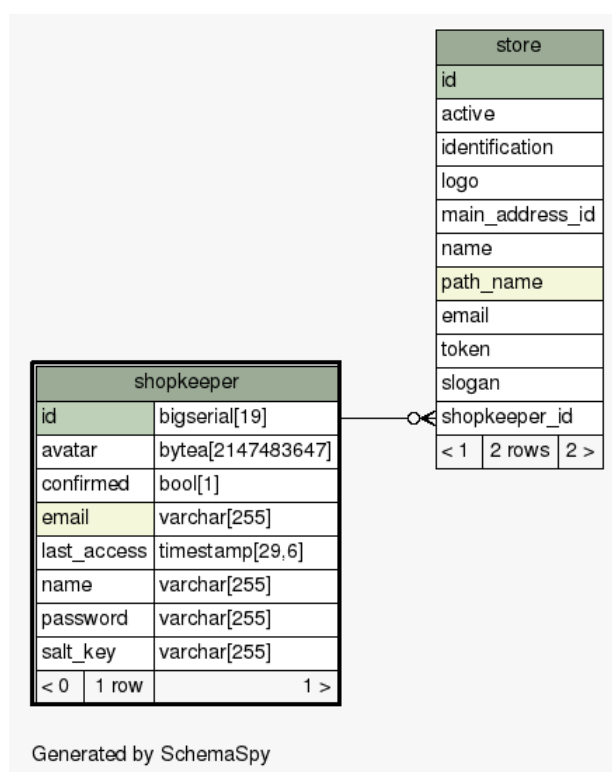


Figura 25: Produto e seus relacionamentos

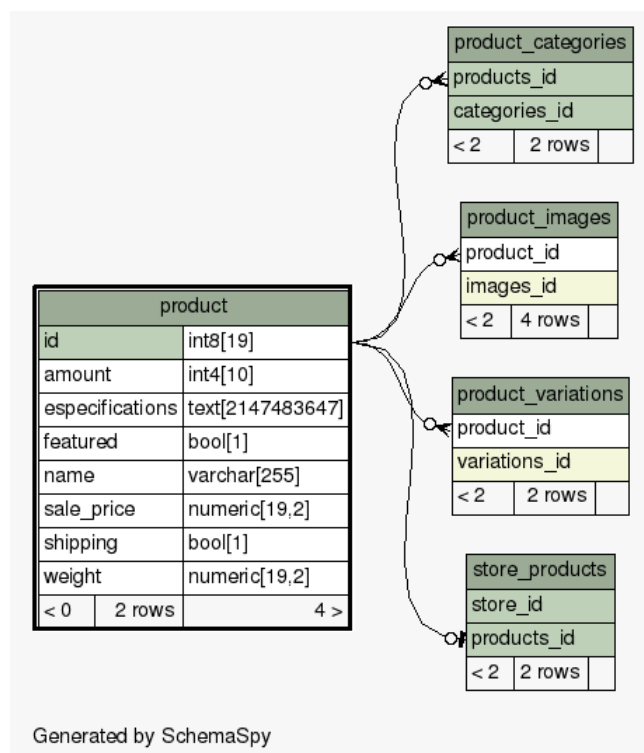


Figura 26: Categoria de Produto e seus relacionamentos

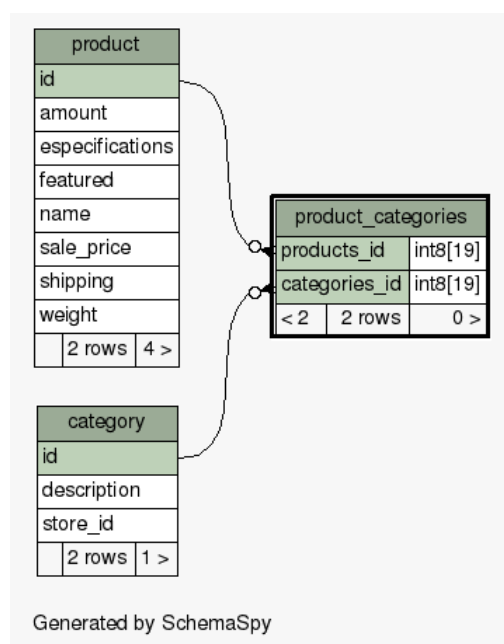


Figura 27: Variação de produto e seus relacionamentos

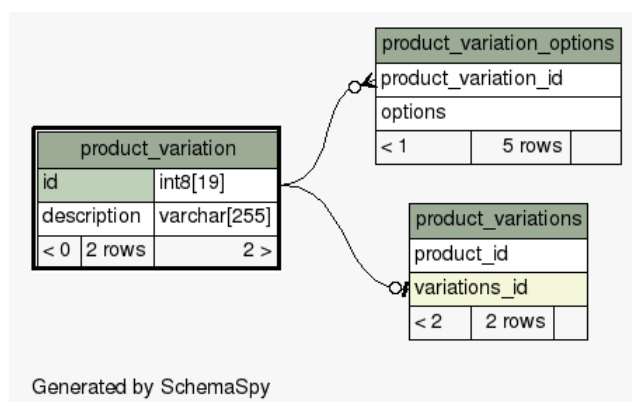


Figura 28: Comentários e seus relacionamentos

