

UNIVERSIDADE REGIONAL DE BLUMENAU
CENTRO DE CIÊNCIAS EXATAS E NATURAIS
CURSO DE SISTEMAS DE INFORMAÇÃO – BACHARELADO

**SISTEMA DE *E-COMMERCE* PARA PEQUENOS
VAREJISTAS**

LEONARDO ROSSETTI

BLUMENAU
2016

LEONARDO ROSSETTI

SISTEMA DE *E-COMMERCE* PARA PEQUENOS

VAREJISTAS

Trabalho de Conclusão de Curso apresentado ao curso de graduação em Sistemas de Informação do Centro de Ciências Exatas e Naturais da Universidade Regional de Blumenau como requisito parcial para a obtenção do grau de Bacharel em Sistemas de Informação.

Prof. Roberto Heinzle, Doutor - Orientador

**BLUMENAU
2016**

SISTEMA DE *E-COMMERCE* PARA PEQUENOS

VAREJISTAS

Por

LEONARDO ROSSETTI

Trabalho de Conclusão de Curso aprovado para obtenção dos créditos na disciplina de Trabalho de Conclusão de Curso II pela banca examinadora formada por:

Presidente: _____
Prof. Roberto Heinzle, Doutor – Orientador, FURB

Membro: _____
Prof. Paulo Fernando da Silva, Doutor – FURB

Membro: _____
Prof. Francisco Adell Pericas, Mestre – FURB

Blumenau, 07 de julho de 2016

Dedico este trabalho a todos os amigos,
especialmente aqueles que me ajudaram
diretamente na realização deste.

AGRADECIMENTOS

A Deus pela dádiva da vida.

À minha família pelo apoio de sempre.

Aos meus amigos pelos momentos de descontração.

Ao meu orientador, Roberto Heinzle, pela motivação e incentivo, por ter auxiliado nas tomadas de decisões e contribuído com boas ideias.

"Mar calmo nunca fez bom marinheiro."

[Autor desconhecido]

RESUMO

Visando uma maior agilidade e um menor custo para pequenos varejistas na geração de um *e-commerce* próprio, o objetivo geral desse trabalho é o desenvolvimento de um sistema *e-commerce* que abranja os aspectos relacionados ao fluxo de uma venda, o controle das vendas, produtos e fornecedores além de possibilitar a emissão de relatórios. O sistema consiste em uma aplicação *web*, desenvolvida na plataforma Asp.Net MVC juntamente com o banco de dados SQL Server. Esse banco de dados permite a integração com o Visual Studio, facilitando assim o desenvolvimento do trabalho. Como resultado final tem-se um sistema que possibilita o fluxo de vendas *online* e o controle de pagamento de boletos através do sistema de pagamento *online* Pague Seguro.

Palavras-chave: *E-Commerce*. *Web*. Vendas.

ABSTRACT

To enable greater agility and lower costs for small retailers to generate an e-commerce itself. The aim of this study is to develop an e-commerce system that covers all aspects necessary for the flow of a sale, control of sales, products and suppliers in addition to enabling the reporting. The system consists of a web application, developed in Asp.Net MVC platform with SQL Server database. This database allows for easy integration with Visual Studio, thus facilitating the development of this work. The end result have a system that enables online sales flow and total control of payment of bills through the online payment system PagSeguro.

Key-words: E-Commerce. Web. Sales.

LISTA DE FIGURAS

Figura 1 - Tela inicial do sistema	19
Figura 2 - Tela inicial do sistema	20
Figura 3 - Diagrama de casos de uso do sistema com foco no administrador.....	23
Figura 4 - Diagrama de casos de uso com foco no cliente	24
Figura 5 - Diagrama de classes do sistema.....	25
Figura 6 - Diagrama de atividades do sistema.....	27
Figura 7 - Modelo de Entidade-Relacionamento.....	29
Figura 8 - Representa uma requisição à API de pagamentos do PagSeguro.....	37
Figura 9 - Fluxo de consulta ao PagSeguro	39
Figura 10 - Página inicial do sistema	40
Figura 11 - Tela de Autenticação de usuário	40
Figura 12 - Tela de cadastro de usuário.....	41
Figura 13 – Tela de detalhes do produto	42
Figura 14 - Tela de carrinho de compras	43
Figura 15 - Tela de pedidos realizados	43
Figura 16 - Tela de emissão de relatório de produtos.....	44
Figura 17 - Tela de impressão de relatório	44
Figura 18 - Tela de pagamento do PagSeguro.....	45

LISTA DE QUADROS

Quadro 1: Requisitos funcionais.....	22
Quadro 2: Requisitos não funcionais.....	23
Quadro 3 - Método de envio de email	31
Quadro 4 - Salva imagens utilizando o método CreateThumbnail.....	32
Quadro 5 - Consulta informações do Cep junto ao webservice dos Correios	33
Quadro 6 - Salvar informações de forma genérica	34
Quadro 7 - Atualizar status dos pedidos junto ao PagSeguro	36
Quadro 8 - Integração com PagSeguro.....	38
Quadro 9: Descrição do caso de uso UC1	50
Quadro 10: Descrição do caso de uso UC02	51
Quadro 11: Descrição do caso de uso UC3	51
Quadro 12: Descrição do caso de uso UC4	52
Quadro 13: Descrição do caso de uso UC5	53
Quadro 14: Descrição do caso de uso UC6	53
Quadro 15: Descrição do caso de uso UC7	54
Quadro 16: Descrição do caso de uso UC8	54
Quadro 17: Descrição do caso de uso UC09	55
Quadro 18: Descrição do caso de uso UC10	56
Quadro 19: Descrição do caso de uso UC11	56
Quadro 20: Descrição do caso de uso UC12	57
Quadro 21: Descrição do caso de uso UC13	58
Quadro 22: Descrição do caso de uso UC14	59
Quadro 22 - Tabela Pedido.....	60
Quadro 23 - Tabela Usuario	61
Quadro 24 - Tabela Tipo_Usuario.....	61
Quadro 25 - Tabela CEP.....	61
Quadro 26 - Tabela Endereco	62
Quadro 27 - Tabela Logradouro	62
Quadro 28 - Tabela Fornecedor.....	62
Quadro 29 - Tabela Cidade.....	62
Quadro 30 - Tabela Bairro.....	62

Quadro 31 - Tabela Estado	63
Quadro 32 - Tabela Genero	63
Quadro 33 - Tabela Item_Pedido	63
Quadro 34 - Tabela Produto	63
Quadro 35 - Tabela Categoria	64
Quadro 36 - Tabela Sub_Categoria	64
Quadro 37 - Tabela Parametros	64

SUMÁRIO

1 INTRODUÇÃO.....	12
1.1 OBJETIVOS.....	12
1.2 ESTRUTURA.....	13
2 FUNDAMENTAÇÃO TEÓRICA	14
2.1 COMÉRCIO ELETRÔNICO	14
2.2 INFLUÊNCIAS DO COMÉRCIO ELETRÔNICO	15
2.2.1 M-COMMERCE.....	15
2.2.2 F-COMMERCE	16
2.3 FRAMEWORKS DE E-COMMERCE.....	16
2.4 PADRÕES DE PROJETO	17
2.4.1 MVC.....	18
2.5 TRABALHOS CORRELATOS	18
3 DESENVOLVIMENTO.....	21
3.1 SISTEMA PROPOSTO.....	21
3.2 ESPECIFICAÇÃO DOS REQUISITOS	22
3.3 MODELAGEM	23
3.3.1 Diagrama de casos de uso	23
3.3.2 Diagrama de classes	24
3.3.3 Diagrama de Atividades	26
3.3.4 Modelo Entidade Relacionamento	28
3.4 IMPLEMENTAÇÃO	30
3.4.1 Técnicas e ferramentas utilizadas.....	30
3.4.2 PagSeguro	35
3.4.3 Operacionalidade da implementação	39
3.5 RESULTADOS E DISCUSSÕES.....	45
4 CONCLUSÕES.....	47
4.1 EXTENSÕES	47
APÊNDICE A – DESCRIÇÃO DOS CASOS DE USO	50
APÊNDICE B – DICIONÁRIO DE DADOS.....	60

1 INTRODUÇÃO

De acordo com dados do e-Bit (2015), o varejo *online* brasileiro está crescendo aproximadamente 30% ao ano. O faturamento no Brasil já é o maior da América Latina. O segmento é movimentado, principalmente, com datas que vêm se tornando tradicionalmente propícias para a compra na internet: Dia das Mães, *Black Friday* (a sexta-feira mundial de descontos na *web*) e claro, o Natal. No próximo ano, a previsão é que o mercado do comércio eletrônico estará ainda mais maduro. Em razão dessa expectativa, varejistas esperam ter um aumento de aproximadamente 17,7% nas vendas *online* (E-BIT, 2015).

Com base no aumento de procura voltada ao comércio eletrônico, muitos empreendedores já não desejam ter uma loja física. Ao invés disso, buscam comercializar seus produtos de forma eletrônica através de um site na internet. Pode-se ver hoje grandes exemplos de varejos que não possuem de fato uma loja física, como por exemplo, as lojas do Submarino e NetShoes (NASAJON, 2013).

Por outro lado, vê-se que a esmagadora maioria das grandes redes varejistas de lojas físicas já está comercializando seus produtos também pela *web* (COMSCORE, 2012). Pode-se ver os exemplos de lojas como Casas Bahia, Magazine Luiza, Ponto Frio entre outras (COMSCORE, 2012). Em comparação às lojas físicas (onde a localização, estrutura e recursos humanos devem ser cuidadosamente estudados), o varejo virtual pode alcançar um público diverso, pois possibilita aos clientes que não possuem tempo ou até mesmo que morem longe dos grandes centros encontrar os produtos que desejam (PRESSE, 2014).

Contudo, ainda vê-se uma grande quantidade de pequenas redes varejistas de lojas físicas que tem interesse em obter um *e-commerce*, mas não o faz por não encontrar um modelo ideal de negócio. Estas lojas buscam um *e-commerce* que atenda às necessidades básicas para vender seus produtos através da internet (VALLE, 2014).

1.1 OBJETIVOS

O objetivo geral desse trabalho é o desenvolvimento de um sistema *e-commerce* para pequenos empreendimentos varejistas.

Os objetivos específicos do trabalho são:

- a) disponibilizar o gerenciamento de uma loja *online*;
- b) permitir que se faça todo o fluxo básico de uma venda e o recebimento do pagamento desta venda;
- c) disponibilizar a integração do sistema com o método de pagamento *online* PagSeguro;

- d) identificar e medir os custos associados à criação e disponibilização do *e-commerce* para as lojas.

1.2 ESTRUTURA

Este trabalho está disposto em quatro capítulos. No primeiro capítulo apresenta-se a introdução, os objetivos a serem alcançados com o desenvolvimento e a estrutura do trabalho.

No segundo capítulo apresenta-se a fundamentação teórica, destacando os conceitos de comércio eletrônico, crescente mercado de *e-commerce*, segurança em transações *online* e os riscos de comprar na internet, futuro do comércio eletrônico, *frameworks* de *e-commerce*, padrões de projeto, MVC, além de trabalhos correlatos.

No terceiro capítulo é apresentado o desenvolvimento do sistema, modelagem, técnicas e ferramentas utilizadas. A operacionalidade do sistema juntamente com os resultados e discussões.

O quarto capítulo apresenta as conclusões e sugestões de extensão e melhorias.

2 FUNDAMENTAÇÃO TEÓRICA

Este capítulo aborda assuntos a serem apresentados nas seções a seguir, tais como o conceito de comércio eletrônico, crescente mercado de *e-commerce*, segurança em transações *online* e os riscos de comprar na internet, futuro do comércio eletrônico, *frameworks de e-commerce*, padrões de projeto, MVC, além de trabalhos correlatos.

2.1 COMÉRCIO ELETRÔNICO

Para Andrade (2001, p.13) comércio eletrônico é “a aplicação de tecnologias de comunicação e informação compartilhadas entre as empresas, procurando atingir seus objetivos”. Já para Smith (2000, p.74), comércio eletrônico são:

Negócios conduzidos exclusivamente através de um formato eletrônico. Sistemas que se comunicam eletronicamente uns com os outros são sistemas de *e-commerce*, e têm de ser capazes de funcionar normalmente com quaisquer aplicações da Internet que estiver planejando utilizar. Também se refere a quaisquer funções eletrônicas que auxiliam uma empresa na condução de seus negócios.

Comércio eletrônico se trata de todos os processos envolvidos da cadeia de valor realizada em um ambiente eletrônico, utilizando de ferramentas com grande tecnologia de informação e de comunicação, tendo como principal objetivo atender as necessidades exigidas pelos negócios. Pode-se realizar de forma completa ou parcialmente, caracterizando por transações negócio a negócio, negócio a consumidor, intra-organizacional. Ou ainda pode-se definir comércio eletrônico como a compra e venda de produtos, informações e serviços através da rede mundial de computadores (NAKAMURA, 2011).

Mesmo num cenário onde se vê o brasileiro consumindo menos, o comércio eletrônico deve crescer 20% em 2015, com a inserção de mais de 40 mil lojas virtuais. Este crescimento vem também gerando novos empregos, uma vez que precisa de pessoas especializadas tanto nas áreas de venda como de finanças (MANSANO, 2015).

Cada vez mais as pessoas estão em busca de comprar sem correria, sem filas, sem ter que procurar vaga no estacionamento, receber o produto em casa, ter facilidade de pagamento, economia de tempo, infinitas possibilidades de pesquisar preços e até mesmo adquirir um produto que não existe no Brasil. Já não se vê tanto receio nas compras feitas *online*. Com isso muitas lojas já migraram para o *e-commerce* e estão contentes com os resultados. E as lojas que ainda não migraram estão a um passo de viabilizar tal investimento (COMSCORE, 2012).

Esse mercado só tende a crescer! Se você ainda acredita que o seu produto não pode ser vendido na internet, é melhor rever seus conceitos e procurar uma empresa que possa te auxiliar a encontrar a estratégia e foco certo para ofertar o produto ou serviço na *web*. (FERREIRA, 2013).

Existem, como em todos os meios de compras, certos riscos ao efetuar a compra de determinados produtos. Ainda mais quando se fala em *e-commerce*, onde o consumidor não está vendo/provando o produto, que até pode nem existir.

O Procon aconselha que o consumidor verifique se o site que está vendendo o produto possui um endereço comercial físico e anote telefones. Pode-se, ainda, checar no Procon se existem reclamações contra o site que ele pretende comprar e também verificar na Junta Comercial do Estado se há denúncias relativas à empresa (PELOGGIA, 2015).

Na hora de efetuar uma compra *online* com o cartão de crédito, é importante que o site possua os dados protegidos. Para verificar se o site é seguro o consumidor pode verificar por meio de um símbolo de um cadeado que aparece na janela do navegador. Isso significa que é um ambiente seguro e os dados não serão interceptados. O protocolo de acesso à loja virtual deve começar com <https://>. Se nenhum destes requisitos de segurança *online* forem preenchidos é melhor evitar a compra e procurar um site onde eles sejam fornecidos ou uma loja física.

2.2 INFLUENCIAS DO COMÉRCIO ELETRÔNICO

O comércio eletrônico vem sofrendo algumas mudanças nos últimos anos. Isso se deve a popularização de novas tecnologias como celulares e *tablets*, e isso tem intensificado a influência no crescimento de outras modalidades de comércio eletrônico (TARDIN, 2011). Como novas modalidades de comércio eletrônico pode-se citar o *mobile commerce* (*m-commerce*) e o *Facebook commerce* (*f-commerce*).

2.2.1 M-COMMERCE

Trata-se de toda e qualquer transação que esteja relacionada à transferência de propriedade ou direitos de uso de bens e serviços, que é começada e terminada utilizando o celular. O comércio eletrônico atual será adaptado para a modalidade *mobile commerce*, ainda em fase de desenvolvimento. Assim como próprio comércio eletrônico foi crescendo e nos dias atuais já faz parte do cotidiano, da mesma maneira o *mobile commerce* vai se desenvolver (NAKAMURA, 2011).

Diante das novas tecnologias, ferramentas e serviços de tráfego de dados para celulares cada vez melhores, com diversas funções, há uma expansão dos negócios. Tardim (2011) lista algumas aplicações *mobile* já estão disponíveis:

- a) *mobile ticketing*: são *tickets* transmitidos para celulares, onde o cliente pode apresentar na bilheteria. Pode ser usado como *ticket* para agência de viagens, diminuindo o tráfego de pessoas em ambientes como aeroportos e estacionamentos;
- b) *mobile vouchers*: o usuário recebe um cupom de descontos para ser utilizado;
- c) compra: serviço de compra de arquivos do tipo mp3, *ringtones* e papéis de parede;
- d) serviços de *location*: baseia-se na localização do usuário; a partir disso é feito incentivo a vendas de produtos da região;
- e) serviços de informação: grande quantidade de informações pode ser comercializada, desde horóscopo até informações financeiras;
- f) *mobile banking* – grandes esforços estão sendo feitos para que seja possível acessar a conta, consultar saldos e extratos, e também realizar a compra de ações e a transferência de dinheiro. Contudo, há problemas com a questão da segurança das informações e transações, o que tem impedido o crescimento;
- g) *mobile* corretora de ações: serve para que o usuário tenha em tempo real as movimentações da sua carteira, com informações atualizadas, podendo interferir de acordo com o mercado de ações;
- h) *mobile marketing* e publicidade: está em grande crescimento, o qual propõem eficácia nas propagandas enviadas para os dispositivos móveis.

2.2.2 F-COMMERCE

Define-se como um comércio eletrônico transacionado dentro do Facebook. Na prática, a transação realizada via *web* acontece da mesma forma que o comércio eletrônico tradicional, diferenciada pelo ambiente onde serão comercializados os produtos: o ambiente do Facebook.

2.3 FRAMEWORKS DE E-COMMERCE

Existem plataformas prontas para criar e personalizar lojas na internet. Essas plataformas auxiliam a criação e personalização de *e-commerces* oferecendo por vezes *layouts* prontos para melhor adequar o site à marca do cliente. Podem-se ver alguns exemplos como o Uolhost ou o Shopify, os quais oferecem esse serviço.

Uma vantagem de utilizar um *framework* de *e-commerce* para criar uma loja virtual é a integração com vários serviços como o PagueSeguro, Facebook, Correios, E-bit, Buscapé, entre outros. Essas integrações já se encontram pré-programadas, o que facilita a montagem do site. Porém pode-se considerar como sendo uma desvantagem que apenas é possível modificar o *site* dentro das limitações que o *framework* permite. Sendo assim não é possível customizar o *site* além das opções pré-programadas pelo *framework* (UOLHOST, 2015).

O crescimento significativo do número de usuários do Facebook motivou o interesse das empresas a estarem presente nessa rede social. É possível desenvolver uma loja virtual dentro do Facebook usando as ferramentas do comércio eletrônico. Esses aplicativos funcionam como vitrine dos produtos e serviços dentro da rede social. Quando alguém tentar realizar a compra virtual, ele é direcionado para o *site* do produto (PERES, 2011).

2.4 PADRÕES DE PROJETO

A ideia de padrões foi apresentada por Alexander (1977, p.470) no contexto de Arquitetura de prédios e cidades:

Cada padrão descreve um C de novo e de novo em nosso ambiente, e então descreve a parte central da solução para aquele problema de uma forma que você pode usar esta solução um milhão de vezes, sem nunca implementa-la duas vezes da mesma forma.

Dessa forma, pode-se entender um padrão de projeto, como a solução recorrente para um problema em um contexto, mesmo que em projetos e áreas distintas. Observe que os termos chaves dessa definição são: contexto, problema e solução, o que torna obrigatório à compreensão inequívoca de cada um. Um contexto diz respeito ao ambiente e as circunstâncias dentro do qual algo existe. O problema é a questão indefinida, algo que precisa ser investigado e solucionado. Normalmente, está atrelado ao contexto em que ocorre. Finalmente, a solução refere à resposta do problema que ajuda a solucioná-lo. Entretanto, se tivermos uma solução para um problema em um determinado contexto, ela não necessariamente pode constituir um padrão, pois é necessário que ela tenha como característica a regularidade, isto é, ela se constituirá como um padrão se puder ser utilizada repetidamente (DEV MEDIA, 2014).

Não pode ser considerado um padrão de projeto trecho de códigos específicos, mesmo que para o seu criador ele reflita um padrão, que soluciona um determinado problema, porque os padrões devem estar a um nível maior de abstração, e não limitado a recursos de programação. Um padrão de projeto nomeia, abstrai e identifica os aspectos chaves de uma estrutura de projeto comum para torna-la útil para a criação de um projeto orientado a objetos

reutilizável (DEV MEDIA, 2014). Dentre os principais padrões existentes está o *Model-view-controller* (MVC).

2.4.1 MVC

O MVC, em português modelo-visão-controlador, é um padrão de arquitetura de *software* que separa a representação da informação da interação do usuário com ele. O modelo (*model*) consiste nos dados da aplicação, regras de negócios, lógica e funções. Uma visão (*view*) pode ser qualquer saída de representação dos dados, como uma tabela ou um diagrama. É possível ter várias visões do mesmo dado, como um gráfico de barras para gerenciamento e uma visão tabular para contadores. O controlador (*controller*) faz a mediação da entrada, convertendo-a em comandos para o modelo ou visão. As ideias centrais por trás do MVC são a reusabilidade de código e separação de conceitos (MACORATTI, 2012).

Com o desenvolvimento e evolução dos programas e, conseqüentemente, da forma de se fazer os programas, novas abordagens tiveram que ser desenvolvidas para facilitar a programação e garantir que o *software*, depois de pronto, fosse mais facilmente manutenível. A partir disso surgiu o conceito de dividir tarefas, de garantir com que cada camada da aplicação tenha seu próprio escopo e definição e que a comunicação entre todas elas se dê de maneira eficiente e controlada (CODEIGNITERBRASIL, 2013).

2.5 TRABALHOS CORRELATOS

Esta sessão apresenta alguns trabalhos que merecem ser destacados, primeiro a monografia do aluno Rafael Capistrano do curso de Sistemas de Informação na Universidade Regional de Blumenau pela similaridade e pela utilização de tecnologias comuns as tecnologias propostas neste trabalho, tais como o Asp.Net. E o Mercado Livre por ser uma empresa de tecnologia líder no segmento de *e-commerces* no Brasil.

Capistrano (2007) desenvolveu um sistema para intermediar a negociação entre clientes e prestadores de serviços. O objetivo geral do trabalho foi o desenvolvimento do sistema de negociação para contratação de serviços através da *web* baseado em um *e-commerce*. O sistema consiste em uma aplicação *web*, desenvolvida em plataforma Java, com *framework* GWT juntamente com JPA para persistir as classes e atributos no banco de dados. Como resultado final tem-se um sistema que facilita a negociação entre prestadores de serviços e seus clientes, onde o cliente selecionava os profissionais que deseja e solicita um orçamento aos mesmos com informações referentes ao serviço desejado.

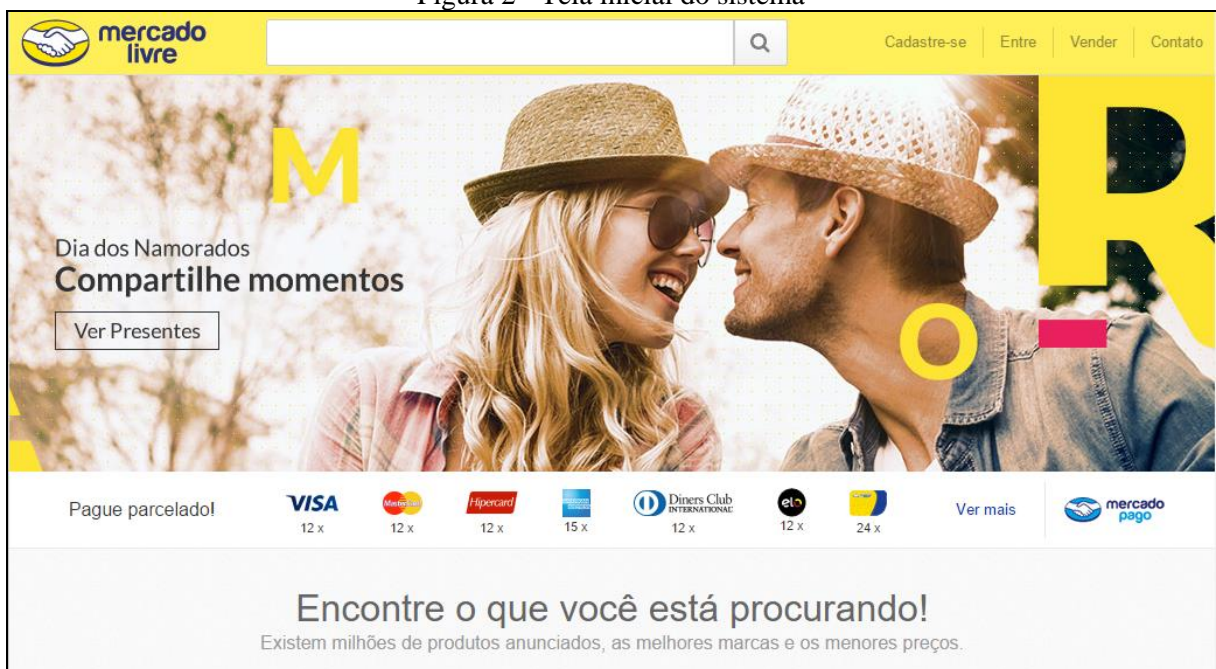
A tela apresentada na figura 1 é a página inicial do sistema. Esta tela surge logo após o usuário efetuar *login*. Nela o usuário visualiza uma lista de ofertas cadastradas pelos profissionais prestadores de serviços. Nesta tela o usuário pode efetuar a solicitação de orçamentos para os prestadores de serviços que desejar, clicando no botão contratar, encontrado ao lado da oferta apresentada (CAPISTRANO, 2007).

Figura 1 - Tela inicial do sistema



O Mercado Livre é uma empresa de tecnologia que oferece soluções de comércio eletrônico para que pessoas e empresas possam comprar, vender, pagar, anunciar e enviar produtos por meio da internet. Segundo ComScore (2012) os destinos de comércio eletrônico com maior engajamento no Brasil apontam o Mercado Livre como líder no mercado, com visitantes permanecendo em média 20 minutos no *site*, consumindo 36 páginas de conteúdo, e acessando o site em média 3,7 vezes. E ainda, o Mercado Livre responde por mais de 1 em cada 5 minutos despendidos na categoria comércio eletrônico. Na figura 2 é apresentada a tela inicial do sistema.

Figura 2 - Tela inicial do sistema



Tsujimoto (2003) mostra algumas características e aspectos do comércio eletrônico em um ambiente comercial, que com o advento da tecnologia da informação possibilitou uma nova tendência na área de suprimentos. Foi abordado também os tipos e etapas do processo de implementação de um *e-commerce*. Esse trabalho aborda também alguns modelos e estruturas encontradas dentro do ambiente do comércio eletrônico com suas vantagens e desvantagens e por fim traz os aspectos do processo de implementação de uma plataforma “*Business to Business*” com as questões técnicas, premissas necessárias e as funcionalidades que esse novo modelo de negócio proporciona.

3 DESENVOLVIMENTO

Neste capítulo estão descritas as particularidades técnicas do sistema proposto tais como a descrição do mesmo e a apresentação dos requisitos funcionais e não funcionais, diagrama de casos de uso, diagrama de classes, diagrama de atividades, técnicas e ferramentas utilizadas, integração com o PagSeguro e também serão mostradas algumas das principais telas do sistema.

3.1 SISTEMA PROPOSTO

Tendo em vista o objetivo de desenvolver um sistema para atender ao anseio do comércio varejista e de empreendedores que não desejam uma loja física em comercializar seus produtos de forma eletrônica através de um *site* na internet (*e-commerce*). O objetivo do presente trabalho é proporcionar um ciclo completo de vendas *online* (carrinho de compras, geração de boleto bancário e confirmação de pagamento pelo cliente).

O desenvolvimento do sistema proposto será dividido em três etapas: a aquisição do conhecimento com *sites* de *e-commerces* já consagrados, mapeando informações necessárias para o funcionamento completo das vendas; a elaboração de uma modelagem dos dados para demonstrar como serão construídas as estruturas de dados que darão suporte ao desenvolvimento; a criação de uma base de dados consistente que permita a fácil manipulação dos dados e o desenvolvimento da aplicação final.

Desta forma, o sistema oferecerá as seguintes funcionalidades:

- a) cadastro de produtos, clientes, categorias e subcategorias de produtos e fornecedores;
- b) cadastro de promoções de produtos individualmente, por categoria ou subcategoria;
- c) carrinho de compras para os clientes;
- d) autenticação para acessar o sistema por parte do administrador ou cliente;
- e) emissão de relatórios de vendas com filtros;
- f) emissão de boleto bancário após a venda;
- g) envio de emails de confirmação de pedido, pagamento e envio do produto ao cliente;
- h) confirmação de recebimento do pagamento;
- i) integração com os correios para buscar as informações de endereço a partir do CEP ao cadastrar um cliente ou fornecedor.

Para a construção da aplicação serão utilizadas as seguintes ferramentas:

- a) Visual Studio como plataforma de desenvolvimento;
- b) SQL Server como banco de dados para armazenamento das informações;
- c) SQL Management Studio para criação de tabelas e consulta de dados;
- d) Asp.Net como linguagem de programação.

3.2 ESPECIFICAÇÃO DOS REQUISITOS

O Quadro 1 apresenta os requisitos funcionais previstos para o sistema e sua rastreabilidade, ou seja, vinculação com os casos de uso associados.

Quadro 1: Requisitos funcionais

Requisitos Funcionais	Caso de Uso
RF01: O sistema deverá permitir manter um cadastro de produtos.	UC01
RF02: O sistema deverá permitir manter um cadastro de categoria de produtos. As categorias devem ser multinível.	UC02
RF03: O sistema deverá permitir que seja salvo um histórico de informações sobre o recebimento e armazenamento de cada produto.	UC03
RF04: O sistema deverá permitir o cadastro de promoções para os produtos obtidos, aplicando neste caso um percentual de desconto vinculando ao cadastro do produto.	UC04
RF05: O sistema deverá permitir que se diminua a quantidade dos itens do carrinho de compras sem que se perca o item, mas também deverá permitir que se remova o item.	UC05
RF06: O sistema deverá permitir a emissão de relatórios (são exemplos de relatórios: produtos vendidos por período, produtos mais/menos vendidos e compras realizadas por período).	UC06
RF07: O sistema deverá permitir manter um cadastro de fornecedores.	UC07
RF08: O sistema deverá permitir manter um cadastro de usuários e classificar tipos de usuário devendo existir ao menos os tipos Administrador e Cliente.	UC08
RF10: O sistema deverá permitir que o usuário possa alterar os dados do seu cadastro.	UC09
RF11: O sistema deverá permitir confeccionar pedidos de venda de produtos.	UC10
RF12: O sistema deverá permitir que se verifique quais vendas foram realizadas, quais delas estão com seu pagamento pendente e informar para quando o produto já foi devidamente enviado ao cliente.	UC11

O Quadro 2 lista os requisitos não funcionais previstos para o sistema.

Quadro 2: Requisitos não funcionais

Requisitos Não Funcionais
RNF01: O sistema deverá operar com o <i>web</i> browser Chrome, da empresa Google.
RNF02: O sistema deverá ser desenvolvido na linguagem C# .Net da empresa Microsoft.
RNF03: O sistema deverá utilizar o banco de dados SQL Sever da empresa Microsoft.

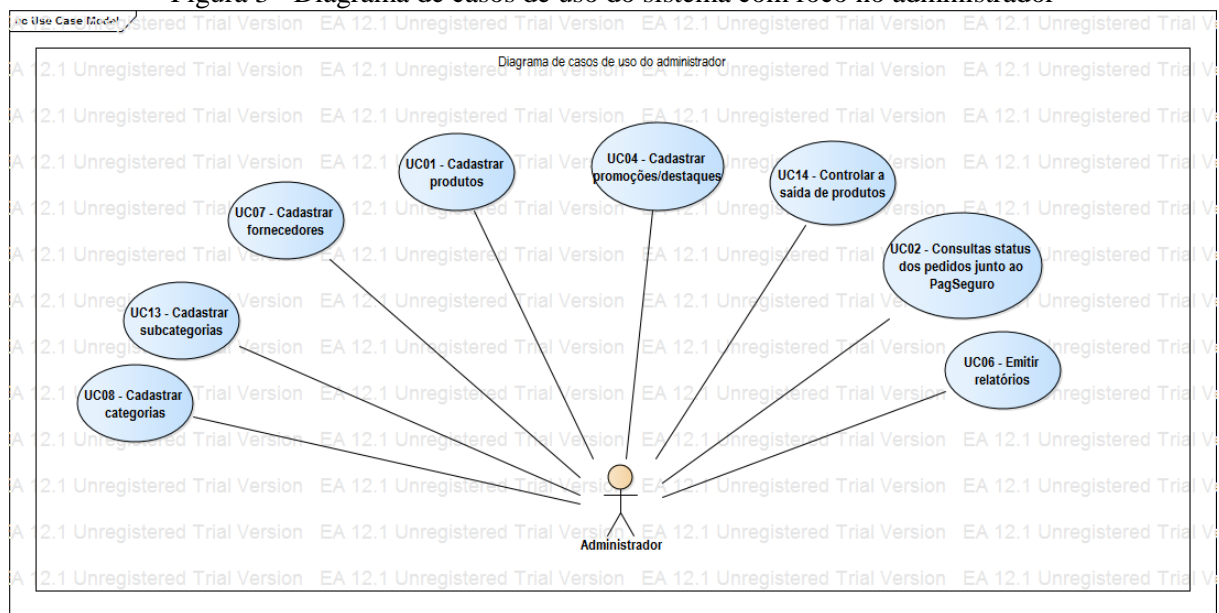
3.3 MODELAGEM

Nesta seção serão apresentados os diagramas de casos de uso e de classes do sistema.

3.3.1 Diagrama de casos de uso

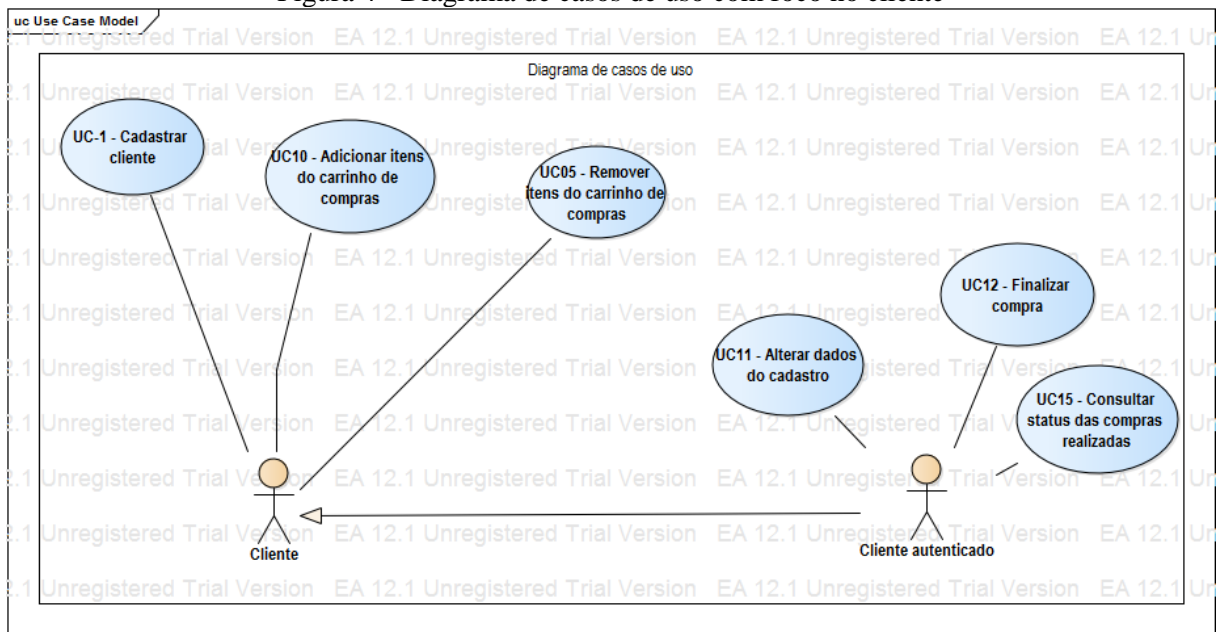
A figura 3 apresenta o diagrama de casos de uso preliminares do sistema proposto tendo foco no administrador. Como pode ser observado, este ator interage com o sistema através de nove casos de uso, dentre eles pode-se destacar os casos de uso de cadastro de produtos, atender pedidos e emitir relatórios.

Figura 3 - Diagrama de casos de uso do sistema com foco no administrador



Já a figura 4 apresenta o diagrama de casos de uso preliminares do sistema com foco no cliente. Como pode ser observado o ator representando o cliente pode interagir com o sistema mesmo sem efetuar *login*, porém, somente após isto ele poderá finalizar a compra de produtos junto ao PagSeguro e consultar as compras já realizadas.

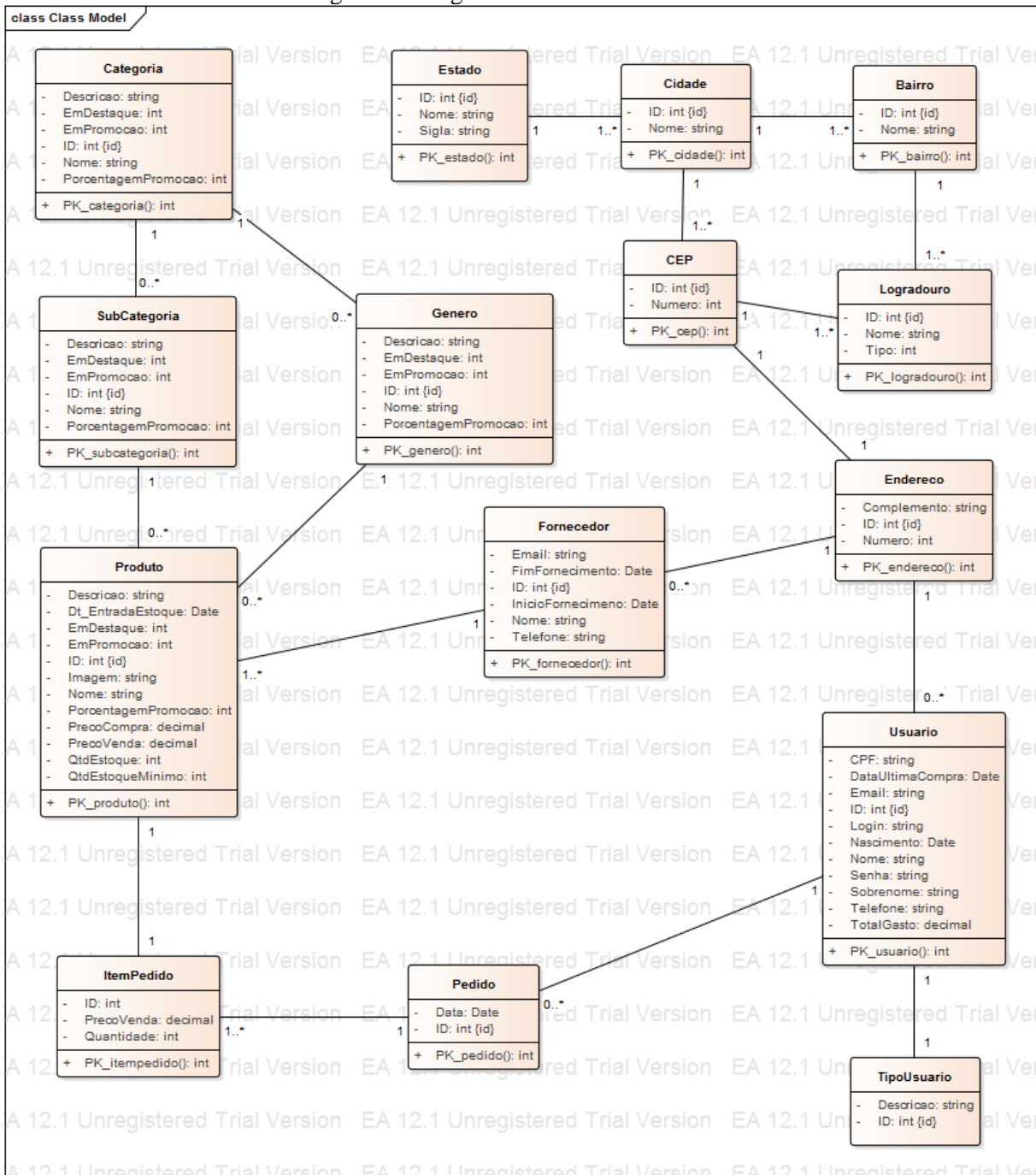
Figura 4 - Diagrama de casos de uso com foco no cliente



3.3.2 Diagrama de classes

Nesta seção são apresentados os diagramas de classes do sistema. A aplicação foi dividida em cinco pacotes, seguindo o modelo *Model View Controller* (MVC), detalhados a seguir: *models*, *views*, *controller*, *classes*. Nas próximas subseções são apresentadas as classes, relacionamentos e estruturas desenvolvidas em cada pacote. A Figura 5 ilustra os pacotes da aplicação e suas respectivas classes.

Figura 5 - Diagrama de classes do sistema



A função de cada uma das classes apresentada no diagrama da figura 5 é detalhada a seguir:

- classe Estado: classe que possui os atributos referentes ao um estado;
- classe Cidade: classe que possui os atributos referentes ao uma cidade;
- classe Bairro: classe que possui os atributos referentes ao um bairro;
- classe CEP: classe que possui os atributos referentes ao um CEP;
- classe Logradouro: classe que possui os atributos referentes ao um logradouro;

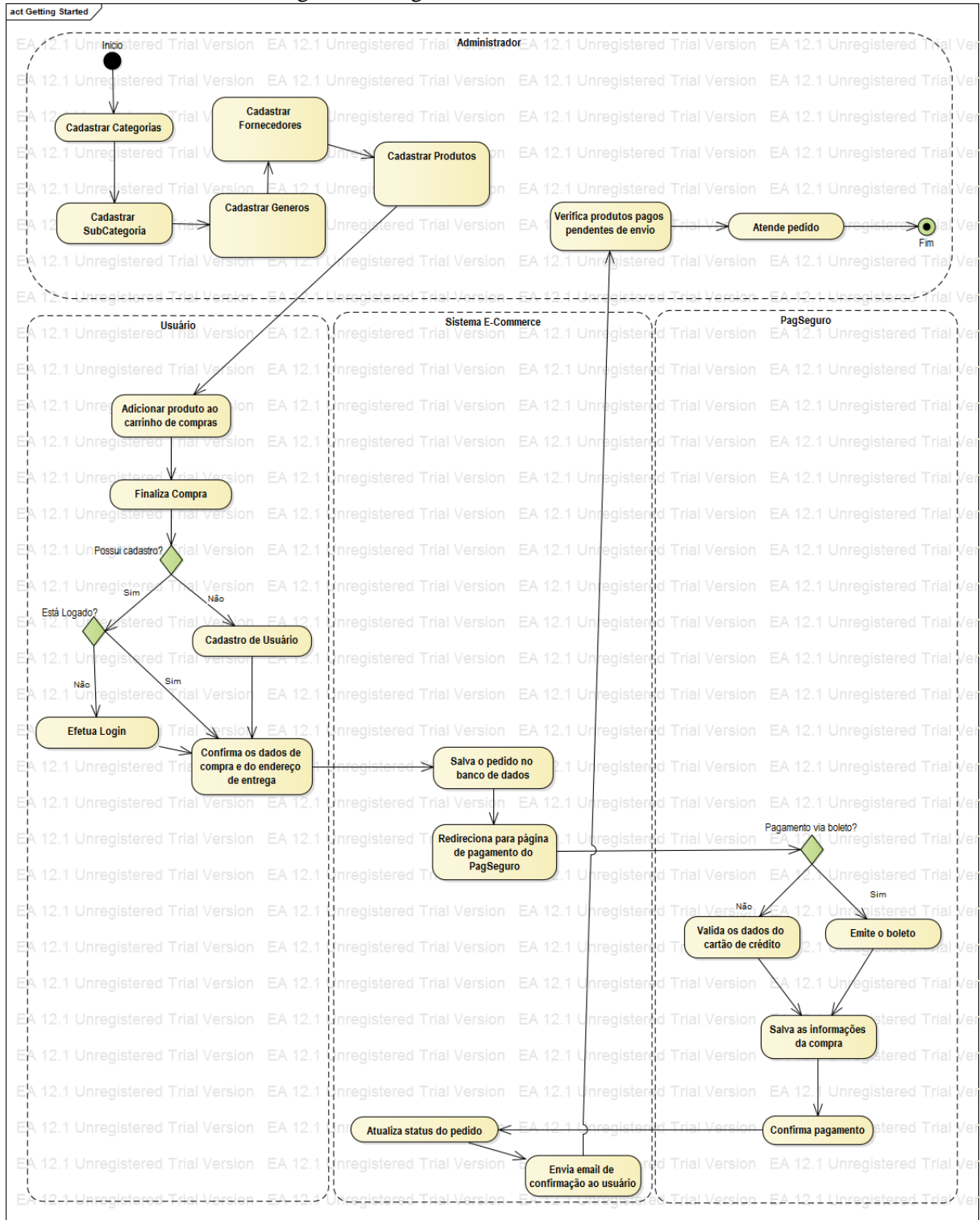
- f) classe `SubCategoria`: classe que possui os atributos referentes a uma subcategoria;
- g) classe `Categoria`: classe que possui os atributos referentes a uma categoria;
- h) classe `Endereco`: classe que possui os atributos referentes a um endereço, nesse caso, o endereço aponta somente para o CEP;
- i) classe `Fornecedor`: classe que possui os atributos referentes a um fornecedor;
- j) classe `Produto`: classe que possui os atributos referentes a um produto;
- k) classe `Usuario`: classe que possui os atributos referentes a um usuário;
- l) classe `ItemPedido`: classe que possui os atributos referentes a um item do pedido. Lembrando que o item do pedido aponta para um produto;
- m) classe `Pedido`: classe que possui os atributos referentes a um pedido. Lembrando que um Pedido aponta para um ou mais itens do pedido e para uma forma de pagamento;
- n) classe `TipoUsuario`: classe que possui os atributos referentes a um determinado tipo de usuário;
- o) classe `Genero`: classe que possui os atributos referentes a um gênero.

3.3.3 Diagrama de Atividades

O diagrama de atividades da figura 6 apresenta o fluxo principal de funcionamento do sistema desenvolvido. Para dar início ao processo, a primeira ação deve ser do administrador que deve cadastrar os produtos no sistema. Para tal, é necessário que exista pelo menos uma categoria, subcategoria, gênero e fornecedores cadastrados. Após o cadastro de produtos, o usuário pode adicionar produtos ao carrinho de compras e, posteriormente, finalizar a compra. Em sequência, é necessário que o usuário esteja autenticado no sistema. Para tal ele precisa efetuar um cadastro, caso ainda não o tenha feito. Após se autenticar no sistema, o usuário deve confirmar os dados da compra e do endereço de envio dos produtos. Ao confirmar os dados o sistema salva o pedido na base de dados, envia o email de confirmação ao usuário e redireciona o cliente para a página do PagSeguro onde este deverá selecionar a forma de pagamento desejada. O usuário terá duas opções de pagamento ativas, sendo elas cartão de crédito ou boleto bancário. Se o usuário informar que deseja pagar com o cartão de crédito o PagSeguro deverá validar as informações do cartão e efetuar o pagamento. Já se o usuário informar que deseja pagar via boleto bancário, o PagSeguro irá incrementar um valor na compra como taxa do serviço de geração do boleto. Após a confirmação do pagamento, o

sistema deverá atualizar as informações dos pedidos. Em sequência, o administrador deve verificar quais pedidos estão com o pagamento efetuado e atender os mesmos, alterando o status destes pedidos no sistema para atendidos.

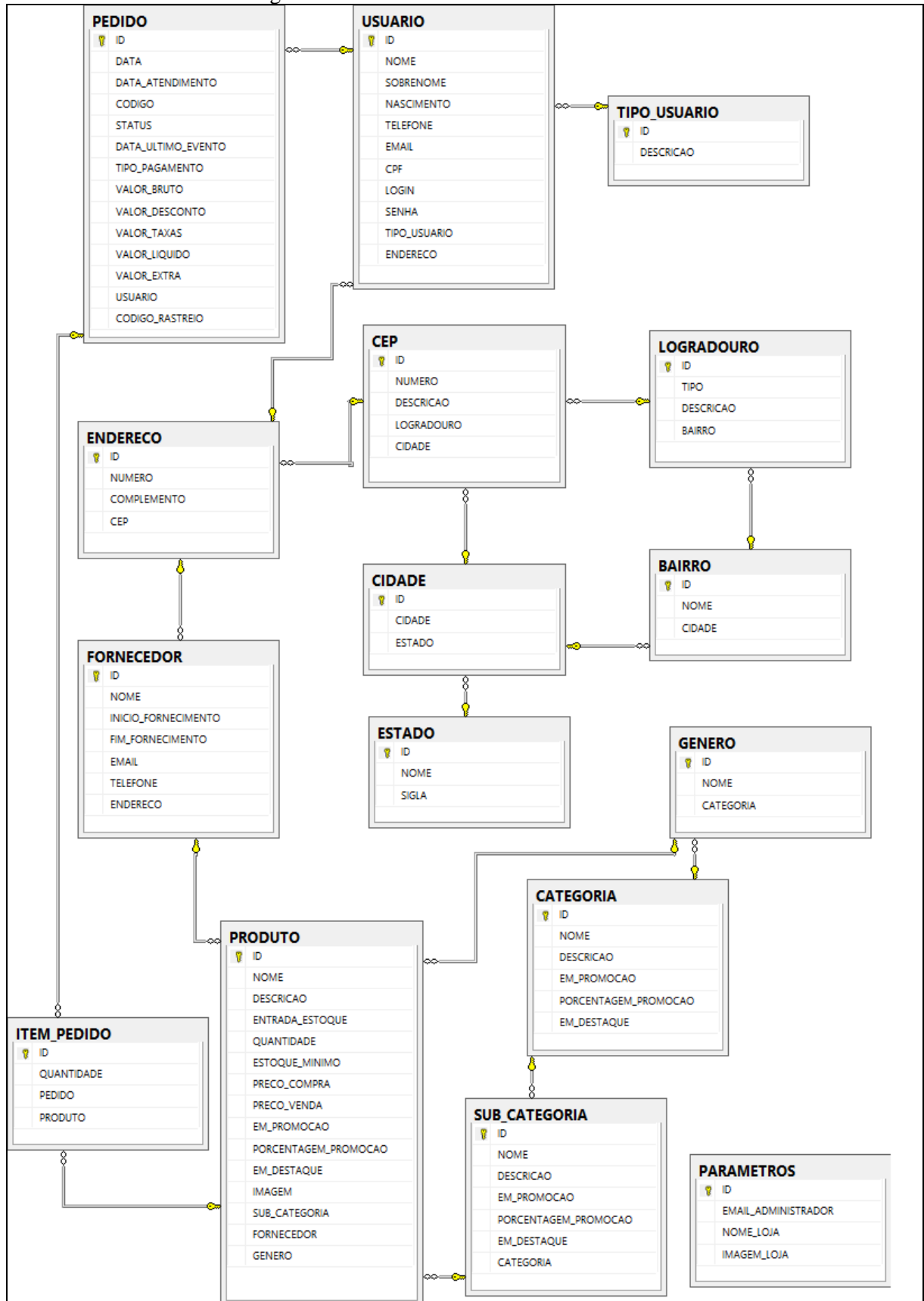
Figura 6 - Diagrama de atividades do sistema



3.3.4 Modelo Entidade Relacionamento

Na figura 7 apresenta-se o Modelo de Entidade-Relacionamento (MER) da base de dados utilizada pelo sistema.

Figura 7 - Modelo de Entidade-Relacionamento



A seguir estão listadas as tabelas do sistema e o tipo de dados que armazenam. O dicionário de dados completo encontra-se no apêndice B:

- a) PEDIDO: pedidos realizados no sistema;
- b) USUARIO: usuários cadastrados no sistema;
- c) TIPO_USUARIO: tipos de usuário;
- d) CEP: CEP do endereço;
- e) ENDERECO: guarda informações do endereço, como o complemento e o número;
- f) LOGRADOURO: logradouro do endereço;
- g) FORNECEDOR: fornecedor de produtos;
- h) CIDADE: contém todas as cidades cadastradas no Brasil até 2015;
- i) BAIRRO: bairro do endereço;
- j) ESTADO: contém todos os estados cadastrados no Brasil até 2015;
- k) GENERO: gênero do produto;
- l) ITEM_PEDIDO: contém um item de um pedido;
- m) PRODUTO: tabela de produtos;
- n) CATEGORIA: tabela de categorias;
- o) SUB_CATEGORIA: tabela de subcategorias de produtos;
- p) PARAMETROS: parâmetros do sistema.

3.4 IMPLEMENTAÇÃO

A seguir são mostradas as técnicas e ferramentas utilizadas e a operacionalidade da implementação.

3.4.1 Técnicas e ferramentas utilizadas

Para o desenvolvimento do sistema, foi utilizada a linguagem de programação C# (.NET Framework), através da IDE de desenvolvimento Visual Studio 2013. Foi utilizada também o padrão de arquitetura de software MVC, além da linguagem de programação JavaScript e do Framework JQuery. Como banco de dados foi utilizado o SQL Server 2012.

3.4.1.1 .NET Framework

O .NET Framework é uma plataforma de desenvolvimento para a criação de aplicativos para Windows, Windows Phone, Windows Store, Windows Server e Windows Azure. A plataforma .NET Framework inclui as linguagens de programação C# e Visual Basic, o Common Language Runtime (CLR) e uma ampla biblioteca de classes, disponível em <http://msdn.microsoft.com/pt-br/vstudio/aa496123>.

O quadro 3 demonstra o trecho de código fonte responsável por enviar os emails de confirmação do pedido após a realização de uma compra. Os emails são enviados através da classe MailMessage do .Net. Para que seja possível exibir as imagens de cada produto listado referente ao pedido, é necessário enviar as imagens como anexo no email. Isto é feito com a classe Attachment do .Net.

Quadro 3 - Método de envio de email

```
public string EnvioEmail(string destinatario, string assunto, string
cabecalho, string conteudo, List<Attachment> anexos)
{
    string remetenteEmail = "testerossetti@gmail.com";
    MailMessage mail = new MailMessage();
    mail.To.Add(destinatario);
    mail.From = new MailAddress(remetenteEmail, cabecalho,
System.Text.Encoding.UTF8);
    mail.Subject = assunto;
    mail.SubjectEncoding = System.Text.Encoding.UTF8;
    mail.Body = conteudo;
    mail.BodyEncoding = System.Text.Encoding.UTF8;
    mail.IsBodyHtml = true;
    mail.Priority = MailPriority.High; //Prioridade do E-Mail

    foreach (var item in anexos)
    {
        mail.Attachments.Add(item);
    }

    SmtplibClient client = new SmtplibClient(); //credenciais
    client.Credentials = new System.Net.NetworkCredential(remetenteEmail,
"testerossetti123");

    client.Port = 587; //porta
    client.Host = "smtp.gmail.com"; //provedor
    client.EnableSsl = true; //Server Secured Layer
    try
    {
        client.Send(mail);
        return "Envio do E-mail com sucesso.";
    }
    catch (Exception ex)
    {
        return "Ocorreu um erro ao enviar email: " + ex.Message;
    }
}
```

O quadro 4 demonstra o trecho de código fonte responsável por salvar as imagens dos produtos cadastrados no sistema utilizando o método CreateThumbnail do .Net. As imagens são salvas em dois tamanhos diferentes, para que sejam exibidas na tela inicial e na tela de detalhes com tamanhos distintos.

Quadro 4 - Salva imagens utilizando o método CreateThumbnail

```

private void SaveImage(HttpPostedFileBase arquivo)
{
    String path = Server.MapPath("~/Content/images/Produtos/" +
arquivo.FileName);

    if (arquivo.ContentLength > 0)
    {
        arquivo.SaveAs(path);

        #region Imagem Maior
        using (Bitmap bitmap = new Bitmap(arquivo.InputStream,
false))
            if (bitmap.Width > bitmap.Height)
            {
                int wi = bitmap.Width;
                int he = bitmap.Height;
                int novaAltura = (he * 400) / wi;
                CreateThumbnail(400, novaAltura, path,
Server.MapPath("~/Content/images/Produtos/Maior/" + arquivo.FileName));
            }
            else
            {
                int he = bitmap.Height;
                int wi = bitmap.Width;
                int novaLargura = (wi * 400) / he;
                CreateThumbnail(novaLargura, 400, path,
Server.MapPath("~/Content/images/Produtos/Maior/" + arquivo.FileName));
            }
        #endregion

        #region Imagem Menor
        using (Bitmap bitmap = new Bitmap(arquivo.InputStream,
false))
            if (bitmap.Width > bitmap.Height)
            {
                int wi = bitmap.Width;
                int he = bitmap.Height;
                int novaAltura = (he * 150) / wi;
                CreateThumbnail(150, novaAltura, path,
Server.MapPath("~/Content/images/Produtos/Menor/" + arquivo.FileName));
            }
            else
            {
                int he = bitmap.Height;
                int wi = bitmap.Width;
                int novaLargura = (wi * 150) / he;
                CreateThumbnail(novaLargura, 150, path,
Server.MapPath("~/Content/images/Produtos/Menor/" + arquivo.FileName));
            }
        #endregion
    }
}

public void CreateThumbnail(int largura, int altura, string pastaOrigem,
string pastaDestino)
{
    System.Drawing.Image img = System.Drawing.Image.FromFile(pastaOrigem);
    System.Drawing.Image imgthumb = img.GetThumbnailImage(largura,

```

```

altura, null, new System.IntPtr(0));
    imgthumb.Save(pastaDestino, ImageFormat.Jpeg);
    img.Dispose();
    imgthumb.Dispose();
}

```

O quadro 5 demonstra o trecho de código fonte responsável por consultar informações referentes ao endereço do usuário a partir do Cep informado. Essas informações são buscadas junto ao Webservice disponibilizado pelos correios. Esse código é executado quando o usuário preenche o campo Cep na tela de cadastro de usuário.

Quadro 5 - Consulta informações do Cep junto ao webservice dos Correios

```

/// <summary>
/// Consulta o endereço a partir do CEP informado pelo usuário
/// </summary>
public ActionResult ConsultaCEP(string cep)
{
    try
    {
        Object umUsuario = null;
        CidadeDao cidadeDao = new CidadeDao();
        UFDao UFDao = new Classes.AcessosBanco.UFDao();

        if (cep.Replace("-", "").Length == 8)
        {
            DataSet ds = new DataSet();
            ds.ReadXml("http://cep.republicavirtual.com.br/web_cep.php?cep="
+ cep.Replace("-", "").Trim() + "&formato=xml");
            string _resultado = "0";
            if (ds != null)
            {
                if (ds.Tables[0].Rows.Count > 0)
                {
                    _resultado = ds.Tables[0].Rows[0]["resultado"].ToString();
                    string siglaEstado = string.Empty;
                    string cidade = string.Empty;
                    string rua = string.Empty;
                    string bairro = string.Empty;
                    switch (_resultado)
                    {
                        case "0":
//nao encontrou o cep informado. Nesse caso seta estado e cidade
padrao
                            siglaEstado = "AC";
                            break;
                        case "1":
                        case "2":
                            siglaEstado = ds.Tables[0].Rows[0]["uf"].ToString().Trim();
                            cidade = ds.Tables[0].Rows[0]["cidade"].ToString().Trim();
                            rua = ds.Tables[0].Rows[0]["logradouro"].ToString().Trim();
                            bairro = ds.Tables[0].Rows[0]["bairro"].ToString().Trim();
                            break;
                    }

                    int idEstado = UFDao.BuscarCodigoUF(siglaEstado);
                    Dictionary<string, string> dicionarioDados = new
Dictionary<string, string>();
                    dicionarioDados.Add("ESTADO", idEstado.ToString());
                    List<Cidade> Cidades = cidadeDao.BuscarCidades(dicionarioDados);

```

```

        string cidadeSelecionada = cidade == string.Empty ?
Cidades[0].Id.ToString() : cidade;
        umUsuario = new
        {
            IdEstado = idEstado,
            Rua = rua,
            Bairro = bairro,
            IdCidade = cidadeDao.BuscarCodigoCidade(idEstado,
cidadeSelecionada),
            ListaCidades = Cidades,
            TotalCidades = Cidades.Count
        };}}}}}}

```

O quadro 6 demonstra o trecho de código fonte responsável por salvar de forma genérica qualquer coleção de valores passados em uma tabela especificada. É passado um dicionário de dados contendo o nome do campo e o valor referente àquele campo. Dessa forma é montada a instrução SQL a partir desse dicionário.

Quadro 6 - Salvar informações de forma genérica

```

/// <summary>
/// Salva dados em uma tabela específica
/// </summary>
/// <param name="tabela">Tabela onde serão salvos os dados</param>
/// <param name="campos">Dicionario contendo <nomeCampo,
valorCampo></param>
/// <returns></returns>
public void Salvar(string tabela, Dictionary<string, string> campoValor)
{
    try
    {
        StringBuilder campos = new StringBuilder();
        StringBuilder valores = new StringBuilder();

        int aux = 0;
        foreach (var item in campoValor)
        {
            aux++;
            if (!(string.IsNullOrEmpty(item.Key)
                && string.IsNullOrEmpty(item.Value)))
            {
                campos.Append(item.Key);
                valores.Append(item.Value);

                if (aux < campoValor.Count)
                { //verifica se tem próximo registro adiciona virgula
                    campos.Append(",");
                    valores.Append(",");
                }
            }
        }

        SqlCommand cmd = new SqlCommand("INSERT INTO " + tabela + " (" +
campos + ") VALUES (" + valores + ")", conexao.GetConnection());
        conexao.GetConnection.Open();
        cmd.ExecuteNonQuery();
    }
    catch (Exception e)
    {
        throw new Exception("Erro ao buscar registros. Erro: " + e.Message);
    }
}

```

```
}  
finally  
{  
    conexao.GetConnection.Close();  
}}
```

3.4.1.2 SQL Server 2012

O Microsoft SQL Server é um sistema gerenciador de banco de dados relacional desenvolvido pela Microsoft. Foi desenvolvido para ser totalmente integrado ao .NET Framework e devido a esse motivo foi escolhido para este trabalho. O uso de SQL Server 2012 em conjunto com o .NET 4.0 representa mais um grande ganho de produtividade para o desenvolvimento.

3.4.2 PagSeguro

Buscou-se uma solução de comércio eletrônico para transações comerciais através de pagamentos *onlines* ou móveis. Com o PagSeguro, lojas ou pessoas podem fazer compras e vendas seguras na internet ou receber pagamentos através do leitor de cartão de débito e crédito para celular e *tablet*. A biblioteca PagSeguro em .NET é um conjunto de classes de domínio que facilitam, para o desenvolvedor .NET, a utilização das funcionalidades que o PagSeguro oferece na forma de APIs.

No quadro 7 demonstra-se um trecho do código fonte referente à tela de Pedidos Realizados. Esse código faz a consulta junto ao site do PagSeguro para atualizar os status de todos os pedidos. A consulta é feita através de parâmetros contendo as credenciais do vendedor, data inicial e data final de vendas, número de páginas que será retornada e quantidade de registros por página.

Quadro 7 - Atualizar status dos pedidos junto ao PagSeguro

```

public static TransactionSearchResult SearchAbandoned(Credentials
credentials, DateTime initialDate, DateTime finalDate, int? pageNumber =
null, int? resultsPerPage = null)
{
    PagSeguroTrace.Info(String.Format(CultureInfo.InvariantCulture,
"TransactionSearchService.SearchAbandoned(initialDate={0},finalDate={1}) -
begin", initialDate, finalDate));
    try
    {
        using (HttpWebResponse response = HttpURLConnectionUtil
.GetHttpGetConnection(BuildSearchUrlAbandoned(credentials, initialDate,
finalDate, pageNumber, resultsPerPage))
        {
            using (XmlReader reader = XmlReader
.Create(response.GetResponseStream()))
            {
                TransactionSearchResult result =
new TransactionSearchResult();
                TransactionSearchResultSerializer.Read(reader, result);
                PagSeguroTrace.Info(String.Format(CultureInfo.InvariantCulture,
"TransactionSearchService.SearchAbandoned(initialDate={0},finalDate={1}) -
end {2}", initialDate, finalDate, result));
                return result;
            }
        }
    }
    catch (WebException exception)
    {
        PagSeguroServiceException pse = HttpURLConnectionUtil
.CreatePagSeguroServiceException((HttpWebResponse)exception.Response);
        PagSeguroTrace.Error(String.Format(CultureInfo.InvariantCulture,
"TransactionSearchService.SearchAbandoned(initialDate={0},finalDate={1}) -
error {2}", initialDate, finalDate, pse));
        throw pse;
    }
}

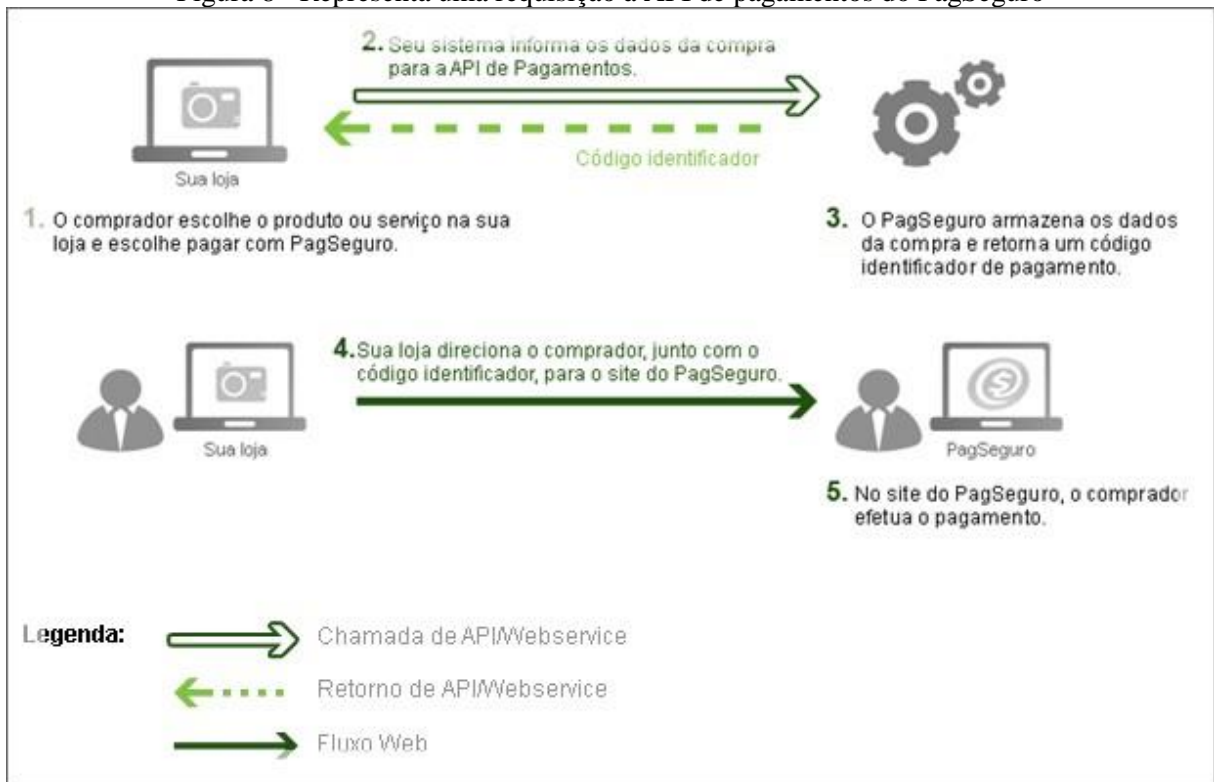
```

Fonte: PagSeguro (2016).

3.4.2.1 API de Pagamentos do PagSeguro

Para iniciar uma requisição de pagamento, é preciso fazer uma chamada à API de Pagamentos do PagSeguro informando os dados do pagamento. Esta requisição cria um código identificador e, com este código, pode-se direcionar o comprador para o *site* do PagSeguro, onde ele realizará o pagamento. Uma forma de fazer a requisição à API de Pagamentos é informar os dados do pagamento diretamente em parâmetros HTTP utilizando o método POST. Esta forma é bastante similar ao Pagamento via HTML, com a diferença que a chamada é feita à API de Pagamentos e não ao formulário de pagamento, conforme mostrado na figura 8.

Figura 8 - Representa uma requisição à API de pagamentos do PagSeguro



Fonte: PagSeguro (2016).

O quadro 8 demonstra o trecho de código fonte responsável por efetuar a integração junto a API de pagamento do PagSeguro. Como se pode observar está se direcionando para o ambiente de testes do PagSeguro, chamado Sandbox. É neste ambiente que o comprador efetua o pagamento.

Quadro 8 - Integração com PagSeguro

```

//Adiciona itens no formato para envio ao pagSeguro
PaymentRequest payment = new PaymentRequest();
foreach (var item in itensPedido)
{
    payment.Items.Add(new Item(item.Produto.ProdutoId.ToString(),
    item.Produto.Nome, item.Quantidade, (decimal)item.Produto.ValorVenda,
    1000, 10.00m));
}

//adiciona dados do vendedor para envio ao pagSeguro
payment.Sender = new Sender(
    pedido.Usuario.Nome + " Teste",
    "v16666697896298074199@sandbox.pagseguro.com.br",
    new Phone(
        pedido.Usuario.Telefone.Substring(1, 2),
        pedido.Usuario.Telefone.Substring(4)
    )
);

//Informa o endereço de envio fornecido pelo comprador
payment.Shipping = new Shipping();
payment.Shipping.ShippingType = ShippingType.Sedex;

payment.Shipping.Address =
    new Address("BRA",
        pedido.Usuario.UF,
        pedido.Usuario.Cidade,
        pedido.Usuario.Bairro,
        pedido.Usuario.Cep,
        pedido.Usuario.Rua,
        pedido.Usuario.Numero,
        pedido.Usuario.Complemento);

//É necessário que você informe a moeda em que o comprador irá realizar o
pagamento. No momento, a única opção disponível é BRL (Real).
payment.Currency = Currency.Brl;

payment.RedirectUri = new Uri("https://sandbox.pagseguro.uol.com.br");

//Agora que você informou os dados da compra, você deve executar o método
Register, que faz a requisição à API de Pagamentos do PagSeguro,
//retornando a URL necessária para o comprador fazer o pagamento:
AccountCredentials credentials = new AccountCredentials(
    "leoo.rossetti@gmail.com",
    "6543638CC79947F6A349E29EB1C3DD01"
);

payment.Reference = idPedido.ToString();

Uri _paymentRedirectUri = payment.Register(credentials);

string paymentRedirectUri = _paymentRedirectUri.ToString();

if (!string.IsNullOrEmpty(paymentRedirectUri))
{
    Response.AppendHeader("Access-Control-Allow-Origin",
    "https://sandbox.pagseguro.uol.com.br");
    Response.Redirect(paymentRedirectUri);
}
else

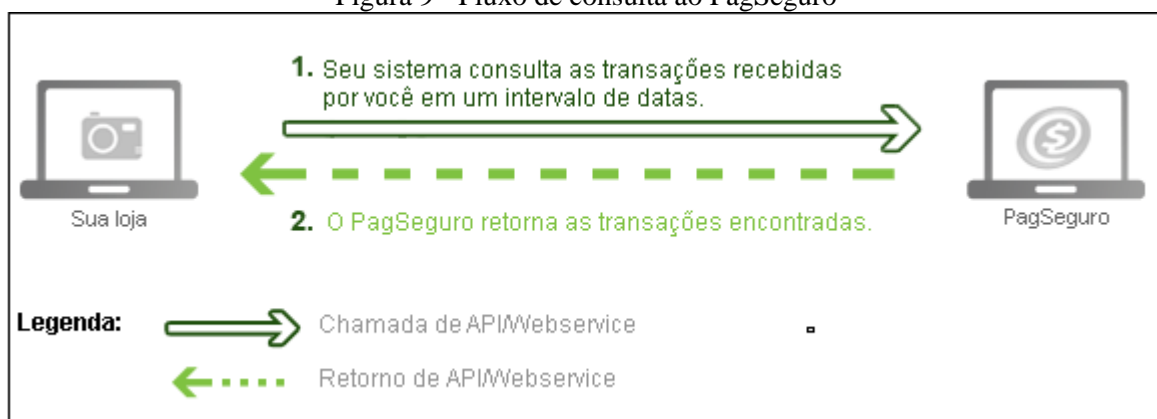
```

```
{
    ViewBag.Mensagem = paymentRedirectUri;
    return View();
}
```

3.4.2.2 API de consulta do PagSeguro

A API de consulta do PagSeguro permite obter as transações recebidas em um intervalo de datas. Ela pode ser usada para verificar se o sistema recebeu as notificações de transações enviadas pelo PagSeguro, de forma a conciliar as transações armazenadas no sistema com o PagSeguro. Na figura 9 é possível verificar o fluxo de uma consulta a esta API.

Figura 9 - Fluxo de consulta ao PagSeguro



Fonte: PagSeguro (2016).

3.4.3 Operacionalidade da implementação

Nesta subseção apresentam-se as telas do sistema com uma apresentação sobre suas funcionalidades. Todos os dados presentes nas telas são fictícios e servem apenas para exemplificar a apresentação da aplicação.

3.4.3.1 Tela inicial do Sistema

A tela apresentada na figura 10 é a página inicial do sistema. Esta é a primeira tela que surge quando o usuário acessa o sistema, sem a necessidade de se fazer *login*. Nessa tela o usuário pode optar por se direcionar a tela de *login*, se cadastrar no sistema, pesquisar por produtos através de um filtro por categoria listado a esquerda dos produtos, ver detalhes dos produtos clicando no link de detalhes e adicionar produtos no carrinho de compras clicando no botão comprar que fica abaixo de cada produto.

Figura 10 - Página inicial do sistema



3.4.3.2 Tela de Autenticação do Usuário

A figura 11 apresenta a tela de autenticação do usuário. Nela o usuário deve informar seu *login* e senha para se autenticar e entrar no sistema. Caso o usuário não possua cadastro, deve clicar no link Cadastrar-se.

Figura 11 - Tela de Autenticação de usuário

3.4.3.3 Tela de Cadastro de Usuário

Na figura 12 é apresentada a tela de cadastro de usuário, que surge no momento em que o usuário clica no link Cadastrar-se. Nessa tela o usuário deve preencher os campos solicitados com suas informações pessoais e clicar em Salvar.

Figura 12 - Tela de cadastro de usuário

Cadastro de Usuario			
Nome	<input type="text"/>	Sobrenome	<input type="text"/>
Nascimento	<input type="text"/>	CPF	<input type="text"/>
Telefone	<input type="text"/>	E-mail	<input type="text"/>
Referente ao Endereço			
CEP	<input type="text"/>		
UF	Acre	Cidade	Acrelândia
Bairro	<input type="text"/>	Rua	<input type="text"/>
Numero	<input type="text"/>	Complemento	<input type="text"/>
Referente ao Login			
Login	<input type="text"/>		
Senha	<input type="text"/>		
Confirmação da senha	<input type="text"/>		
Salvar		Limpar	

3.4.3.4 Tela de Detalhes dos Produtos

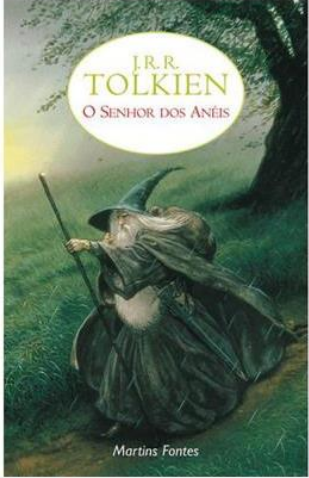
Na figura 13 é apresentada a tela de detalhes dos produtos onde é possível verificar maiores detalhes de algum produto. Essa tela será exibida quando o usuário clicar no botão Detalhes de algum produto. Nessa tela é possível também verificar os produtos similares ao produto selecionado. A consulta por produtos similares é feita através da categoria e gênero de cada produto.

Figura 13 – Tela de detalhes do produto


[Voltar](#)

Detalhes: Livro - O Senhor dos Anéis - Volume Único

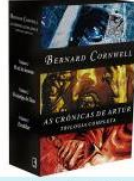
Preço	R\$ 159,90
Detalhes	Este livro reúne em um único volume a trilogia O Senhor dos Anéis, que já vendeu milhões de cópias em todo o mundo e continua encantando pessoas de todas as idades. A Sociedade do Anel, As duas torres, O retorno do rei, respectivamente primeiro, segundo e terceiro volume em dose única para os aficionados por Tolkien, ou simplesmente para os que mergulham pela primeira vez num emaranhado de acontecimentos improváveis, mas absolutamente convincente em seus detalhes.




Produtos Similares



Livro - Star Wars: A Tr...
R\$ 39,00
[Detalhes](#)



Livro - Box As Crônicas...
R\$ 99,00
[Detalhes](#)



As Crônicas de Nárnia
R\$ 49,90
[Detalhes](#)


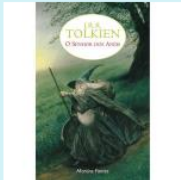
3.4.3.5 Tela de Carrinho de Compras

Na figura 14 é apresentada a tela de carrinho de compras, que surge no momento em que o usuário adiciona algum item no carrinho de compras. Nessa tela é possível ver quais produtos estão no carrinho, quantidade de cada produto, o valor dos produtos e a soma total dos produtos contidos no carrinho.

Figura 14 - Tela de carrinho de compras

Carrinho de compras

[Continuar Comprando](#)

Produto	Nome	Quantidade	Preço Un.	Preço Total	Remove
	Livro - Star Wars: A Trilogia - Special Edition	2 Aumentar Diminuir	R\$ 39,00	R\$ 78,00	Remove
	Livro - O Senhor dos Anéis - Volume Único	1 Aumentar Diminuir	R\$ 159,90	R\$ 159,90	Remove
Total:		R\$ R\$ 237,90			

Finalizar Compra

3.4.3.6 Tela de Pedidos Realizados

Na figura 15 é apresentada a tela de pedidos realizados, onde são apresentados todos os pedidos realizados, podendo verificar os detalhes dos pedidos, os detalhes do cliente e quais dos pedidos estão pendentes de pagamento ou de envio. Somente o administrador do sistema tem acesso a essa tela.

Figura 15 - Tela de pedidos realizados

Pedidos realizados

[Voltar](#)

[Atualizar Status](#)

Nº Pedido	Data	Status	Tipo de Pagamento	Código do Pedido	Cliente	Itens
51	04/05/2016	Disponível	Boleto	5B31DEA3-16CD-40B2-8A53-1E99D2B37A5F	Leonardo	Itens Atendido em: 06/05/2016
52	05/05/2016	Paga	Boleto	97F012B9-345C-4ABF-B527-37269DD67DB8	Leonardo	Itens Atender

3.4.3.7 Telas de Emissão de Relatórios

Conforme especificado, o sistema *e-commerce* deve disponibilizar a emissão de relatórios. A figura 16 apresenta um dos principais relatórios que o sistema disponibiliza, sendo ele o relatório de produtos. Nessa tela pode-se filtrar por categoria, subcategoria, gênero e fornecedor dos produtos. Nesse relatório podemos ver, dentre outras informações a

quantidade vendida de cada produto, valor de compra, de venda e a maior quantidade vendida em um único pedido.

Figura 16 - Tela de emissão de relatório de produtos

Produtos
[Voltar](#)

Filtros

Categoria: Subcategoria:

Genero: Fornecedor:

Filtrar **Imprimir**

Nome	Qtd. Estoque	Em destaque	Em promoção	Entrada estoque	Valor compra	Valor venda	Total vendido	Maior venda
The Walking Dead	23	Sim	Não	22/04/2016	R\$ 18,00	R\$ 38,00	0	0
Batman vs Superman	29	Não	Não	22/04/2016	R\$ 20,00	R\$ 40,00	0	0
HD Externo 1TB	150	Não	Não	10/10/2010	R\$ 150,00	R\$ 259,00	0	0
Livro - Star Wars: A Trilogia - Special Edition	100	Não	Não	10/10/2010	R\$ 20,00	R\$ 39,00	0	0
Livro - Box As Crônicas de Artur	50	Não	Não	10/10/2016	R\$ 50,00	R\$ 99,00	0	0
Livro - O Senhor dos Anéis - Volume Único	99	Não	Sim	10/03/2016	R\$ 75,00	R\$ 159,90	1	1
As Crônicas de Nárnia	50	Não	Não	11/05/2016	R\$ 20,00	R\$ 49,90	0	0
Call of Duty	50	Não	Não	11/05/2016	R\$ 49,00	R\$ 89,90	0	0

Pode-se também imprimir o relatório, ao clicar no botão imprimir será impresso o conteúdo atual do *grid* através da função *print* do navegador, conforme figura 17.

Figura 17 - Tela de impressão de relatório

Imprimir

Total: 1 página

Destino:

Páginas: Tudo por exemplo, 1-5, 8, 11-13

Layout:

Tamanho do papel:

Margens:

Opções: Cabeçalhos e rodapés

12/05/2016 about:blank

Nome	Qtd. Estoque	Em destaque	Em promoção	Entrada estoque	Valor compra	Valor venda	Total vendido	Maior venda
The Walking Dead	23	Sim	Não	22/04/2016	R\$ 18,00	R\$ 38,00	0	0
Batman vs Superman	29	Não	Não	22/04/2016	R\$ 20,00	R\$ 40,00	0	0
HD Externo 1TB	150	Não	Não	10/10/2010	R\$ 150,00	R\$ 259,00	0	0
Livro - Star Wars: A Trilogia - Special Edition	100	Não	Não	10/10/2010	R\$ 20,00	R\$ 39,00	0	0
Livro - Box As Crônicas de Artur	50	Não	Não	10/10/2016	R\$ 50,00	R\$ 99,00	0	0
Livro - O Senhor dos Anéis - Volume Único	99	Não	Sim	10/03/2016	R\$ 75,00	R\$ 159,90	1	1
As Crônicas de Nárnia	50	Não	Não	11/05/2016	R\$ 20,00	R\$ 49,90	0	0
Call of Duty	50	Não	Não	11/05/2016	R\$ 49,00	R\$ 89,90	0	0

3.4.3.8 Tela de pagamento do PagSeguro

Na figura 18 é apresentada a tela de pagamento do PagSeguro, onde são apresentados os produtos, podendo verificar os valores do pedido, e especificar o tipo de pagamento desejado.

Figura 18 - Tela de pagamento do PagSeguro

pagseguro | SANDBOX UOL beta

(47) 9621-1418 Leonardo Rossetti
leao.rossetti@gmail.com

Você está em ambiente de teste

Olá, Vendedor de Testes

Pagamento

Cartão de crédito
Em até 18x*

Boleto

Número do cartão

Data de validade MM AA

VISA 12x 18x e outros

Nome do dono do cartão

Código de segurança ?

Ex.: CARLOS A F DE OLIVEIRA

[Todas as formas de pagamento](#)

Resumo da compra

Endereço de entrega
89012-201
Teste
10 - teste
Teste
Blumenau - SC

Descrição	Valor
Game Uncharted 4: A Thiefs End - PS4 Quantidade: 2 Valor do item: R\$ 169,90	R\$ 339,80
Console PlayStation 4 500GB + Controle Dualshock 4 Quantidade: 1 Valor do item: R\$ 1.799,91	R\$ 1.799,91
Subtotal	R\$ 2.139,71
Frete: Sedex	R\$ 30,00
Total a pagar	R\$ 2.169,71

3.5 RESULTADOS E DISCUSSÕES

O presente trabalho apresentou a criação de um sistema *e-commerce* desenvolvido na plataforma Asp.Net MVC. Com este sistema é possível efetuar as principais funcionalidades de um *e-commerce*, tais como cadastro de usuário, fornecedores, produtos, promoções, geração de relatórios, além de efetuar vendas integradas a plataforma do PagSeguro e controlar todas as vendas realizadas. O diferencial deste trabalho para os demais citados como correlatos está na utilização de uma linguagem de programação de fácil manutenção, permitindo que o sistema possa ser dinâmico atendendo a diferentes nichos de mercado.

O trabalho desenvolvido por Capistrano (2007) com o objetivo de desenvolver um sistema de negociação para contratação de serviços através da web tem relação com este trabalho por apresentar um modelo de comércio eletrônico, porém se distancia deste por apresentara um sistema que facilita a negociação entre prestadores de serviços e seus clientes, onde o cliente selecionava os profissionais que deseja e solicita um orçamento aos mesmos com informações referentes ao serviço desejado.

O sistema de *e-commerce* do Mercado Livre tem relação com este trabalho por se tratar de um *e-commerce* e por permitir algumas funcionalidades parecidas, como a busca por produtos similares e a emissão de relatórios. Além disso, o Mercado Livre é referência no Brasil quando se fala de comércio eletrônico. Porém o Mercado Livre permite que pessoas e empresas possam comprar, vender, pagar, anunciar e enviar produtos por meio da internet. Enquanto este trabalho foi feito para atender a uma empresa que deseja vender seus produtos por meio da internet.

O trabalho desenvolvido por Tsujimoto (2003) com o objetivo de mostrar algumas características e aspectos do comércio eletrônico em um ambiente comercial tem relação com este trabalho por abordar os tipos e etapas do processo de implementação de um *e-commerce*. Porém se distancia deste trabalho por tratar apenas dos aspectos técnico-administrativos de um *e-commerce*. Esse trabalho aborda também alguns modelos e estruturas encontradas dentro do ambiente do comércio eletrônico com suas vantagens e desvantagens e por fim traz os aspectos do processo de implementação de uma plataforma “*Business to Business*” com as questões técnicas, premissas necessárias e as funcionalidades que esse novo modelo de negócio proporciona.

Durante o desenvolvimento deste trabalho, as dificuldades encontradas estavam relacionadas à própria linguagem e à criação de um modelo de classes que permitisse uma melhor comunicação entre os componentes, modo de iteração entre os componentes MVC, no envio de emails e na busca por uma solução de integração entre PagSeguro e o sistema. Os testes se restringiram ao navegador Chrome da empresa Google, porém, em conformidade com um requisito não funcional estabelecido desde a proposta deste trabalho.

4 CONCLUSÕES

O objetivo de se desenvolver um sistema que disponibilizasse as principais funcionalidades de um *e-commerce* foi atingido. Os testes junto ao PagSeguro foram bem sucedidos, ainda que tenha sido utilizado apenas o ambiente de homologação disponibilizado pelo próprio PagSeguro.

As ferramentas utilizadas para a elaboração deste trabalho mostraram-se adequadas. A utilização do *framework* .Net facilitou a integração com o banco de dados SQL Server, a criação de um sistema *web* em MVC e a integração com o PagSeguro. Os objetivos específicos deste trabalho foram atingidos, sendo eles: disponibilizar o gerenciamento de uma loja online, permitir que se faça todo o fluxo básico de uma venda e disponibilizar a integração do sistema com um método de pagamento online, tal como o PagSeguro.

O desenvolvimento deste trabalho contribuiu positivamente para o desenvolvimento pessoal e profissional do autor. Gerou maior conhecimento e experiência com programação e manipulação de dados, bem como de boas práticas no desenvolvimento com a plataforma .Net e na criação de sistemas de transações *online*.

Conclui-se com este trabalho a percepção da grande oportunidade que existe no mercado de *e-commerces* para pequenos empresários. O desenvolvimento deste trabalho proporcionou ao autor do mesmo um entusiasmo em dar continuidade ao sistema, aumentando as possibilidades de transformar o sistema em um negócio próprio.

4.1 EXTENSÕES

Como extensão deste trabalho sugere-se:

- a) integrar ao sistema a uma ferramenta de busca baseada em inteligência artificial, para facilitar ao cliente a pesquisa por produtos do seu interesse;
- b) implementar gráficos na geração de relatórios para uma melhor visualização dos resultados;
- c) implementar recurso para o compartilhamento de informações sobre os produtos por parte dos usuários.
- d) implementar um recurso para comparar as informações dos produtos cadastrados com sites de comparação de preços como o Buscapé. Isso facilitaria ao lojista colocar um preço mais competitivo em seus produtos e entender o porquê de alguns dos seus produtos não estarem com um volume de vendas muito alto, uma vez que outros sites podem estar oferecendo o mesmo produto por um preço mais acessível.

REFERÊNCIAS

- ALEXANDER, Christopher, et al. **A pattern language**. New York: Oxford University Press, 1977.
- ANDRADE, Rogério de. **Guia prático de e-commerce**. 1. ed. São Paulo: Angra, 2001.
- AVEIRO, Conrado Gonçalves de. **As estratégias competitivas do e-commerce brasileiro**. [S.I.], 2013. Disponível em: <<http://www.faceq.edu.br/e-faceq/downloads/numero03/2-as-estrategias-competitivas-do-e-commerce-brasileiro.pdf>>. Acesso em: 20 dez. 2015.
- CAPISTRANO, Rafael de. **Sistema de negociação para prestadores de serviços através de e-commerce**. Trabalho de Conclusão de Curso (Graduação em Sistemas de Informação) – Universidade Regional de Blumenau, Blumenau, 2011. Disponível em: <<http://dsc.inf.furb.br/arquivos/tccs/apresentacoes/TCC2011-2-26-AP-RafaelCapistrano.pdf>>. Acesso em: 16 set. 2015.
- CODEIGNITERBRASIL. **MVC (Model – View – Controller)**. [S.I.], 2013. Disponível em: <<http://codeigniterbrasil.com/passos-iniciais/mvc-model-view-controller/>>. Acesso em: 3 set. 2015.
- COMSCORE. **Comércio online continua a crescer no Brasil**. [S.I.], 2012. Disponível em: <<http://www.comscore.com/por/Imprensa-e-eventos/Press-Releases/2012/3/Online-Shopping-Continues-its-Ascent-in-Brazil>>. Acesso em: 18 set. 2015.
- DEVMEDIA. **Conheça os padrões de projeto**. [S.I.], 2014. Disponível em: <<http://www.devmedia.com.br/conheca-os-padroes-de-projeto/957#ixzz3qO8qbNRA>>. Acesso em: 3 nov. 2015.
- E-BIT. **Portal e-commerce Brasil**. [S.I.], 2015. Disponível em: <<https://www.ecommercebrasil.com.br/noticias/e-bit-divulga-31-relatorio-webshoppers-2015/>>. Acesso em: 16 set. 2015.
- FELIPINI, Dailton. **A vez do pequeno na internet**. [S.I.], 2014. Disponível em: <http://www.e-commerce.org.br/artigos/pequeno_empreendedor.php>. Acesso em: 20 set. 2015.
- FERREIRA, Gustavo. **O e-commerce é uma oportunidade crescente**. [S.I.], 2013. Disponível em: <<http://www.ceschini.com.br/2013/01/o-e-commerce-e-uma-oportunidade-crescente>> Acesso em 14 set. 2015.
- MACORATTI, José Carlos. **ASP.NET, MVC (Model-View-Controller)**. [S.I.], 2012. Disponível em: <http://www.macoratti.net/08/06/asp_mvc1.htm>. Acesso em: 3 nov. 2015.
- MANSANO, Fernando; **Setor de comércio pela internet cresce em meio à crise financeira no Brasil**. Ribeirão Preto, SP, 2015. Disponível em: <<http://g1.globo.com/sp/ribeirao-preto-franca/bom-dia-cidade/videos/t/edicoes/v/setor-de-comercio-pela-internet-cresce-em-meio-a-crise-financeira-no-brasil/4420641>> Acesso em: 14 set. 2015.
- NAKAMURA, André Massami. **Comércio eletrônico riscos nas compras pela internet**. São Paulo, 2011. Disponível em: <<http://www.fatecsp.br/dti/tcc/tcc0027.pdf>>. Acesso em: 20 set. 2015.
- NASAJON, Claudio. **Empresas apostam no caminho inverso: do online para o offline**. [S.I.], 2013. Disponível em: <<http://www.claudionasajon.com.br/empresas-apostam-no-caminho-inverso-do-online-para-o-offline/>>. Acesso em: 4 nov. 2015.

NASCIMENTO, André R.; SANTOS, Gisele G.; SILVA, Bruna F. **E-commerce: o melhor caminho no mercado atual**. Marília, 2009. Disponível em: <http://aberto.univem.edu.br/bitstream/handle/11077/496/E-commerce%3A%20O%20Melhor%20Caminho%20no%20Mercado%20Atual.pdf?sequence=1> Acesso em: 4 nov. 2015.

PELOGGIA, Natália. **Procon de Campinas**. São Paulo, 2015. Disponível em: <http://www.procon.campinas.sp.gov.br/compras-pela-internet-procon-d-dicas-seguran> Acesso em: 20 set. 2015.

PERES, Emmanuelle de Moura. **F-Commerce: facebook sua loja virtual**. [S.I.], 2011. Disponível em: <http://pro-manu.blogspot.com/2011/08/f-commerce-facebook-sua-loja-virtual.html>. Acesso em: 03 out. 2014.

PRESSE, **E-commerce para pequenos negócios**. Blumenau, SC, 2014. Disponível em: <http://www.presse.inf.br/e-commerce-para-pequenos-negocios/>. Acesso em: 19 out. 2015.

SMITH, Rob. ; SPEAKER, Mark. ; THOMPSON, Mark. **O mais completo guia sobre e-commerce** 1. ed. São Paulo: Futura, 2000.

TARDIN, Vicente. **Mobile commerce é o comércio eletrônico no celular**. [S.I.], 2011. Disponível em: <http://webinsider.uol.com.br/2011/10/31/mobile-commerce-e-o-comercio-eletronico-no-celular/>. Acesso em: 15 set. 2015.

TSUJIMOTO, Marcelo. **Comércio Eletrônico: aspectos e processo de implementação**. Taubaté, 2003. Monografia (Especialização, Gerencia Empresarial e Negócios). Departamento de Economia, Contabilidade, Administração e Secretariado – ECASE, Universidade de Taubaté, Taubaté. Disponível em: <http://www.ppga.com.br/mba/2003/gen/tsujimoto-marcelo.pdf> Acesso em: 20 mai. 2016.

UOLHOST. **Como criar uma loja virtual**. [S.I.], 2015. Disponível em: <http://www.uolhost.com.br/lp/loja-virtual.html?psid=1#rml>. Acesso em: 10 set. 2015.

VALLE, Alberto. **Como montar uma pequena loja virtual**. [S.I.], 2014. Disponível em: <http://www.empreendedoresweb.com.br/como-montar-uma-pequena-loja-virtual/#>. Acesso em: 19 out. 2015.

APÊNDICE A – Descrição dos Casos de Uso

Esta seção apresenta a descrição dos casos de uso conforme previstos no diagrama apresentado na seção 3.3.1.

No quadro 9 apresenta-se o caso de uso “Cadastrar produtos”.

Quadro 9: Descrição do caso de uso UC1

Caso de uso	UC01 - Cadastrar produtos
Descrição	Permite ao administrador informar os dados de um novo produto no sistema, bem como alterar ou excluir informações dos produtos já cadastrados.
Ator	Administrador
Pré-condição	<ul style="list-style-type: none"> • O Administrador deve ter efetuado o login no sistema; • Devem existir um ou mais registros de fornecedores cadastrados no sistema; • Devem existir um ou mais registros de categorias cadastradas no sistema; • Devem existir um ou mais registros de subcategorias cadastradas no sistema; • Deve existir um ou mais gêneros cadastrados no sistema.
Fluxo principal	a) o sistema informa os produtos já cadastrados; b) o administrador opta por alterar, apagar ou cadastrar um novo produto.
Fluxo alternativo (a)	<ul style="list-style-type: none"> • campo(s) obrigatório(s) não preenchido(s); • alerta com a mensagem “Favor preencher todos os campos obrigatórios” é apresentada.
Cenário - Edição	<ol style="list-style-type: none"> 1. Sistema informa os produtos cadastrados; 2. Administrador seleciona um produto para edição; 3. Sistema mostra as informações do produto selecionado; 4. Administrador realiza as alterações necessárias; 5. Administrador seleciona a opção salvar; 6. Sistema altera o registro e apresenta a mensagem “Produto alterado com sucesso!”.
Cenário – Inclusão	<ol style="list-style-type: none"> 1. Sistema mostra a tela de cadastro de produtos; 2. Administrador informa todos os campos obrigatórios; 3. Administrador informa a opção cadastrar; 4. Sistema cadastra o registro e apresenta a mensagem “Produto cadastrado com sucesso!”.
Cenário – Exclusão	<ol style="list-style-type: none"> 1. Sistema informa os produtos cadastrados; 2. Administrador seleciona um produto para exclusão e clica no botão excluir ao lado do produto listado; 3. Sistema exclui o registro e apresenta a mensagem “Produto excluído com sucesso!”.
Pós-condição	Usuário editou, apagou ou cadastrou um produto.

No quadro 10 apresenta-se o caso de uso “Consultar status dos pedidos junto ao PagSeguro”.

Quadro 10: Descrição do caso de uso UC02

Caso de uso	UC02 - Consultar status dos pedidos junto ao PagSeguro
Descrição	Permite ao administrador verificar os status dos pedidos junto ao pagSeguro.
Ator	Administrador
Pré-condição	<ul style="list-style-type: none"> • O Administrador deve ter efetuado o login no sistema; • Devem haver uma ou mais vendas realizadas.
Fluxo principal	<p>a) o sistema informa as vendas realizadas;</p> <p>b) o administrador opta por atualizar os status dos pedidos junto ao pagSeguro.</p>
Pós-condição	Os status de todos os pedidos são atualizados.

No quadro 11 apresenta-se o caso de uso “Controlar a entrada de produtos”.

Quadro 11: Descrição do caso de uso UC3

Caso de uso	UC03 - Controlar a entrada de produtos
Descrição	Permite ao administrador controlar/acompanhar o recebimento e correto armazenamento do produto e alimentar o sistema com essas informações.
Ator	Administrador
Pré-condição	O Administrador deve ter efetuado o login no sistema
Fluxo principal	<p>a) o sistema informa os produtos cadastrados;</p> <p>b) o administrador opta por alterar, apagar ou cadastrar um novo incidente para o produto.</p>
Fluxo alternativo (a)	<ul style="list-style-type: none"> • campo(s) obrigatório(s) não preenchido(s); • alerta com a mensagem “Favor preencher todos os campos obrigatórios” é apresentada.
Cenário - Edição	<ol style="list-style-type: none"> 1. Sistema informa os produtos cadastrados; 2. Administrador seleciona um produto que contenha incidente(s) para edição; 3. Sistema mostra as informações de incidente para o produto selecionado; 4. Administrador realiza as alterações necessárias; 5. Administrador seleciona a opção salvar; 6. Sistema altera o registro e apresenta a mensagem “Incidente alterado com sucesso!”.
Cenário – Inclusão	<ol style="list-style-type: none"> 1. Sistema mostra os produtos cadastrados; 2. Administrador seleciona algum produto de clica no botão de adicionar incidente ao lado do produto listado; 3. Sistema mostra a tela de cadastro de incidentes; 4. Administrador informa todos os campos obrigatórios; 5. Administrador informa a opção cadastrar; 6. Sistema cadastra o registro e apresenta a mensagem “Incidente cadastrado com sucesso!”.
Cenário – Exclusão	<ol style="list-style-type: none"> 1. Sistema informa os incidentes cadastrados; 2. Administrador seleciona um incidente para exclusão e clica no botão excluir ao lado do incidente listado;

	3. Sistema exclui o registro e apresenta a mensagem “Incidente excluído com sucesso!”.
Pós-condição	Usuário editou, apagou ou cadastrou um incidente.

No quadro 12 apresenta-se o caso de uso “Cadastrar promoções”.

Quadro 12: Descrição do caso de uso UC4

Caso de uso	UC04 - Cadastrar promoções
Descrição	Permite ao administrador cadastrar promoções para determinados produtos do sistema, aplicando neste caso um percentual de desconto vinculando ao cadastro do produto.
Ator	Administrador
Pré-condição	a) O Administrador deve ter efetuado o login no sistema; b) Devem existir produtos cadastrados no sistema.
Fluxo principal	a) o sistema informa os produtos cadastrados; b) o administrador opta por alterar, apagar ou cadastrar uma nova promoção para o produto.
Fluxo alternativo (a)	<ul style="list-style-type: none"> • campo(s) obrigatório(s) não preenchido(s); • alerta com a mensagem “Favor preencher todos os campos obrigatórios” é apresentada.
Cenário - Edição	<ol style="list-style-type: none"> 1. Sistema informa os produtos cadastrados; 2. Administrador seleciona um produto para edição; 3. Administrador realiza as alterações necessárias; 4. Administrador seleciona a opção salvar; 5. Sistema altera o registro e apresenta a mensagem “Produto alterado com sucesso!”.
Cenário – Inclusão	<ol style="list-style-type: none"> 1. Sistema informa os produtos cadastrados; 2. Administrador seleciona algum produto para edição; 3. Sistema exibe informações do produto selecionado; 4. Administrador que existe promoção para este produto; 5. Administrador informa a porcentagem de promoção; 6. Administrador informa a opção salvar; 7. Sistema cadastra o registro e apresenta a mensagem “Produto alterado com sucesso!”.
Cenário – Exclusão	<ol style="list-style-type: none"> 1. Sistema informa os produtos cadastrados; 2. Administrador seleciona um produto para edição; 3. Administrador informa que não existe promoção para este produto; 4. Sistema exclui a promoção e apresenta a mensagem “Produto alterado com sucesso!”.
Pós-condição	Usuário editou, apagou ou cadastrou uma promoção.

No quadro 13 apresenta-se o caso de uso “Remover itens do carrinho de compras”.

Quadro 13: Descrição do caso de uso UC5

Caso de uso	UC05 - Remover itens do carrinho de compras
Descrição	Permite ao cliente remover itens já adicionados ao carrinho de compras.
Ator	Cliente
Pré-condição	O Cliente deve ter incluído itens no carrinho de compras.
Fluxo principal	a) o sistema informa os produtos contidos no carrinho de compras com suas devidas quantidades; b) o cliente opta por diminuir a quantidade ou apagar um produto do carrinho de compras.
Cenário – Exclusão	<ol style="list-style-type: none"> 1. Sistema informa os produtos cadastrados; 2. Cliente seleciona produto e clica no botão remover exibido ao lado de cada item no carrinho; 3. Sistema exclui o produto do carrinho e apresenta a mensagem “Produto excluído com sucesso!”.
Pós-condição	Usuário apagou um item do carrinho de compras.

No quadro 14 apresenta-se o caso de uso “Emitir relatórios”.

Quadro 14: Descrição do caso de uso UC6

Caso de uso	UC06 - Emitir relatórios
Descrição	Permite ao administrador emitir relatórios de produtos vendidos por período, produtos mais/menos vendidos. E permite ao cliente emitir o relatório de compras realizadas por período.
Ator	<ul style="list-style-type: none"> • Administrador • Cliente

No quadro 15 apresenta-se o caso de uso “Cadastrar Fornecedores”.

Quadro 15: Descrição do caso de uso UC7

Caso de uso	UC07 - Cadastrar Fornecedores
Descrição	Permite ao administrador informar os dados de um novo fornecedor no sistema, bem como alterar ou excluir informações das categorias já cadastradas.
Ator	Administrador
Pré-condição	O Administrador deve ter efetuado o login no sistema
Fluxo principal	a) o sistema informa os fornecedores já cadastrados; b) o administrador opta por alterar, apagar ou cadastrar um novo fornecedor.
Fluxo alternativo (a)	<ul style="list-style-type: none"> • campo(s) obrigatório(s) não preenchido(s); • alerta com a mensagem “Favor preencher todos os campos obrigatórios” é apresentada.
Cenário - Edição	<ul style="list-style-type: none"> • Sistema informa os fornecedores cadastrados; • Administrador seleciona um fornecedor para edição; • Sistema mostra as informações do fornecedor selecionado; • Administrador realiza as alterações necessárias; • Administrador seleciona a opção salvar; • Sistema altera o registro e apresenta a mensagem “Fornecedor alterado com sucesso!”.
Cenário – Inclusão	<ol style="list-style-type: none"> 1. Sistema mostra a tela de cadastro de fornecedor; 2. Administrador informa todos os campos obrigatórios; 3. Administrador informa a opção cadastrar; 4. Sistema cadastra o registro e apresenta a mensagem “Fornecedor cadastrado com sucesso!”.
Cenário – Exclusão	<ol style="list-style-type: none"> 1. Sistema informa os fornecedores cadastrados; 2. Administrador seleciona um fornecedor para exclusão e clica no botão excluir ao lado do fornecedor listado; 3. Sistema exclui o registro e apresenta a mensagem “Fornecedor excluído com sucesso!”.
Pós-condição	Usuário editou, apagou ou cadastrou um fornecedor.

No quadro 16 apresenta-se o caso de uso “Cadastrar Categorias”.

Quadro 16: Descrição do caso de uso UC8

Caso de uso	UC08 - Cadastrar Categorias de produtos
Descrição	Permite ao administrador informar os dados de uma nova categoria no sistema, bem como alterar ou excluir informações das categorias já cadastradas.
Ator	Administrador
Pré-condição	<ul style="list-style-type: none"> • O Administrador deve ter efetuado o login no sistema.
Fluxo principal	a) o sistema informa as categorias já cadastradas; b) o administrador opta por alterar, apagar ou cadastrar uma nova categoria.
Fluxo alternativo (a)	<ul style="list-style-type: none"> • campo(s) obrigatório(s) não preenchido(s); • alerta com a mensagem “Favor preencher todos os campos obrigatórios” é apresentada.
Cenário - Edição	<ol style="list-style-type: none"> 1. Sistema informa as categorias cadastradas;

	<ol style="list-style-type: none"> 2. Administrador seleciona uma categoria para edição; 3. Sistema mostra as informações da categoria selecionada; 4. Administrador realiza as alterações necessárias; 5. Administrador seleciona a opção salvar; 6. Sistema altera o registro e apresenta a mensagem “Categoria alterada com sucesso!”.
Cenário – Inclusão	<ol style="list-style-type: none"> 1. Sistema mostra a tela de cadastro de categoria; 2. Administrador informa todos os campos obrigatórios; 3. Administrador informa a opção cadastrar; 4. Sistema cadastra o registro e apresenta a mensagem “Categoria cadastrada com sucesso!”.
Cenário – Exclusão	<ol style="list-style-type: none"> 1. Sistema informa as subcategorias cadastradas; 2. Administrador seleciona uma categoria para exclusão e clica no botão excluir ao lado da categoria listada; 3. Sistema exclui o registro e apresenta a mensagem “Categoria excluída com sucesso!”.
Pós-condição	Usuário editou, apagou ou cadastrou uma categoria.

No quadro 17 apresenta-se o caso de uso “Cadastrar Cliente”.

Quadro 17: Descrição do caso de uso UC09

Caso de uso	UC09 - Cadastrar cliente
Descrição	Permite ao cliente fazer o seu cadastro ou excluir a sua conta do sistema.
Ator	Cliente
Pré-condição	O Cliente deve ter efetuado o login no sistema
Fluxo principal	O cliente opta por se cadastrar no sistema ou apagar sua conta.
Fluxo alternativo (a)	<ul style="list-style-type: none"> • campo(s) obrigatório(s) não preenchido(s); • alerta com a mensagem “Favor preencher todos os campos obrigatórios” é apresentada.
Cenário – Inclusão	<ol style="list-style-type: none"> 1. Sistema mostra a tela de cadastro de usuário; 2. Cliente informa todos os campos obrigatórios; 3. Cliente informa a opção cadastrar; 4. Sistema cadastra o registro e apresenta a mensagem “Usuário cadastrado com sucesso!”.
Cenário – Exclusão	<ol style="list-style-type: none"> 1. Sistema informa dados do cliente; 2. Cliente clica em excluir conta; 3. Sistema exclui o registro e apresenta a mensagem “Usuário excluído com sucesso!”.
Pós-condição	Usuário editou, apagou ou cadastrou-se no sistema.

No quadro 18 apresenta-se o caso de uso “Adicionar itens no carrinho de compras”.

Quadro 18: Descrição do caso de uso UC10

Caso de uso	UC10 - Adicionar itens no carrinho de compras
Descrição	Permite ao cliente adicionar itens ao carrinho de compras.
Ator	Cliente
Pré-condição	Devem haver produtos cadastrados no sistema
Fluxo principal	O cliente opta por adicionar algum produto no seu carrinho de compras.
Cenário – Inclusão	<ol style="list-style-type: none"> 1. Sistema mostra a tela produtos; 2. Cliente seleciona um produto e clica em adicionar ao carrinho de compras; 3. Sistema adiciona item no carrinho e apresenta a mensagem “Produto adicionado com sucesso no carrinho de compras!”.
Pós-condição	Usuário inseriu um item ao carrinho de compras.

No quadro 19 apresenta-se o caso de uso “Alterar os dados do cadastro”.

Quadro 19: Descrição do caso de uso UC11

Caso de uso	UC11 - Alterar os dados do cadastro
Descrição	Permite ao cliente alterar os dados do seu cadastro, sendo que os dados serão todos revalidados.
Ator	Cliente
Pré-condição	O Cliente deve ter efetuado o login no sistema
Fluxo principal	O cliente opta por alterar os dados da sua conta ou apagar sua conta.
Fluxo alternativo (a)	<ul style="list-style-type: none"> • campo(s) obrigatório(s) não preenchido(s); • alerta com a mensagem “Favor preencher todos os campos obrigatórios” é apresentada.
Cenário - Edição	<ol style="list-style-type: none"> 1. Sistema informa dados do cliente; 2. Cliente realiza as alterações necessárias; 3. Cliente seleciona a opção salvar; 4. Sistema revalida todos os dados; 5. Sistema altera o registro e apresenta a mensagem “Usuário alterado com sucesso!”.
Pós-condição	Usuário editou seus dados no sistema.

No quadro 20 apresenta-se o caso de uso “Finalizar compra”.

Quadro 20: Descrição do caso de uso UC12

Caso de uso	UC12 - Finalizar compra
Descrição	Permite ao cliente finalizar a sua compra após estar devidamente cadastrado no sistema e possuir um ou mais produtos adicionados ao seu carrinho de compras.
Ator	Cliente
Pré-condição	O Cliente deve ter efetuado o login no sistema
Fluxo principal	O cliente opta por finalizar a compra após adicionar os itens no carrinho de compras.
Cenário - Inclusão	<ul style="list-style-type: none"> a) Sistema informa os produtos selecionados no carrinho para confirmação do cliente; b) Cliente realiza as alterações necessárias; c) Cliente seleciona em finalizar compra; d) Sistema confirma a forma de pagamento e o endereço de entrega; e) Sistema salva a compra e exibe a mensagem “Compra finalizada com sucesso!”.
Pós-condição	Usuário cadastrou uma compra no sistema.

No quadro 21 apresenta-se o caso de uso “Cadastrar subcategorias de produtos”.

Quadro 21: Descrição do caso de uso UC13

Caso de uso	UC13 - Cadastrar subcategorias de produtos
Descrição	Permite ao administrador informar os dados de uma nova subcategoria no sistema, bem como alterar ou excluir informações das subcategorias já cadastradas.
Ator	Administrador
Pré-condição	<ul style="list-style-type: none"> • O Administrador deve ter efetuado o login no sistema; • Devem haver uma ou mais categorias cadastradas.
Fluxo principal	<p>a) o sistema informa as subcategorias já cadastradas;</p> <p>b) o administrador opta por alterar, apagar ou cadastrar uma nova categoria.</p>
Fluxo alternativo (a)	<ul style="list-style-type: none"> • campo(s) obrigatório(s) não preenchido(s); • alerta com a mensagem “Favor preencher todos os campos obrigatórios” é apresentada.
Cenário - Edição	<ol style="list-style-type: none"> 7. Sistema informa as subcategorias cadastradas; 8. Administrador seleciona uma subcategoria para edição; 9. Sistema mostra as informações da subcategoria selecionada; 10. Administrador realiza as alterações necessárias; 11. Administrador seleciona a opção salvar; 12. Sistema altera o registro e apresenta a mensagem “Subcategoria alterada com sucesso!”.
Cenário – Inclusão	<ol style="list-style-type: none"> 5. Sistema mostra a tela de cadastro de subcategoria; 6. Administrador informa todos os campos obrigatórios; 7. Administrador informa a opção cadastrar; 8. Sistema cadastra o registro e apresenta a mensagem “Subcategoria cadastrada com sucesso!”.
Cenário – Exclusão	<ol style="list-style-type: none"> 4. Sistema informa as subcategorias cadastradas; 5. Administrador seleciona uma subcategoria para exclusão e clica no botão excluir ao lado da subcategoria listada; 6. Sistema exclui o registro e apresenta a mensagem “Subcategoria excluída com sucesso!”.
Pós-condição	Usuário editou, apagou ou cadastrou uma subcategoria.

No quadro 22 apresenta-se o caso de uso “Controlar a saída de produtos”.

Quadro 22: Descrição do caso de uso UC14

Caso de uso	UC14 - Controlar a saída de produtos
Descrição	Permite ao administrador verificar quais vendas foram realizadas, quais delas estão com seu pagamento pendente e informar quando o produto já foi devidamente enviado ao cliente.
Ator	Administrador
Pré-condição	<ul style="list-style-type: none"> • O Administrador deve ter efetuado o login no sistema; • Devem haver uma ou mais vendas realizadas.
Fluxo principal	<p>a) o sistema informa as vendas realizadas;</p> <p>b) o administrador opta por alterar o registro informando que ele já foi devidamente enviado ao cliente.</p>
Cenário - Edição	<ol style="list-style-type: none"> 1. Sistema informa as vendas realizadas; 2. Administrador seleciona uma venda para edição; 3. Sistema mostra as informações da venda selecionada; 4. Administrador realiza as alterações necessárias; 5. Administrador seleciona a opção salvar; 6. Sistema altera o registro e apresenta a mensagem “Venda alterada com sucesso!”.
Pós-condição	Usuário editou uma venda.

APÊNDICE B – Dicionário de Dados

Este Apêndice apresenta o dicionário de dados implementado para o sistema.

Quadro 23 - Tabela Pedido

A tabela PEDIDO serve para armazenar os pedidos do sistema		
Campo	Tipo	Descrição
ID	NUMERIC (14)	Identificação do pedido
DATA	DATE	Data do pedido
DATA_ATENDIMENTO	DATE	Data do atendimento
CODIGO	VARCHAR (40)	Código do pedido no PagSeguro
STATUS	NUMERIC (2)	Status do pedido no PagSeguro
DATA_ULTIMO_EVENTO	DATE	Data do ultimo evento ocorrido para este pedido
TIPO_PAGAMENTO	NUMERIC (2)	Tipo de pagamento selecionado pelo usuário
VALOR_BRUTO	NUMERIC (11,2)	Valor bruto do pedido
VALOR_DESCONTO	NUMERIC (11,2)	Valor do desconto dado para este pedido
VALOR_TAXAS	NUMERIC (11,2)	Valor das taxas cobradas para este pedido pelo PagSeguro
VALOR_LIQUIDO	NUMERIC (11,2)	Valor líquido do pedido
VALOR_EXTRA	NUMERIC (11,2)	Valor extra cobrado por algum pedido
CODIGO_RASTREIO	VARCHAR (20)	Código de rastreio do pedido. Este código é obtido pelos correios

Quadro 24 - Tabela Usuario

A tabela USUARIO serve para armazenar os usuários do sistema		
Campo	Tipo	Descrição
ID	NUMERIC (14)	Identificação do usuário
NOME	VARCHAR (50)	Nome do usuário
SOBRENOME	VARCHAR (50)	Sobrenome do usuário
NASCIMENTO	VARCHAR (10)	Nascimento do usuário
TELEFONE	VARCHAR (15)	Telefone do usuário
EMAIL	VARCHAR (50)	Email do usuário. Pode ser utilizado para efetuar <i>login</i> no sistema.
CPF	VARCHAR (15)	CPF do usuário
LOGIN	VARCHAR (20)	<i>Login</i> do usuário. Pode ser utilizado para efetuar <i>login</i> no sistema.
SENHA	VARCHAR (256)	Senha já criptografada do usuário
TIPO_USUARIO	NUMERIC (2)	Tipo de usuário. Por padrão existem dois tipos (Administrador e Cliente). Aqui é apenas um ponteiro para a tabela de TIPO_USUARIO.
ENDEREÇO	NUMERIC (11)	Endereço do usuário. Aqui é apenas um ponteiro para a tabela de pedidos.

Quadro 25 - Tabela Tipo_Usuario

A tabela TIPO_USUARIO serve para armazenar os usuários do sistema		
Campo	Tipo	Descrição
ID	NUMERIC (14)	Identificação do tipo de usuário
DESCRICAO	VARCHAR (50)	Descrição do tipo de usuário

Quadro 26 - Tabela CEP

A tabela CEP serve para armazenar os CEP cadastrados		
Campo	Tipo	Descrição
ID	NUMERIC (14)	Identificação do CEP
NUMERO	CHAR (8)	Número do CEP
DESCRICAO	VARCHAR (60)	Descrição do CEP
LOGRADOURO	NUMERIC (11)	Logradouro vinculado ao CEP
CIDADE	NUMERIC (11)	Cidade vinculada ao CEP

Quadro 27 - Tabela Endereco

A tabela ENDERECO serve para armazenar os Endereços cadastrados		
Campo	Tipo	Descrição
ID	NUMERIC (14)	Identificação do endereço
NUMERO	NUMERIC (11)	Numero do endereço
COMPLEMENTO	VARCHAR (100)	Complemento do endereço
CEP	NUMERIC (11)	CEP vinculado ao endereço

Quadro 28 - Tabela Logradouro

A tabela LOGRADOURO serve para armazenar os Logradouros cadastrados		
Campo	Tipo	Descrição
ID	NUMERIC (14)	Identificação do logradouro
TIPO	VARCHAR (15)	Tipo de logradouro (Rua, Avenida, Etc.)
DESCRICAO	VARCHAR (60)	Descrição do logradouro
BAIRRO	NUMERIC (11)	Bairro vinculado ao endereço

Quadro 29 - Tabela Fornecedor

A tabela FORNECEDOR serve para armazenar os Fornecedores cadastrados		
Campo	Tipo	Descrição
ID	NUMERIC (14)	Identificação do fornecedor
NOME	VARCHAR (100)	Nome do fornecedor
INICIO_FORNECIMENTO	VARCHAR (10)	Data de início do fornecimento
FIM_FORNECIMENTO	VARCHAR (10)	Data de fim do fornecimento
EMAIL	VARCHAR (50)	Email do fornecedor
TELEFONE	VARCHAR (15)	Telefone do fornecedor
ENDERECO	NUMERIC (11)	Endereço vinculado ao fornecedor

Quadro 30 - Tabela Cidade

A tabela CIDADE serve para armazenar as Cidades cadastradas		
Campo	Tipo	Descrição
ID	NUMERIC (14)	Identificação da cidade
CIDADE	VARCHAR (50)	Nome da cidade
ESTADO	NUMERIC (11)	Estado vinculado à cidade

Quadro 31 - Tabela Bairro

A tabela BAIRRO serve para armazenar os Bairros cadastrados		
Campo	Tipo	Descrição
ID	NUMERIC (14)	Identificação do bairro
NOME	VARCHAR (50)	Nome do bairro
CIDADE	NUMERIC (11)	Cidade vinculada ao bairro

Quadro 32 - Tabela Estado

A tabela ESTADO serve para armazenar os Estados cadastrados		
Campo	Tipo	Descrição
ID	NUMERIC (14)	Identificação do estado
NOME	VARCHAR (50)	Nome do estado
SIGLA	CHAR (2)	Sigla do estado

Quadro 33 - Tabela Genero

A tabela GENERO serve para armazenar os Gêneros cadastrados		
Campo	Tipo	Descrição
ID	NUMERIC (14)	Identificação do gênero
NOME	VARCHAR (100)	Nome do gênero
CATEGORIA	NUMERIC (14)	Categoria vinculada ao gênero

Quadro 34 - Tabela Item_Pedido

A tabela ITEM_PEDIDO serve para armazenar os Itens dos pedidos efetuados		
Campo	Tipo	Descrição
ID	NUMERIC (14)	Identificação do item
QUANTIDADE	NUMERIC (11)	Quantidade de um item do pedido
PEDIDO	NUMERIC (14)	Pedido vinculado ao item
PRODUTO	NUMERIC (14)	Produto vinculado ao item

Quadro 35 - Tabela Produto

A tabela PRODUTO serve para armazenar os Produtos cadastrados		
Campo	Tipo	Descrição
ID	NUMERIC (14)	Identificação do produto
NOME	VARCHAR (50)	Nome do produto
DESCRICA0	VARCHAR (1000)	Descrição do produto
ENTRADA_ESTOQUE	VARCHAR (10)	Data de entrada ao estoque
QUANTIDADE	NUMERIC (11)	Quantidade do produto
ESTOQUE_MINIMO	NUMERIC (11)	Quantidade mínima do produto
PRECO_COMPRA	NUMERIC (11,2)	Preço de compra
PRECO_VENDA	NUMERIC (11,2)	Preço de venda
EM_PROMOCAO	NUMERIC (1)	Indica se o produto se encontra em promoção
PORCENTAGEM_PROMOCAO	NUMERIC (11)	Percentual da promoção
EM_DESTAQUE	NUMERIC (1)	Indica se o produto está em destaque
IMAGEM	VARCHAR (150)	Imagem do produto
SUB_CATEGORIA	NUMERIC (14)	Subcategoria vinculada ao produto
FORNECEDOR	NUMERIC (14)	Fornecedor vinculado ao produto
GENERO	NUMERIC (14)	Gênero vinculado ao produto

Quadro 36 - Tabela Categoria

A tabela CATEGORIA serve para armazenar as Categorias cadastradas		
Campo	Tipo	Descrição
ID	NUMERIC (14)	Identificação da categoria
NOME	VARCHAR (100)	Nome da categoria
DESCRICAÇÃO	VARCHAR (150)	Descrição da categoria
EM_PROMOCAO	NUMERIC (1)	Indica se a categoria está em promoção
PORCENTAGEM_PROMOCAO	NUMERIC (11)	Percentual de promoção
EM_DESTAQUE	NUMERIC (1)	Indica se a categoria está em destaque

Quadro 37 - Tabela Sub_Categoria

A tabela SUB_CATEGORIA serve para armazenar as Subcategorias cadastradas		
Campo	Tipo	Descrição
ID	NUMERIC (14)	Identificação da Subcategoria
NOME	VARCHAR (100)	Nome da Subcategoria
DESCRICAÇÃO	VARCHAR (150)	Descrição da Subcategoria
EM_PROMOCAO	NUMERIC (1)	Indica se a Subcategoria está em promoção
PORCENTAGEM_PROMOCAO	NUMERIC (11)	Percentual de promoção
EM_DESTAQUE	NUMERIC (1)	Indica se a Subcategoria está em destaque

Quadro 38 - Tabela Parametros

A tabela PARAMETROS serve para armazenar os Parâmetros cadastrados		
Campo	Tipo	Descrição
ID	NUMERIC (14)	Identificação o parâmetro
EMAIL_ADMINISTRADOR	VARCHAR(100)	Email do administrador
NOME_LOJA	VARCHAR(100)	Nome da loja
IMAGEM_LOJA	VARCHAR(150)	Imagem da loja