

**INSTITUTO FEDERAL DE EDUCAÇÃO, CIÊNCIAS E TECNOLOGIA DE
RONDÔNIA – CAMPUS JI-PARANÁ
CURSO SUPERIOR DE TECNOLOGIA EM ANÁLISE E DESENVOLVIMENTO DE
SISTEMAS**

**DESENVOLVIMENTO DE UM SISTEMA E-COMMERCE DE VENDA
DE CUPONS: UMA VISÃO A PARTIR DO BACK-END**

TIAGO RAFAEL OLIVEIRA DAS VIRGENS

Ji-Paraná – RO

2025

TIAGO RAFAEL OLIVEIRA DAS VIRGENS

**DESENVOLVIMENTO DE UM SISTEMA E-COMMERCE DE VENDA
DE CUPONS: UMA VISÃO A PARTIR DO BACK-END**

Monografia apresentada ao Instituto Federal de Rondônia - Campus Ji-Paraná, como requisito para obtenção do título de Tecnólogo em Análise e Desenvolvimento de Sistemas.

Área de Concentração: Ciências Exatas e da Terra.

Orientador (a): Prof. Dr. Jackson Henrique da Silva Bezerra

Ji-Paraná – RO

2025

FICHA CATALOGRÁFICA

Ficha catalográfica elaborada pelo Sistema Gerador de Ficha Catalográfica do IFRO.

Virgens, Tiago Rafael Oliveira das.
Desenvolvimento de um sistema e-commerce de venda de
cupons: uma visão a partir do back-end / Tiago Rafael Oliveira das
Virgens. - Ji-Paraná, 2025.
81 f. : il.

Orientador(a): Jackson Henrique da Silva Bezerra.

Trabalho de Conclusão de Curso (Superior de Tecnologia em
Análise e Desenvolvimento de Sistemas) – Instituto Federal de
Educação, Ciência e Tecnologia de Rondônia - IFRO, Ji-Paraná,
2025.

1. E-commerce. 2. Florianópolis. 3. Comércio local. I. Bezerra,
Jackson Henrique da Silva (orient.). II. Instituto Federal de Educação,
Ciência e Tecnologia de Rondônia - IFRO. III. Título.

Bibliotecário(a) Responsável: Roseni Santos Rodrigues, CRB-11/916

BANCA EXAMINADORA

Dr. Gleison Guardia

Avaliador n° 1

Dr. Clayton Ferraz Andrade

Avaliador n° 2

Dr. Jackson Henrique da Silva Bezerra

Presidente da Banca Examinadora

Processo SEI n°: _____

Data da Defesa: 20 de Agosto de 2025.

Nota Final: 95

RESUMO

Este trabalho apresenta o desenvolvimento do back-end de um sistema de e-commerce voltado à venda de cupons promocionais, com foco inicial na cidade de Florianópolis (SC) e acessibilidade nacional. A proposta surgiu da necessidade de fortalecer a visibilidade do comércio local em uma região turística, mas ainda carente de soluções digitais eficientes para a promoção de ofertas regionais. O principal objetivo do projeto é criar uma plataforma que conecte consumidores e estabelecimentos, permitindo a oferta de cupons em categorias como gastronomia, hospedagem e entretenimento, com foco na usabilidade, especialmente para pessoas de meia-idade e da terceira idade. A metodologia utilizada combina o estudo de caso, que permitiu compreender as particularidades do contexto local e as demandas dos usuários, com a aplicação do framework ágil Scrum, que guiou o processo de desenvolvimento em sprints de quatro semanas. Foram aplicados princípios da engenharia de software, utilizando o framework NestJS para a construção da API RESTful, integrada ao banco de dados MySQL. Também foram utilizadas APIs externas, como ViaCEP para consulta de endereços e Mercado Pago para processamento de pagamentos. O desenvolvimento seguiu práticas modernas de segurança, organização de rotas, autenticação de usuários e geração automatizada de documentação via Swagger. Os resultados alcançados incluem a implementação de funcionalidades essenciais, como cadastro de usuários, emissão de vouchers, controle de transações e integração com serviços externos. A estrutura modular da API garante escalabilidade e facilita futuras expansões. Conclui-se que o sistema desenvolvido tem potencial para melhorar o acesso da população a promoções locais, contribuir com o fortalecimento do comércio de Florianópolis e servir como base para soluções similares em outras regiões do país.

Palavras-chave: e-commerce, Florianópolis, comércio local.

ABSTRACT

This paper presents the development of the backend of an e-commerce system focused on the sale of promotional coupons, with an initial focus on the city of Florianópolis (SC) and nationwide accessibility. The proposal arose from the need to strengthen the visibility of local businesses in a tourist region, but still lacking efficient digital solutions for promoting regional offers. The main objective of the project is to create a platform that connects consumers and establishments, allowing the offering of coupons in categories such as gastronomy, lodging and entertainment, with a focus on usability, especially for middle-aged and elderly people. The methodology adopted involved the application of software engineering principles, using the NestJS framework to build the RESTful API, integrated with the MySQL database. External APIs were also used, such as ViaCEP for address queries and Mercado Pago for payment processing. The development followed modern security practices, route organization, user authentication and automated generation of documentation via Swagger. The results achieved include the implementation of essential functionalities, such as user registration, voucher issuance, transaction control and integration with external services. The modular structure of the API ensures scalability and facilitates future expansions. It is concluded that the system developed has the potential to improve the population's access to local promotions, contribute to strengthening commerce in Florianópolis and serves as a basis for similar solutions in other regions of the country.

Keywords: e-commerce, Florianópolis, local businesses.

LISTA DE FIGURAS

Figura 1 – Exemplo do site Laçador de Ofertas	18
Figura 2 - Processo Scrum.....	26
Figura 3 – Rota de cep	33
Figura 4 – Rota de cnpj	34
Figura 5 – Rota de cnpj	34
Figura 6 - Rota de solicitar redefinição de senha	35
Figura 7 - Rota de solicitar redefinição de senha	35
Figura 8 - Rota para alterar senha.....	36
Figura 9 - Rota para buscar informações do usuário autenticado.....	36
Figura 10 - Rota para cadastrar um cliente.....	37
Figura 11 - Rota para listar os clientes	37
Figura 12 - Rota para buscar informações detalhadas de um cliente	38
Figura 13 - Rota para atualizar um cliente	38
Figura 14 - Rota para excluir um cliente	39
Figura 15 - Rota para gerar um relatório de clientes	39
Figura 16 - Rota para cadastrar uma categoria	40
Figura 17 - Rota para listar todas as categorias	40
Figura 18 - Rota para buscar informações detalhadas de uma categoria.....	41
Figura 19 - Rota para atualizar uma categoria.....	41
Figura 20 - Rota para excluir uma categoria	42
Figura 21 - Rota para gerar um relatório de categorias	42
Figura 22 - Rota para cadastrar um parceiro	43
Figura 23 - Rota para listar todos os parceiros	43
Figura 24 - Rota para listar todas os parceiros inativos.....	44
Figura 25 - Rota para buscar informações detalhadas de um parceiro	44
Figura 26 - Rota para atualizar um parceiro	45
Figura 27 - Rota para excluir um parceiro.....	45
Figura 28 - Rota para gerar um relatório de parceiros.....	46
Figura 29 - Rota para ativar um parceiro.....	46
Figura 30 - Rota para não aceitar o parceiro	47
Figura 31 - Rota para gerar o dashboard de parceiro	48
Figura 32 - Rota para cadastrar um administrador	48

Figura 33 - Rota para gerar o dashboard de administrador	49
Figura 34 - Rota para listar todas os estados	49
Figura 35 - Rota para buscar informações detalhadas de um estado.....	50
Figura 36 - Rota para listar todas as cidades	50
Figura 37 - Rota para buscar informações detalhadas de uma cidade.....	51
Figura 38 - Rota para listar cidades de um estado.....	51
Figura 39 - Rota para cadastrar um endereço	52
Figura 40 - Rota para listar todas os endereços	52
Figura 41 - Rota para buscar informações detalhadas de um endereço.....	53
Figura 42 - Rota para atualizar um endereço.....	53
Figura 43 - Rota para excluir um endereço	54
Figura 44 - Rota para cadastrar um cupom.....	54
Figura 45 - Rota para listar todas os cupons.....	55
Figura 46 - Rota para adicionar imagem a um cupom	55
Figura 47 - Rota para listar o menu de cupons filtrados por categorias	56
Figura 48 - Rota para listar os cupons inativos	56
Figura 49 - Rota para buscar informações detalhadas de um cupom	57
Figura 50 - Rota para atualizar um cupom	57
Figura 51 - Rota para excluir um cupom.....	58
Figura 52 - Rota para retornar todos os cupons de um parceiro.....	58
Figura 53 - Rota para aceitar a solicitação e ativar um cupom.....	59
Figura 54 - Rota para negar a solicitação e desativar o cupom	59
Figura 55 - Rota para gerar um relatório de cupons	60
Figura 56 - Rota para cadastrar uma agenda	60
Figura 57 - Rota para listar todas as agendas	61
Figura 58 - Rota para buscar informações detalhadas de uma agenda.....	61
Figura 59 - Rota para atualizar uma agenda.....	62
Figura 60 - Rota para excluir uma agenda.....	62
Figura 61 - Rota para gerar um relatório de agendas	63
Figura 62 - Rota para cadastrar um favorito.....	63
Figura 63 - Rota para retornar todos os favoritos por cliente.....	64
Figura 64 - Rota para excluir uma agenda.....	64
Figura 65 - Rota para gerar um relatório de favoritos	65
Figura 66 - Rota para adicionar cupom no carrinho.....	65

Figura 67 - Rota para listar carrinho do cliente	66
Figura 68 - Rota para atualizar quantidade de um cupom no carrinho.....	66
Figura 69 - Rota para remover um cupom do carrinho	67
Figura 70 - Rota para cadastrar uma venda	67
Figura 71 - Rota para listar todas as vendas	68
Figura 72 - Rota para compra rápida de um cupom	68
Figura 73 - Rota para gerar um relatório de vendas	69
Figura 74 - Rota para buscar venda pelo id do carrinho.....	69
Figura 75 - Rota para buscar informações da venda	70
Figura 76 - Rota para cadastrar um recebimento.....	70
Figura 77 - Rota para listar todos os recebimentos	71
Figura 78 - Rota para buscar informações detalhadas de um recebimento	71
Figura 79 - Rota para gerar um relatório de recebimentos	72
Figura 80 - Rota para gerar link de pagamento	72
Figura 81 - Rota para consultar uma ordem de compra	73
Figura 82 - Rota para consultar detalhes do pagamento.....	73
Figura 83 - Rota para consultar detalhes do pagamento.....	74
Figura 84 - Rota para receber notificações de pagamento pelo webhook.....	74
Figura 85 - Rota para gerar um relatório de voucher	75
Figura 86 - Rota para gerar um voucher para a venda.....	75
Figura 87 - Rota para listar todos os vouchers	76
Figura 88 - Rota para buscar informações detalhadas de um voucher	76
Figura 89 - Rota para validar o voucher	77

LISTA DE TABELAS

Tabela 1 - Sprints.....	29
-------------------------	----

LISTA DE ABREVIATURAS E SIGLAS

API – Application Programming Interface

SOA – Service-Oriented Architecture

CRM – Customer Relationship Management

URL – Uniform Resource Locator

EDI – Electronic Data Interchange

ORM – Object-Relational Mapping

ACID – Atomicidade, Consistência, Isolamento e Durabilidade

CEP – Código de Endereçamento Postal

HTTP – Hypertext Transfer Protocol

SI – Sistemas de Informação

SDK – Software Development Kit

CORS – Cross-Origin Resource Sharing

SUMÁRIO

1. INTRODUÇÃO	16
1.1. PROBLEMÁTICA	17
1.2. JUSTIFICATIVA	18
1.3. OBJETIVOS	18
1.3.1. Objetivo Geral	18
1.3.2. Objetivos Específicos	19
1.4. ESTRUTURA DO TRABALHO	19
2. REFERENCIAL TEÓRICO	21
2.1. SISTEMAS WEB	21
2.1.1. E-commerce	21
2.1.2. Marketing Digital	22
2.2. TECNOLOGIAS UTILIZADAS NO DESENVOLVIMENTO	23
2.2.1. Nestjs	24
2.2.2. Mysql	24
2.2.3. API ViaCep	24
2.2.4. API Mercado Pago	25
3. METODOLOGIA	28
3.1. METODOLOGIA DE PESQUISA	28
3.2. METODOLOGIA DE DESENVOLVIMENTO DE SOFTWARE	28
3.3. PROCESSO DE INTEGRAÇÃO COM O FRONT-END	30
4. RESULTADOS E DISCUSSÕES	32
4.1. SWAGGER	32
4.2. ROTAS GERAIS	33
4.2.1. Rota de cep	33
4.2.2. Rota de cnpj	33
4.3. ROTAS DE AUTENTICAÇÃO	34
4.3.1. Rota de login	34
4.3.2. Rota de solicitar redefinição de senha	35
4.3.3. Rota de redefinição da senha	35
4.3.4. Rota para alterar senha	36
4.3.5. Rota para obter informações do usuário autenticado	36
4.4. ROTAS DE CLIENTE	37
4.4.1. Rota para cadastrar um cliente	37
4.4.2. Rota para listar todos os clientes	37
4.4.3. Rota para buscar informações detalhadas de um cliente	38
4.4.4. Rota para atualizar um cliente	38

4.4.5.	Rota para excluir um cliente.....	39
4.4.6.	Rota para gerar um relatório de clientes.....	39
4.5.	ROTAS DE CATEGORIA	40
4.5.1.	Rota para cadastrar uma categoria.....	40
4.5.2.	Rota para listar todas as categorias.....	40
4.5.3.	Rota para buscar informações detalhadas de uma categoria	41
4.5.4.	Rota para atualizar uma categoria	41
4.5.5.	Rota para excluir uma categoria	42
4.5.6.	Rota para gerar um relatório de categorias.....	42
4.6.	ROTAS DE PARCEIRO	43
4.6.1.	Rota para cadastrar um parceiro	43
4.6.2.	Rota para listar todos os parceiros.....	43
4.6.3.	Rota para listar todas os parceiros inativos	44
4.6.4.	Rota para buscar informações detalhadas de um parceiro.....	44
4.6.5.	Rota para atualizar um parceiro.....	45
4.6.6.	Rota para excluir um parceiro	45
4.6.7.	Rota para gerar um relatório de parceiros	46
4.6.8.	Rota para ativar um parceiro	46
4.6.9.	Rota para não aceitar o parceiro	47
4.6.10.	Rota para gerar o dashboard de parceiro	47
4.7.	ROTAS DE ADMIN	48
4.7.1.	Rota para cadastrar um administrador.....	48
4.7.2.	Rota para gerar o dashboard de administrador	49
4.8.	ROTAS DE ESTADO.....	49
4.8.1.	Rota para listar todas os estados	49
4.8.2.	Rota para buscar informações detalhadas de um estado.....	50
4.9.	ROTAS DE CIDADE	50
4.9.1.	Rota para listar todas as cidades	50
4.9.2.	Rota para buscar informações detalhadas de uma cidade	51
4.9.3.	Rota para listar cidades de um estado.....	51
4.10.	ROTAS DE ENDEREÇO	52
4.10.1.	Rota para cadastrar um endereço.....	52
4.10.2.	Rota para listar todas os endereços.....	52
4.10.3.	Rota para buscar informações detalhadas de um endereço	53
4.10.4.	Rota para atualizar um endereço	53
4.10.5.	Rota para excluir um endereço	54
4.11.	ROTAS DE ENDEREÇO	54
4.11.1.	Rota para cadastrar um cupom	54
4.11.2.	Rota para listar todas os cupons	55
4.11.3.	Rota para adicionar imagem a um cupom	55
4.11.4.	Rota para listar o menu de cupons filtrados por categorias.....	56
4.11.5.	Rota para listar os cupons inativos	56

4.11.6.	Rota para buscar informações detalhadas de um cupom.....	57
4.11.7.	Rota para atualizar um cupom.....	57
4.11.8.	Rota para excluir um cupom.....	58
4.11.9.	Rota para retornar todos os cupons de um parceiro.....	58
4.11.10.	Rota para aceitar a solicitação e ativar um cupom.....	59
4.11.11.	Rota para negar a solicitação e desativar o cupom	59
4.11.12.	Rota para gerar um relatório de cupons	60
4.12.	ROTAS DE AGENDA.....	60
4.12.1.	Rota para cadastrar uma agenda	60
4.12.2.	Rota para listar todas as agendas	61
4.12.3.	Rota para buscar informações detalhadas de uma agenda.....	61
4.12.4.	Rota para atualizar uma agenda.....	62
4.12.5.	Rota para excluir uma agenda	62
4.12.6.	Rota para gerar um relatório de agendas	63
4.13.	ROTAS DE FAVORITO.....	63
4.13.1.	Rota para cadastrar um favorito.....	63
4.13.2.	Rota para retornar todos os favoritos por cliente.....	64
4.13.3.	Rota para excluir uma agenda	64
4.13.4.	Rota para gerar um relatório de favoritos	65
4.14.	ROTAS DE CARRINHO	65
4.14.1.	Rota para adicionar cupom no carrinho.....	65
4.14.2.	Rota para listar carrinho do cliente.....	66
4.14.3.	Rota para atualizar quantidade de um cupom no carrinho	66
4.14.4.	Rota para remover um cupom do carrinho	67
4.15.	ROTAS DE VENDA	67
4.15.1.	Rota para cadastrar uma venda.....	67
4.15.2.	Rota para listar todas as vendas	68
4.15.3.	Rota para compra rápida de um cupom	68
4.15.4.	Rota para gerar um relatório de vendas	69
4.15.5.	Rota para buscar venda pelo id do carrinho	69
4.15.6.	Rota para buscar informações da venda	70
4.16.	ROTAS DE RECEBIMENTO.....	70
4.16.1.	Rota para cadastrar um recebimento	70
4.16.2.	Rota para listar todos os recebimentos	71
4.16.3.	Rota para buscar informações detalhadas de um recebimento	71
4.16.4.	Rota para gerar um relatório de recebimentos.....	72
4.17.	ROTAS DE PAGAMENTO (Mercado Pago)	72
4.17.1.	Rota para gerar link de pagamento	72
4.17.2.	Rota para consultar uma ordem de compra	73
4.17.3.	Rota para consultar detalhes do pagamento	73
4.17.4.	Rota para consultar detalhes do pagamento	74
4.17.5.	Rota para receber notificações de pagamento pelo webhook.....	74

4.18. ROTAS DE VOUCHER.....	75
4.18.1. Rota para gerar um relatório de voucher	75
4.18.2. Rota para gerar um voucher para a venda	75
4.18.3. Rota para listar todos os vouchers	76
4.18.4. Rota para buscar informações detalhadas de um voucher	76
4.18.5. Rota para validar o voucher	77
5. CONCLUSÃO	79
6. REFERÊNCIAS	80
APÊNDICE A: Projeto do Software (Obrigatório).....	82
7. ANEXO I: Título do Anexo (Opcional).....	83

1. INTRODUÇÃO

A transformação digital tem promovido mudanças significativas na forma como consumidores se relacionam com empresas e serviços. Nesse cenário, os sistemas de e-commerce se destacam como ferramentas estratégicas para impulsionar o comércio local, ampliar o alcance de negócios e oferecer comodidade aos usuários (SEBRAE, 2024).

De fato, o e-commerce brasileiro manteve sua trajetória de crescimento, alcançando R\$ 196,1 bilhões em vendas em 2023, um aumento de aproximadamente 4,8% em relação a 2022, quando o setor movimentou R\$ 187,9 bilhões (NIELSENIQ, 2024; AGÊNCIA GOV, 2024; PODER360, 2024). Esse dado reforça o papel cada vez mais relevante do comércio eletrônico no país e evidencia como plataformas digitais se consolidam como canais essenciais de vendas. Além disso, o setor representou cerca de 9% do total de vendas do varejo brasileiro em 2023 (GOIÁS.GOV, 2024), demonstrando seu impacto no mercado e na economia local.

O e-commerce também favorece pequenos e médios negócios: micro e pequenas empresas movimentaram cerca de R\$ 945 milhões nos primeiros três meses de 2024, crescimento de 34% em relação ao mesmo período de 2023. Esses dados reforçam a importância de plataformas digitais acessíveis, como a desenvolvida neste trabalho, para aumentar a competitividade e a visibilidade de negócios locais em Florianópolis.

Florianópolis, capital do estado de Santa Catarina, é reconhecida por seu potencial turístico, recebendo visitantes de todas as regiões do Brasil ao longo do ano. No entanto, apesar de sua vocação turística e diversidade de estabelecimentos comerciais, a cidade ainda carecia de plataformas acessíveis e eficazes voltadas para a promoção de ofertas locais por meio de cupons de desconto.

Diante desse cenário, este trabalho teve como foco o desenvolvimento do back-end de um sistema de e-commerce destinado à comercialização de cupons promocionais, com implementação inicial em Florianópolis, mas projetado para operar em nível nacional. A plataforma foi voltada, principalmente, a restaurantes, bares e pousadas da região, que puderam utilizar os cupons como estratégia para atrair novos clientes e aumentar a visibilidade de seus serviços. O sistema buscou fomentar o crescimento do comércio local, promovendo uma maior circulação de consumidores e contribuindo para o fortalecimento da economia regional.

O sistema também considerou a importância de uma experiência de usuário intuitiva e inclusiva, com design acessível e usabilidade adaptada ao público de meia-idade e da terceira idade grupo que, muitas vezes, encontra dificuldades ao utilizar sistemas digitais. Assim, a plataforma apresentou uma interface agradável, com uma paleta de cores e tipografia pensadas

para esse público, facilitando a navegação mesmo para aqueles com menor familiaridade tecnológica.

Além de atender à demanda local, o sistema demonstrou potencial para alcançar usuários de todo o Brasil, especialmente turistas que buscam ofertas antes de visitar a cidade. Como diferencial, a plataforma foi desenvolvida com foco na segurança de dados e contou com funcionalidades como autenticação de usuários, integração com serviços de pagamento e categorização de cupons por tipo de serviço (como hospedagem, alimentação e lazer).

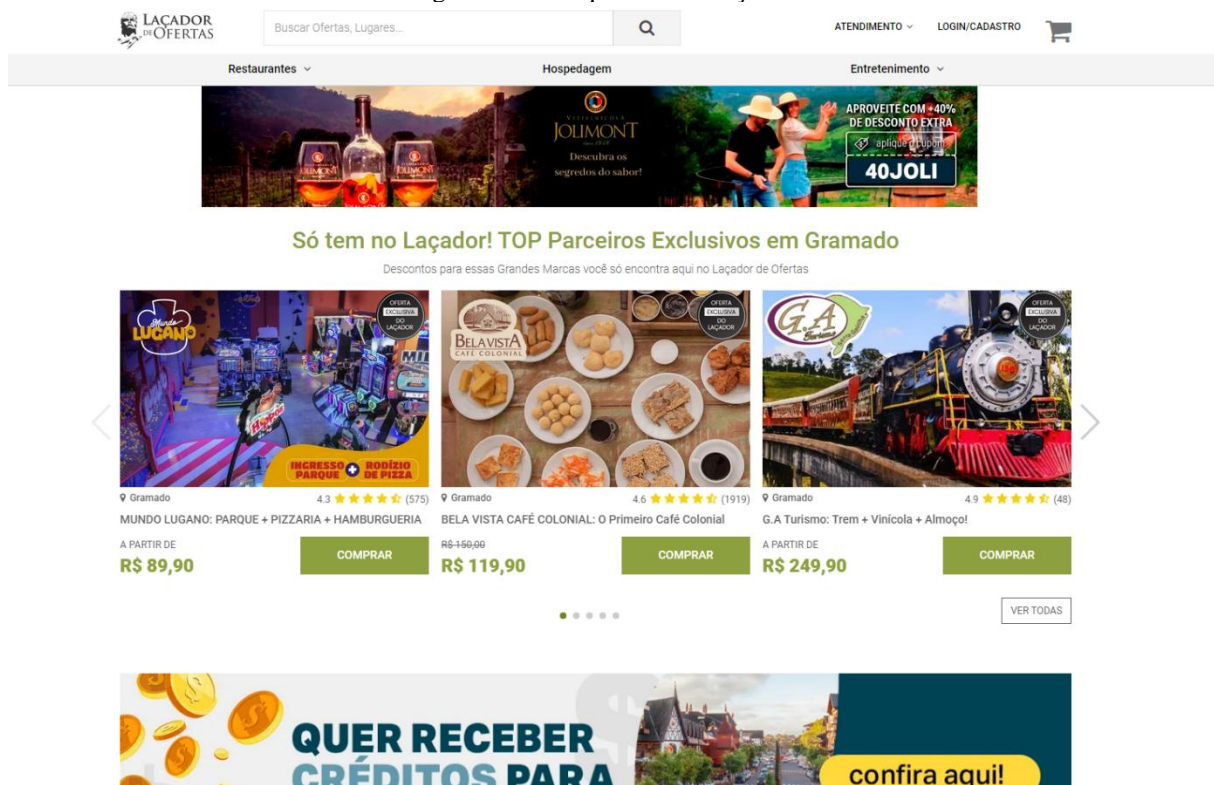
Dessa forma, este projeto atendeu a uma carência observada no mercado local de Florianópolis e ofereceu uma solução tecnológica escalável e segura, capaz de conectar consumidores a oportunidades exclusivas, ao mesmo tempo em que contribuiu para a inovação no setor de vendas promocionais por meio da internet.

1.1. PROBLEMÁTICA

Atualmente, há uma carência de plataformas de e-commerce eficientes e acessíveis que permitam a popularização das vendas locais por meio de cupons promocionais. Embora existam soluções de cupons disponíveis no Brasil, o mercado de Florianópolis demanda uma atenção especial, devido à sua forte vocação turística e à diversidade de atrações locais. O sistema desenvolvido busca justamente atender essa demanda, facilitando o acesso dos comerciantes a um público mais amplo e potencializando as oportunidades de negócios na cidade.

O sistema oferece uma interface visual agradável e intuitiva, com uma paleta de cores e fontes acessíveis, pensadas especialmente para o público-alvo, que inclui pessoas de meia-idade e da terceira idade. Essa abordagem visa garantir que a navegação seja simples e confortável, facilitando o uso do sistema por pessoas com diferentes níveis de familiaridade com tecnologia. Como referência, pode-se observar um site já existente de venda de cupons na cidade de Gramados, conforme demonstrado na Figura 1 a seguir.

Figura 1 – Exemplo do site Laçador de Ofertas



Fonte: Site Laçador de Ofertas (2024). Disponível em: <https://www.lacadordeofertas.com.br/>

1.2. JUSTIFICATIVA

Um sistema de cupons de e-commerce para Florianópolis é uma contribuição à crescente demanda por soluções tecnológicas que colocam consumidores e empresas locais em geral em uma posição vantajosa. Ao se tornar uma cidade de turismo com vários itens em seu mercado, como este, Florianópolis ainda carece de plataformas adequadas que coloquem varejistas locais diante dos clientes de uma forma inovadora e acessível.

Além disso com o turismo elevado na cidade a plataforma tem potencial para atender o Brasil inteiro, caso as pessoas busquem por viagens para Florianópolis. Como diferencial o sistema traz um sistema seguro que não irá deixar os dados dos usuários de forma aberta e várias opções para os clientes, podendo se expandir por categorias de entretenimento, hospedagens, entre outros.

1.3. OBJETIVOS

1.3.1. Objetivo Geral

Desenvolver o back-end de um sistema de *e-commerce* para a venda de cupons, com foco inicial em Florianópolis e acesso nacional, proporcionando uma plataforma eficiente e atrativa para consumidores locais e de todo o Brasil.

1.3.2. Objetivos Específicos

- Analisar e projetar o sistema, visando a regra de negócios e características específicas dos e-commerces de vendas, incluindo o levantamento de requisitos, análise de segurança, escalabilidade e usabilidade.
- Desenvolver e testar o back-end, implementando o back-end com o NestJS, integrado ao banco de dados MySQL, com foco na eficiência, segurança, performance, integração com serviços de pagamento, autenticação de usuários e gerenciamento dos cupons.
- Criar uma API Restful que permita comunicar o back-end com o front-end, realizar cadastros, realizar consultas, emitir relatórios, entre outros.
- Documentar o processo de desenvolvimento da API, afim de facilitar a integração com o front-end.

1.4. ESTRUTURA DO TRABALHO

A partir desta Introdução, o trabalho está organizado em seis seções principais. A primeira seção, denominada Problemática, apresenta o contexto que motivou o desenvolvimento do sistema, destacando as dificuldades enfrentadas por consumidores e estabelecimentos na divulgação e aquisição de cupons promocionais. Explica-se como a ausência de uma solução tecnológica integrada dificulta a distribuição eficiente desses cupons. Na seção Justificativa, são discutidas as razões que sustentam a realização do projeto, ressaltando a importância de criar uma plataforma acessível, automatizada e segura para potencializar a divulgação de promoções e estimular o comércio local, com foco inicial em Florianópolis. Em Objetivos, são definidos o objetivo geral, o desenvolvimento do back-end de um sistema de venda de cupons e os objetivos específicos, que orientaram o processo de desenvolvimento e validação da aplicação. A seção Referencial Teórico reúne conceitos fundamentais para o entendimento do projeto, como sistemas web, e-commerce, marketing digital e transformação digital. Também são descritas as principais tecnologias utilizadas, como o framework NestJS, o banco de dados MySQL e as APIs ViaCEP e Mercado Pago, que foram integradas ao sistema para automatizar processos essenciais. Em Metodologia, são apresentadas

as abordagens utilizadas tanto para a pesquisa quanto para o desenvolvimento do software. Descrevem-se as etapas do levantamento de requisitos, definição da arquitetura da aplicação, implementação das funcionalidades e realização de testes. Na seção Resultados e Discussões, são expostos os principais resultados obtidos com o desenvolvimento da aplicação. São analisados aspectos como a estrutura da API, a integração com serviços externos, a organização do código e a documentação gerada, destacando os benefícios da solução proposta. Por fim, a Conclusão retoma os objetivos alcançados, discute a contribuição prática e acadêmica do trabalho e aponta direções para futuras melhorias, incluindo a implementação completa do sistema com parcerias reais na cidade de Florianópolis.

2. REFERENCIAL TEÓRICO

2.1. SISTEMAS WEB

Com o avanço da digitalização e conectividade, muitas empresas estão investindo em soluções de software para otimizar processos e aumentar a produtividade. Os sistemas web se destacam como uma alternativa eficaz e econômica para atender a essas necessidades. Um sistema web é um software acessível através da internet, permitindo que usuários de qualquer lugar e a partir de qualquer dispositivo com navegador e conexão possam interagir com a aplicação. Essa acessibilidade elimina a necessidade de instalação local e simplifica o processo de atualização e manutenção, que ocorre diretamente no servidor (CUNHA, 2024).

O funcionamento de um sistema web envolve a hospedagem do software em um servidor, que processa as solicitações dos usuários e gerencia o banco de dados. Desenvolvidos em linguagens web como JavaScript, esses sistemas podem ser acessados via desktops, notebooks, tablets e smartphones. A evolução dos sistemas web transformou a maneira como armazenamos e compartilhamos dados. Antigamente, o armazenamento local limitava a eficiência do compartilhamento de arquivos, mas com a melhoria da banda larga e a popularização da internet, o compartilhamento e a gestão de dados se tornaram mais rápidos e simplificados (CUNHA, 2024).

Entre as principais vantagens dos sistemas web estão a maior produtividade, pelo acesso remoto e resolução rápida de problemas; o custo-benefício, evitando a compra de hardware específico; a facilidade de atualizações e melhorias; a integração eficiente entre diferentes processos; a segurança com backups automáticos e controle de acesso; e a personalização, que permite adaptar o software às necessidades específicas da empresa. Portanto, os sistemas web são uma solução atraente para empresas que buscam flexibilidade, eficiência e inovação em suas operações digitais (CUNHA, 2024).

2.1.1. E-commerce

A internet revolucionou a maneira como as pessoas fazem compras, dando origem ao e-commerce, termo que deriva de "*electronic commerce*", ou comércio eletrônico em português. Esse tipo de comércio começou a ganhar relevância nos anos 1960, quando algumas empresas começaram a usar o *Electronic Data Interchange* (EDI) para trocar documentos eletronicamente, marcando o início do e-commerce (MATA, 2021).

Com a expansão do uso de smartphones, uma grande parte dos consumidores passou a realizar suas pesquisas de compra exclusivamente online, com cerca de 53% dos usuários utilizando apenas a internet para isso, enquanto 37% alternam entre lojas físicas e o ambiente virtual (MENDONÇA, 2016). Diante desse cenário, as empresas perceberam a oportunidade de ampliar seu alcance e reduzir custos operacionais, ao mesmo tempo em que oferecem uma experiência de compra mais personalizada e conveniente. Assim, o comércio eletrônico firmou-se como uma prática indispensável, permitindo que os consumidores realizem suas compras a qualquer momento e em qualquer lugar, o que tem impulsionado o crescimento constante desse setor.

2.1.2. Marketing Digital

O Marketing Digital tornou-se uma ferramenta indispensável para microempresas e pequenas empresas que desejam se destacar na era digital. Com a crescente presença da população mundial online, seja em redes sociais, sites de compras ou ferramentas de comunicação como e-mail e WhatsApp, é crucial que as empresas estejam visíveis e acessíveis no ambiente digital (SEBRAE, 2024).

Marketing Digital é o processo de promoção de produtos e marcas através de canais online, como blogs, sites, redes sociais, e-mails e aplicativos. Esta abordagem permite uma comunicação direta e personalizada com os consumidores, atendendo às suas necessidades e desejos no momento oportuno. A eficácia do Marketing Digital está na sua capacidade de medir o desempenho das ações promocionais, avaliar o retorno sobre o investimento e fornecer métricas claras para otimizar as decisões estratégicas. Isso é fundamental para expandir negócios, aumentar vendas e fortalecer a marca (SEBRAE, 2024).

Entre as principais vantagens do Marketing Digital estão a interatividade e o engajamento com o público, possibilitando uma comunicação mais próxima e relacional (MO4 WEB, 2024; HARVESTO, 2024). Além disso, a capacidade de mensurar resultados em tempo real permite o monitoramento e ajustes rápidos, otimizando o retorno sobre investimento (MO4 WEB, 2024; FOCUSKEYWORDFINDER, 2023). A internet também oferece alcance global e segmentação precisa, permitindo que pequenas empresas atinjam públicos específicos em diferentes regiões e melhorem a eficácia das campanhas (MO4 WEB, 2024; MZCLICK, 2024). Outro ponto importante é o funcionamento contínuo, que garante contato com clientes 24 horas por dia, 7 dias por semana, fortalecendo a fidelização e a autoridade da marca (HARVESTO, 2024; MZCLICK, 2024).

Para implementar o Marketing Digital de forma eficaz, é necessário um planejamento estruturado. Esse planejamento deve começar com uma análise da situação atual da empresa, incluindo o mercado, a concorrência e a presença online. É importante definir metas claras e mensuráveis para as campanhas, criar perfis detalhados do público-alvo, e desenvolver um plano de conteúdo e mídia relevante. Além disso, é essencial estabelecer indicadores de desempenho e métodos de avaliação de sucesso, além de manter flexibilidade para adaptar as estratégias conforme os resultados e as mudanças de mercado (SEBRAE, 2024).

O Marketing Digital, portanto, não é apenas uma estratégia de promoção, mas uma ferramenta abrangente que permite às empresas alcançar seus objetivos, expandir sua presença e melhorar continuamente suas operações no ambiente digital (SEBRAE, 2024).

2.2. TECNOLOGIAS UTILIZADAS NO DESENVOLVIMENTO

Para o back-end, foi utilizado o NestJS, um framework que adota uma estrutura modular e facilita o desenvolvimento de aplicações server-side em TypeScript. O TypeScript é uma linguagem que adiciona tipagem estática ao JavaScript, proporcionando maior segurança e manutenção do código (NESTJS, 2017).

O banco de dados escolhido para armazenar as informações da aplicação é o MySQL, um dos sistemas de gerenciamento de banco de dados relacional mais populares e confiáveis, amplamente utilizado em aplicações de alta performance (ORACLE, 2024).

Para a integração entre o back-end e o banco de dados, foi utilizado o Prisma, um ORM (*Object-Relational Mapping*) de nova geração que simplifica o acesso e a manipulação dos dados (PRISMA, 2024). Além de fornecer uma interface intuitiva para realizar consultas e operações no banco de dados, o Prisma também facilita o gerenciamento de migrações de esquema, permitindo uma evolução contínua do banco de dados de forma segura e controlada.

Esse conjunto de tecnologias foi escolhido pela sua maturidade, suporte da comunidade e pela capacidade de atender às demandas atuais e futuras da aplicação, garantindo assim uma base sólida para o desenvolvimento de funcionalidades complexas, com a devida atenção à performance, segurança e escalabilidade.

2.2.1. Nestjs

O NestJS é um framework para desenvolvimento de aplicações back-end em Node.js, que adota uma arquitetura modular inspirada nos princípios do Angular. Ele é escrito em TypeScript, aproveitando os benefícios dessa linguagem, como tipagem estática, que melhora a segurança e a manutenibilidade do código. O NestJS facilita a construção de aplicações escaláveis e bem organizadas, com suporte embutido para a Injeção de Dependência (Dependency Injection), Middleware, Pipes, Guards, e muito mais. Ele também se integra facilmente com outras bibliotecas e frameworks do ecossistema Node.js, como Express ou Fastify, permitindo flexibilidade no desenvolvimento (GADO, 2024).

2.2.2. Mysql

O MySQL é um dos sistemas de gerenciamento de banco de dados (SGBD) mais populares e amplamente utilizados no mundo. Desenvolvido inicialmente pela MySQL AB, e posteriormente adquirido pela Oracle Corporation, o MySQL é conhecido por sua performance, confiabilidade, e facilidade de uso. Ele é uma escolha comum para aplicações web devido à sua eficiência na gestão de grandes volumes de dados e suporte a transações ACID (Atomicidade, Consistência, Isolamento e Durabilidade). O MySQL também oferece uma ampla gama de funcionalidades, como suporte a vários mecanismos de armazenamento (ex: InnoDB, MyISAM), replicação de dados, e otimização de consultas. Essas características fazem do MySQL uma opção robusta tanto para pequenas aplicações quanto para grandes sistemas corporativos (ORACLE, 2024).

2.2.3. API ViaCep

Para facilitar o preenchimento de endereços durante o cadastro de usuários e parceiros no sistema, foi utilizada a API pública do ViaCEP. Essa API permite, a partir do número do CEP informado, obter automaticamente dados como logradouro, bairro, cidade e estado, agilizando o processo de cadastro e evitando erros manuais. A integração foi feita por meio de requisições HTTP no back-end, consumindo os dados fornecidos pela API e retornando-os já formatados para o front-end. Por ser gratuita, leve e de fácil uso, a API do ViaCEP se mostrou uma solução eficiente e adequada ao propósito do sistema.

2.2.4. API Mercado Pago

Para a identificação e validação dos pagamentos realizados no sistema de venda de cupons, foi integrada à aplicação a API oficial do Mercado Pago (MERCADO PAGO, 2025). A integração foi feita utilizando a biblioteca oficial do Mercado Pago para Node.js, o que facilitou o consumo dos endpoints da plataforma e garantiu uma comunicação segura com os serviços de pagamento. Por meio dessa biblioteca, foi possível gerar preferências de pagamento, consultar o status das transações e receber notificações automáticas sobre as atualizações dos pagamentos.

O sistema foi configurado para trabalhar com webhooks, ou seja, notificações automáticas que o próprio Mercado Pago envia ao servidor back-end sempre que ocorre uma mudança no status de um pagamento (como aprovação, rejeição ou cancelamento). O servidor foi ajustado para receber essas notificações em uma rota específica, interpretar os dados enviados e atualizar o status da compra no banco de dados. Quando o pagamento é aprovado, o sistema libera automaticamente o voucher correspondente para o usuário.

Essa abordagem garante um fluxo totalmente automatizado e seguro, permitindo que a liberação dos cupons ocorra apenas após a confirmação do pagamento. Além disso, a escolha pela API do Mercado Pago se justifica por sua ampla aceitação no mercado brasileiro, facilidade de integração, suporte a múltiplas formas de pagamento (como cartão de crédito, Pix e boleto bancário) e uma documentação clara que favoreceu o desenvolvimento da solução.

2.3.METODOLOGIA ÁGIL SCRUM

O Scrum se trata de um framework de gerenciamento utilizado por equipes para se auto-organizarem e trabalharem de forma colaborativa em direção a um objetivo comum. Esse framework estabelece um conjunto de reuniões, ferramentas e papéis que facilitam a entrega eficiente de projetos. Assim como uma equipe esportiva que treina para uma competição importante, o Scrum promove o autogerenciamento, incentiva o aprendizado contínuo com base em experiências passadas e permite que as equipes se adaptem rapidamente às mudanças. Esse método é amplamente utilizado por equipes de desenvolvimento de software para resolver problemas complexos de maneira eficiente e sustentável (AMAZON WEB SERVICES, 2024).

A metodologia Scrum é baseada em três pilares: transparência, inspeção e adaptação. Isso significa que a equipe deve ser transparente sobre seu progresso, inspecionar regularmente seu trabalho e se adaptar às mudanças de forma rápida e eficiente. O Scrum também define

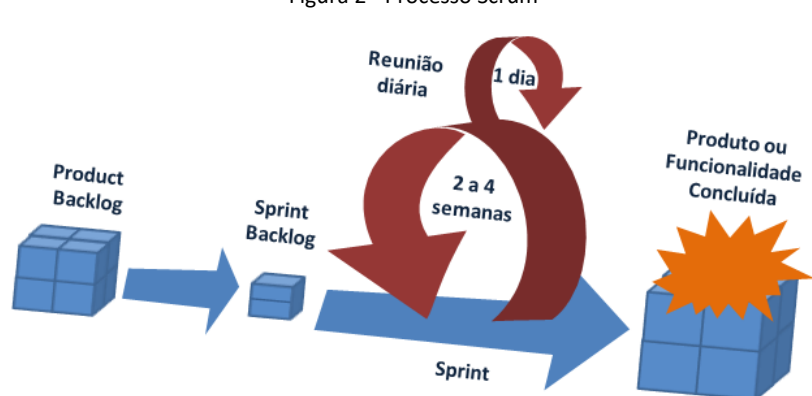
alguns papéis importantes, incluindo o Scrum Master, o Product Owner e os membros do time de desenvolvimento (AMAZON WEB SERVICES, 2024).

O Scrum Master tem o papel de garantir que a equipe siga corretamente a metodologia Scrum e de remover qualquer obstáculo que possa atrapalhar o progresso. Ele também facilita os encontros do Scrum e assegura que a equipe esteja adotando as melhores práticas. O Product Owner, por sua vez, é o representante dos interesses do cliente e garante que o trabalho da equipe esteja alinhado com os objetivos do negócio. Além disso, ele é responsável por manter o backlog do produto priorizado e atualizado constantemente. Já os membros da equipe de desenvolvimento são encarregados de planejar, construir e entregar o produto final. Eles colaboram para atingir as metas da sprint e assegurar que o produto atenda às expectativas do cliente, além de tomar decisões técnicas e garantir que o trabalho seja realizado de maneira eficiente e eficaz (AMAZON WEB SERVICES, 2024).

A metodologia Scrum também estabelece algumas cerimônias essenciais, como o Daily Scrum, a Sprint Review e a Sprint Retrospective. O Daily Scrum é uma breve reunião diária onde a equipe se encontra para revisar o andamento das tarefas e planejar os próximos passos. A Sprint Review ocorre ao final de cada sprint e é o momento em que a equipe apresenta o trabalho concluído ao cliente, discutindo possíveis melhorias para o próximo ciclo. Já a Sprint Retrospective, também realizada ao término de cada sprint, é uma reunião em que a equipe avalia o que funcionou bem e identifica áreas que precisam ser aprimoradas para a próxima sprint (AMAZON WEB SERVICES, 2024).

Além disso, o Scrum define o conceito de Sprint, um período fixo (geralmente de 1 a 4 semanas) no qual a equipe colabora para completar um conjunto de tarefas priorizadas. Ao final de cada sprint, o objetivo é entregar uma parte funcional do produto, sempre com foco em melhorar continuamente o processo (AMAZON WEB SERVICES, 2024).

Figura 2 - Processo Scrum



Processo Scrum

Fonte: MindMaster (2024). Scrum: A Metodologia Ágil Explicada de forma Definitiva.

No Scrum, temos os artefatos, que são elementos essenciais que ajudam a organizar o trabalho e monitorar o progresso da equipe. O Product Backlog é uma lista ordenada de tudo o que precisa ser feito no produto, mantida pelo Product Owner. Essa lista inclui funcionalidades, melhorias, correções e outras tarefas, priorizadas para que a equipe sempre trabalhe no que agrega mais valor. O Product Backlog é dinâmico e pode ser atualizado conforme novas informações surgem ou as necessidades mudam. O Sprint Backlog é um subconjunto do Product Backlog, composto pelos itens que a equipe se compromete a completar durante a sprint. Ele é criado na reunião de planejamento da sprint e serve como um guia para a equipe durante o ciclo de trabalho, com um plano detalhado de como cada tarefa será realizada. O objetivo é entregar, ao final da sprint, um Incremento, que é a soma de todos os itens completados, resultando em uma versão utilizável e funcional do produto. Cada incremento deve estar pronto para uso e atender aos critérios de aceitação definidos pela equipe (AMAZON WEB SERVICES, 2024).

3. METODOLOGIA

3.1. METODOLOGIA DE PESQUISA

Para a condução da pesquisa, será utilizado a metodologia de pesquisa do estudo de caso. Essa metodologia se trata de uma abordagem científica de pesquisa que examina um fenômeno contemporâneo dentro de seu contexto real, considerando as diversas variáveis que o impactam. Trata-se de uma análise detalhada e profunda, conduzida de forma rigorosa, sobre uma instituição, comunidade ou indivíduo, permitindo a investigação de fenômenos complexos (MENEZES, 2024).

No campo de Sistema da Informação, o método do caso de uso é muito utilizado para a investigação de como os sistemas serão desenvolvidos, implementados. Ele permite investigar as interações entre tecnologia, pessoas e organizações, considerando a complexidade dos SI no mundo real. Esse método envolve fases como o planejamento, coleta de dados, análise e interpretação, com foco na compreensão dos SI como sistemas sociotécnicos complexos, e não apenas como artefatos tecnológicos (ANTONIO, N., FORNAZIN, M., ARAUJO, R, 2018).

No presente trabalho, a metodologia de estudo de caso foi aplicada com o objetivo de compreender e propor uma solução real para uma necessidade específica identificada no comércio local de Florianópolis: a ausência de uma plataforma acessível para divulgação de cupons promocionais. A aplicação dessa metodologia ocorreu em etapas bem definidas. A primeira foi o planejamento, em que se definiu o escopo da pesquisa, os objetivos, as perguntas de investigação e os recursos necessários. Em seguida, foi realizada a coleta de dados, que incluiu a análise de referências bibliográficas, levantamento de demandas de mercado, observação de plataformas existentes e identificação de oportunidades de melhoria. A fase seguinte foi a análise e interpretação, na qual os dados foram organizados e confrontados com o problema levantado, fundamentando o desenvolvimento do sistema proposto. Por fim, a fase de construção e validação consistiu na implementação do sistema de e-commerce e na verificação prática de seu funcionamento, com base nos objetivos estabelecidos e nas funcionalidades planejadas. Essa abordagem permitiu que o desenvolvimento da aplicação fosse conduzido de forma contextualizada, realista e alinhada às necessidades do público-alvo.

3.2. METODOLOGIA DE DESENVOLVIMENTO DE SOFTWARE

A metodologia Scrum foi escolhida para o desenvolvimento do sistema de e-commerce devido à sua abordagem ágil, iterativa e incremental, que permite maior flexibilidade e

adaptação durante o processo de desenvolvimento. Essa metodologia facilita o acompanhamento contínuo do progresso do projeto, melhora a comunicação entre os membros da equipe e permite entregas frequentes de funcionalidades, garantindo que o produto final esteja alinhado com as necessidades do usuário e dos stakeholders.

Para a aplicação no projeto, o framework ágil Scrum foi adaptado a uma equipe reduzida, composta por poucos integrantes, com papéis definidos conforme as necessidades do desenvolvimento. O papel de Product Owner (PO) foi exercido por Lucas Lopes, responsável por organizar e priorizar o backlog do produto, além de garantir que as funcionalidades estivessem alinhadas com os objetivos do sistema e as expectativas dos usuários. O papel de Scrum Master foi desempenhado por Tiago Rafael Oliveira das Virgens, que atuou facilitando o processo ágil, assegurando o cumprimento dos princípios do Scrum e removendo impedimentos durante as sprints. No desenvolvimento, Tiago Rafael também assumiu a responsabilidade pela construção do back-end da aplicação, utilizando práticas modernas de arquitetura e desenvolvimento de APIs. Já o front-end do sistema ficou sob responsabilidade de Mateus França da Cunha, que trabalhou na implementação da interface voltada ao usuário final.

O desenvolvimento foi dividido entre as camadas de interface (front-end) e lógica de negócios (back-end). Este trabalho apresenta exclusivamente a parte do back-end, detalhando as decisões técnicas, ferramentas utilizadas e funcionalidades implementadas na API. A integração entre as duas partes ocorre por meio de endpoints RESTful, permitindo a comunicação eficiente entre o sistema e a interface construída por Mateus.

As sprints foram organizadas em ciclos mensais, com duração de quatro semanas cada. A seguir, a Tabela 1 apresenta o cronograma de execução das sprints, destacando os períodos de cada uma e os principais objetivos atingidos ao seu término:

Tabela 1 - Sprints

Sprint	Período	Objetivo
01	01/03/2025 – 31/03/2025	Criação da estrutura do projeto, modelagem do banco de dados e autenticação de usuários
02	01/04/2025 – 30/04/2025	Implementação das rotas de gerenciamento de cupons e integração com a API ViaCEP

03	01/05/2025 – 31/05/2025	Integração com o Mercado Pago e funcionalidades de carrinho de compras
04	01/06/2025 – 21/06/2025	Testes finais, ajustes e geração de documentação via Swagger

Fonte: Próprio autor, 2025.

Esse formato de organização permitiu uma visão clara dos avanços do projeto, facilitando a adaptação às mudanças de escopo e possibilitando a entrega de um produto funcional e coerente com os objetivos estabelecidos.

3.3.PROCESSO DE INTEGRAÇÃO COM O FRONT-END

A integração entre o back-end, desenvolvido em NestJS, e o front-end do sistema de e-commerce foi realizada por meio de uma API RESTful, que garante a comunicação entre ambas as camadas da aplicação de forma padronizada, segura e escalável. Esse modelo de integração permite que o front-end consuma os dados expostos pelo back-end por meio de requisições HTTP nos métodos GET, POST, PUT, PATCH e DELETE, obedecendo às práticas recomendadas de construção de APIs modernas.

Durante o desenvolvimento, foram definidos endpoints claros e bem documentados, facilitando a implementação das funcionalidades no front-end, como o cadastro e autenticação de usuários, listagem e compra de cupons, gerenciamento do carrinho e integração com o serviço de pagamento. A ferramenta Swagger foi utilizada para documentar automaticamente a API, fornecendo uma interface visual que permite aos desenvolvedores do front-end testarem as rotas e compreenderem os dados esperados e retornados.

Além disso, foram utilizadas técnicas como CORS (Cross-Origin Resource Sharing) para permitir que o front-end, hospedado em domínio diferente do back-end, pudesse consumir a API sem restrições de segurança. Durante o processo de integração, foi estabelecido um fluxo de testes contínuos entre as duas partes da aplicação, garantindo que as funcionalidades implementadas estivessem de acordo com os requisitos definidos.

A escolha por uma API RESTful também possibilita futuras integrações com outros sistemas, como aplicativos mobile ou plataformas de terceiros (por exemplo, gateways de

pagamento ou sistemas de CRM), reforçando o caráter escalável e modular da arquitetura proposta.

4. RESULTADOS E DISCUSSÕES

A seguir, são apresentadas as principais rotas da API desenvolvida para o sistema. Essa API RESTful foi documentada utilizando o Swagger, permitindo a visualização, teste e validação das funcionalidades de forma prática e intuitiva. Cada rota está associada a uma funcionalidade específica do sistema e representa o back-end da aplicação em operação. Para ver o Swagger é necessário acessar <https://editor.swagger.io/>. E colar o conteúdo do link abaixo.

Documentação pública da API (Swagger): <https://objectstorage.sa-saopaulo-1.oraclecloud.com/n/gr500xifmria/b/bucket-pessoal/o/TARRAFADA%2Fswagger.json>.

Acesso ao sistema desenvolvido para avaliação da banca:

Área administrativa: <https://www.mateusfranca.site/loginAuthAdm>

Área do parceiro (estabelecimento): <https://www.mateusfranca.site/loginAuthPartner>

Área do cliente: <https://www.mateusfranca.site/loginAuthCustomer>

Login: mateusfranca0220@gmail.com

Senha: General123@

4.1. SWAGGER

O Swagger é uma ferramenta amplamente utilizada no desenvolvimento de APIs RESTful para documentação automática de rotas e endpoints. Através do uso de anotações e estruturas padronizadas, o Swagger permite que desenvolvedores descrevam a estrutura da API de forma clara, promovendo uma comunicação eficaz entre o front-end e o back-end e facilitando testes manuais e validações durante o desenvolvimento.

Ao integrar o Swagger ao projeto, é possível gerar uma interface interativa que lista todos os endpoints disponíveis, os métodos HTTP suportados (GET, POST, PUT, PATCH, DELETE), os parâmetros exigidos em cada requisição, os modelos de entrada e saída, e os possíveis códigos de resposta da API. Isso contribui diretamente para a transparência, manutenção e escalabilidade do sistema.

Além disso, o Swagger se destaca por ser compatível com o padrão OpenAPI, permitindo a exportação da especificação da API para uso em outras ferramentas, como geradores de SDKs ou validadores de segurança.

No contexto deste projeto, o Swagger foi utilizado para documentar todas as rotas

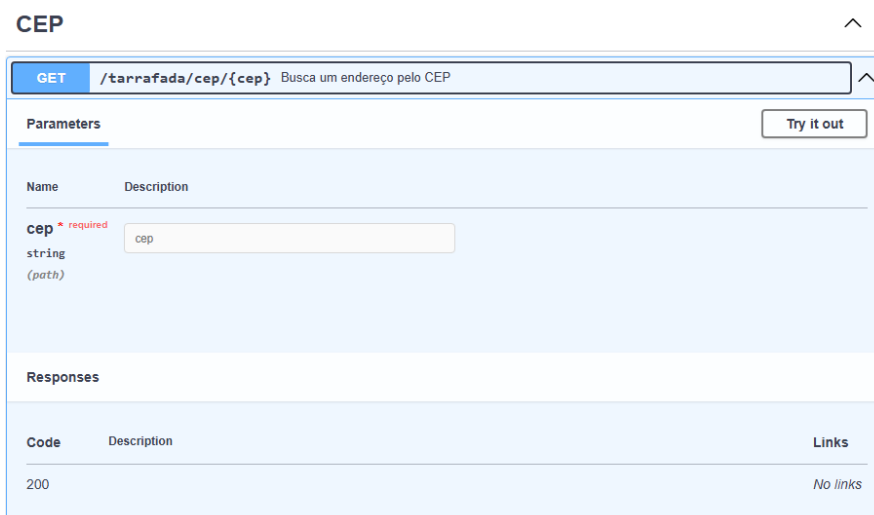
implementadas no NestJS, permitindo que os testes das funcionalidades fossem realizados de forma visual e prática, sem a necessidade de ferramentas externas como Postman ou cURL. Sua adoção foi essencial para garantir o alinhamento entre as equipes de desenvolvimento e para assegurar que a API estivesse funcionando conforme o planejado desde as fases iniciais da implementação.

4.2. ROTAS GERAIS

4.2.1. Rota de cep

A rota GET /tarrafada/cep/{cep} permite buscar um endereço completo a partir do CEP informado como parâmetro obrigatório na URL.

Figura 3 – Rota de cep

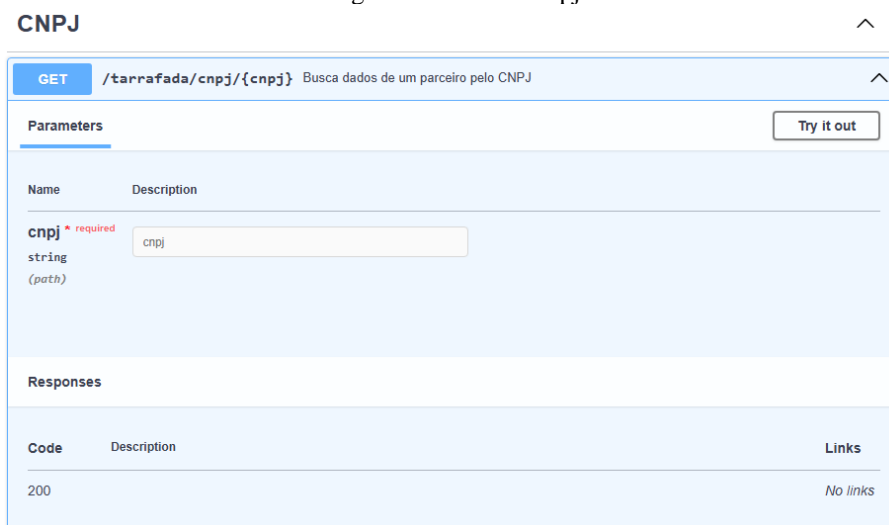


Fonte: Próprio autor, 2025

4.2.2. Rota de cnpj

A rota GET /tarrafada/cnpj/{cnpj} permite consultar os dados de uma empresa a partir do CNPJ informado como parâmetro obrigatório na URL.

Figura 4 – Rota de cnpj



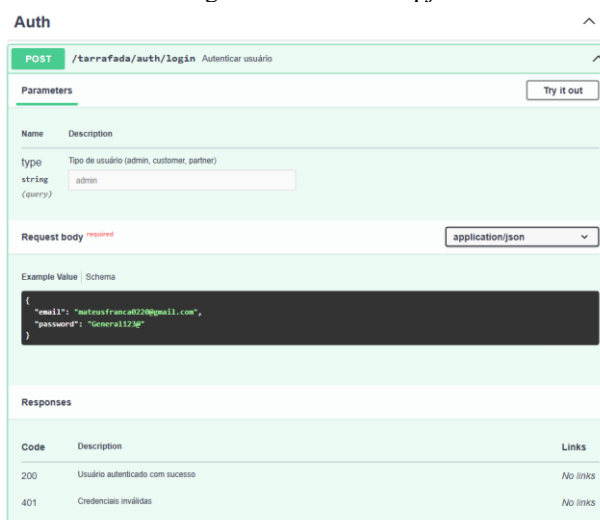
Fonte: Próprio autor, 2025

4.3. ROTAS DE AUTENTICAÇÃO

4.3.1. Rota de login

A rota POST /tarrafada/auth/login autentica o usuário. Recebe no corpo da requisição o email e a senha e um parâmetro obrigatório na query chamado type, que indica o tipo de usuário (admin, customer ou partner).

Figura 5 – Rota de cnpj



Fonte: Próprio autor, 2025

4.3.2. Rota de solicitar redefinição de senha

A rota POST `/tarrafada/auth/password-reset-request` permite solicitar a redefinição de senha. Recebe no corpo da requisição o email do usuário e o type indicando o tipo de usuário.

Figura 6 - Rota de solicitar redefinição de senha

The image shows the Swagger UI for the endpoint `POST /tarrafada/auth/password-reset-request`. The interface includes a 'Parameters' section with 'No parameters', a 'Request body' section with a dropdown set to 'application/json', and an 'Example Value' section showing a JSON object: `{ "email": "usuario@exemplo.com", "type": "admin" }`. The 'Responses' section contains a table with two entries: a 200 status code for 'Email de redefinição enviado' and a 404 status code for 'Usuário não encontrado'.

Code	Description	Links
200	Email de redefinição enviado	No links
404	Usuário não encontrado	No links

Fonte: Próprio autor, 2025

4.3.3. Rota de redefinição da senha

A rota POST `/tarrafada/auth/password-reset/{token}` redefine a senha do usuário. Recebe o token de redefinição na URL e a nova senha no corpo da requisição.

Figura 7 - Rota de solicitar redefinição de senha

The image shows the Swagger UI for the endpoint `POST /tarrafada/auth/password-reset/{token}`. The 'Parameters' section includes a path parameter 'token' of type 'string' with a description 'Token de redefinição de senha'. The 'Request body' section has a dropdown set to 'application/json' and an 'Example Value' section showing a JSON object: `{ "newPassword": "novaSenha123" }`. The 'Responses' section contains a table with two entries: a 200 status code for 'Senha redefinida com sucesso' and a 400 status code for 'Token inválido ou expirado'.

Code	Description	Links
200	Senha redefinida com sucesso	No links
400	Token inválido ou expirado	No links

Fonte: Próprio autor, 2025

4.3.4. Rota para alterar senha

A rota PUT /tarrafada/auth/change-password permite alterar a senha do usuário. Recebe no corpo da requisição o email, a nova senha (newPassword) e o type do usuário.

Figura 8 - Rota para alterar senha

The screenshot shows the Swagger UI for the endpoint PUT /tarrafada/auth/change-password. The interface includes a 'Parameters' section with 'No parameters', a 'Request body' section with a dropdown set to 'application/json', and an 'Example Value' section showing a JSON object: { "email": "matheusfranca0220@gmail.com", "newPassword": "NovaSenha123e", "type": "admin" }. Below this is a 'Responses' table with three rows: 200 (Senha alterada com sucesso), 400 (A nova senha não pode ser igual à senha atual), and 404 (Usuário não encontrado).

Code	Description	Links
200	Senha alterada com sucesso	No links
400	A nova senha não pode ser igual à senha atual	No links
404	Usuário não encontrado	No links

Fonte: Próprio autor, 2025

4.3.5. Rota para obter informações do usuário autenticado

A rota GET /tarrafada/auth/me retorna as informações do usuário autenticado. Requer o token de autorização no cabeçalho da requisição.

Figura 9 - Rota para buscar informações do usuário autenticado

The screenshot shows the Swagger UI for the endpoint GET /tarrafada/auth/me. The 'Parameters' section includes a required 'Authorization' header parameter with a text input field. The 'Responses' table has two rows: 200 (Informações do usuário retomadas com sucesso) and 401 (Token inválido ou não fornecido).

Code	Description	Links
200	Informações do usuário retomadas com sucesso	No links
401	Token inválido ou não fornecido	No links

Fonte: Próprio autor, 2025

4.4. ROTAS DE CLIENTE

4.4.1. Rota para cadastrar um cliente

A rota POST /tarrafada/customers cria um novo cliente, recebendo no corpo da requisição os dados pessoais e de contato do cliente.

Figura 10 - Rota para cadastrar um cliente

The screenshot shows a REST client interface for the endpoint `POST /tarrafada/customers`. The title is "Customers". The method is "POST" and the description is "Cria um cliente". There are no parameters. The request body is required and set to "application/json". An example JSON body is shown in a dark box:

```
{
  "id_cus": "string",
  "name_cus": "string",
  "cpf_cus": "string",
  "telephone_cus": "string",
  "birth_date_cus": "string",
  "email_cus": "string",
  "password_cus": "string",
  "id_addr": "string"
}
```

The responses section shows a table with one entry:

Code	Description	Links
201		No links

Fonte: Próprio autor, 2025

4.4.2. Rota para listar todos os clientes

A rota GET /tarrafada/customers retorna todos os clientes, podendo receber parâmetros opcionais na query para paginação (page, limit) e filtro por texto (searchText).

Figura 11 - Rota para listar os clientes

The screenshot shows a REST client interface for the endpoint `GET /tarrafada/customers`. The title is "Customers". The method is "GET" and the description is "Retornar todos os clientes". There are three query parameters:

- `page`: string (query), value: page
- `limit`: string (query), value: limit
- `searchText`: string (query), value: searchText

The responses section shows a table with one entry:

Code	Description	Links
200		No links

Fonte: Próprio autor, 2025

4.4.3. Rota para buscar informações detalhadas de um cliente

A rota GET `/tarrafada/customers/{id}` retorna os dados de um cliente específico, identificado pelo ID fornecido como parâmetro na URL.

Figura 12 - Rota para buscar informações detalhadas de um cliente

The screenshot shows a REST client interface for the endpoint `GET /tarrafada/customers/{id}` with the description "Retornar um cliente por ID". The interface is divided into several sections:

- Parameters:** A table with columns "Name" and "Description". It contains one parameter: `id` (string, path), marked as required. A text input field contains the value "id". A "Try it out" button is located to the right.
- Responses:** A table with columns "Code", "Description", and "Links". It shows a response with code "200" and "No links".

Fonte: Próprio autor, 2025

4.4.4. Rota para atualizar um cliente

A rota PATCH `/tarrafada/customers/{id}` atualiza os dados de um cliente específico, identificado pelo ID fornecido como parâmetro na URL.

Figura 13 - Rota para atualizar um cliente

The screenshot shows a REST client interface for the endpoint `PATCH /tarrafada/customers/{id}` with the description "Atualiza um cliente por ID". The interface includes the following sections:

- Parameters:** A table with columns "Name" and "Description". It contains one parameter: `id` (string, path), marked as required. A text input field contains the value "id". A "Try it out" button is located to the right.
- Request body:** A section labeled "Request body" (required) with a dropdown menu set to "application/json".
- Example Value:** A section with a "Schema" tab selected, showing an empty JSON object `{}`.
- Responses:** A table with columns "Code", "Description", and "Links". It shows a response with code "200" and "No links".

Fonte: Próprio autor, 2025

4.4.5. Rota para excluir um cliente

A rota DELETE `/tarrafada/customers/{id}` exclui uma categoria específica, identificada pelo ID fornecido como parâmetro na URL.

Figura 14 - Rota para excluir um cliente

Name	Description
id * required	id
string	
(path)	

Code	Description	Links
200		No links

Fonte: Próprio autor, 2025

4.4.6. Rota para gerar um relatório de clientes

A rota GET `/tarrafada/customers/report/pdf` gera um relatório de clientes em PDF, com base nos parâmetros obrigatórios `date_in` e `date_out`, e o parâmetro opcional `status`.

Figura 15 - Rota para gerar um relatório de clientes

Name	Description
date_in * required	date_in
string	
(query)	
date_out * required	date_out
string	
(query)	
status	status
string	
(query)	

Code	Description	Links
200		No links

Fonte: Próprio autor, 2025

4.5. ROTAS DE CATEGORIA

4.5.1. Rota para cadastrar uma categoria

A rota POST /tarrafada/categories cria uma nova categoria, recebendo no corpo da requisição os dados da categoria.

Figura 16 - Rota para cadastrar uma categoria

The screenshot shows a REST client interface for the endpoint `POST /tarrafada/categories`. The title bar indicates the method and path, along with a description: "Cria uma categoria". Below the title bar, there are sections for "Parameters" (showing "No parameters"), "Request body" (with a dropdown menu set to "application/json" and a "required" label), and "Responses". The "Request body" section contains an "Example Value" field with a JSON object: `{ "name_cat": "string", "description_cat": "string" }`. The "Responses" section is a table with columns "Code", "Description", and "Links". It shows a response with code "201" and "No links".

Code	Description	Links
201		No links

Fonte: Próprio autor, 2025

4.5.2. Rota para listar todas as categorias

A rota GET /tarrafada/categories retorna todos os clientes, podendo receber parâmetros opcionais na query para paginação (page, limit) e filtro por texto (searchText) e o (status).

Figura 17 - Rota para listar todas as categorias

The screenshot shows a REST client interface for the endpoint `GET /tarrafada/categories`. The title bar indicates the method and path, along with a description: "Retornar todas as categorias". Below the title bar, there are sections for "Parameters" and "Responses". The "Parameters" section contains a table with columns "Name" and "Description". It lists four query parameters: "page" (string, query), "limit" (string, query), "searchText" (string, query), and "status" (string, query). Each parameter has a corresponding input field. The "Responses" section is a table with columns "Code", "Description", and "Links". It shows a response with code "200" and "No links".

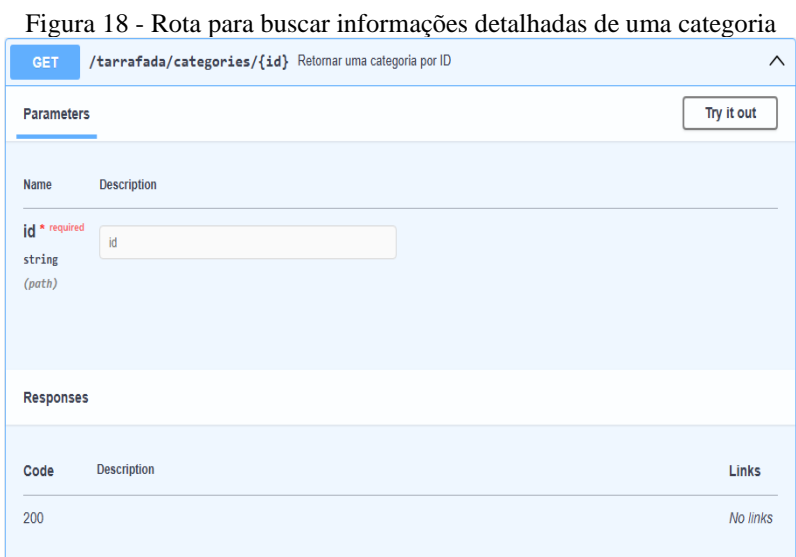
Name	Description
page	string (query)
limit	string (query)
searchText	string (query)
status	string (query)

Code	Description	Links
200		No links

Fonte: Próprio autor, 2025

4.5.3. Rota para buscar informações detalhadas de uma categoria

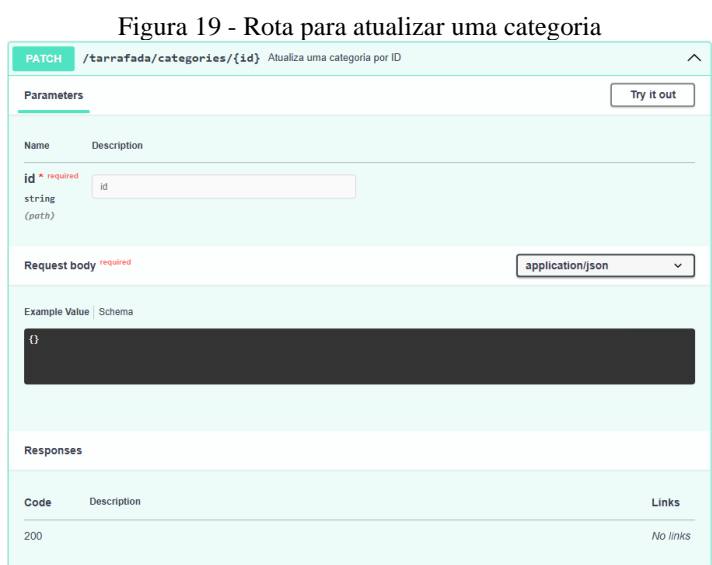
A rota GET `/tarrafada/categories/{id}` retorna os dados de uma categoria específica, identificado pelo ID fornecido como parâmetro na URL.



Fonte: Próprio autor, 2025

4.5.4. Rota para atualizar uma categoria

A rota PATCH `/tarrafada/categories/{id}` atualiza os dados de uma categoria específica, identificado pelo ID fornecido como parâmetro na URL.



Fonte: Próprio autor, 2025

4.5.5. Rota para excluir uma categoria

A rota DELETE /tarrafada/categories/{id} exclui uma categoria específica, identificada pelo ID fornecido como parâmetro na URL.

Figura 20 - Rota para excluir uma categoria

DELETE /tarrafada/categories/{id} Exclui uma categoria por ID

Try it out

Name	Description
id * required	
string	id
(path)	

Responses

Code	Description	Links
200		No links

Fonte: Próprio autor, 2025

4.5.6. Rota para gerar um relatório de categorias

A rota GET /tarrafada/customers/report/pdf gera um relatório de clientes em PDF, com base nos parâmetros obrigatórios date_in e date_out, e o parâmetro opcional status.

Figura 21 - Rota para gerar um relatório de categorias

GET /tarrafada/categories/report/pdf Gera um relatório de categorias em PDF

Try it out

Name	Description
date_in * required	
string	date_in
(query)	
date_out * required	
string	date_out
(query)	
status	
string	status
(query)	

Responses

Code	Description	Links
200		No links

Fonte: Próprio autor, 2025

4.6. ROTAS DE PARCEIRO

4.6.1. Rota para cadastrar um parceiro

A rota POST /tarrafada/partners cria um novo parceiro, recebendo no corpo da requisição os dados do parceiro.

Figura 22 - Rota para cadastrar um parceiro

The screenshot displays a REST client interface for the endpoint `POST /tarrafada/partners` with the description "Cria um parceiro". The "Parameters" section is empty. The "Request body" is set to `application/json` and contains the following JSON schema:

```
{
  "id_par": "string",
  "name_par": "string",
  "password_par": "string",
  "email_par": "string",
  "telephone_par": "string",
  "fantasy_name_par": "string",
  "cnpj_par": "string",
  "id_add_fk": "string"
}
```

The "Responses" section shows a table with one entry:

Code	Description	Links
201		No links

Fonte: Próprio autor, 2025

4.6.2. Rota para listar todos os parceiros

A rota GET /tarrafada/partners retorna todos os parceiros, podendo receber parâmetros opcionais na query para paginação (page, limit) e filtro por texto (searchText) e o (status).

Figura 23 - Rota para listar todos os parceiros

The screenshot displays a REST client interface for the endpoint `GET /tarrafada/partners` with the description "Retornar todos os parceiros". The "Parameters" section includes the following query parameters:

- `page`: string (query) with input field "page"
- `limit`: string (query) with input field "limit"
- `searchText`: string (query) with input field "searchText"
- `status`: any (query) with input field "status"

The "Responses" section shows a table with one entry:

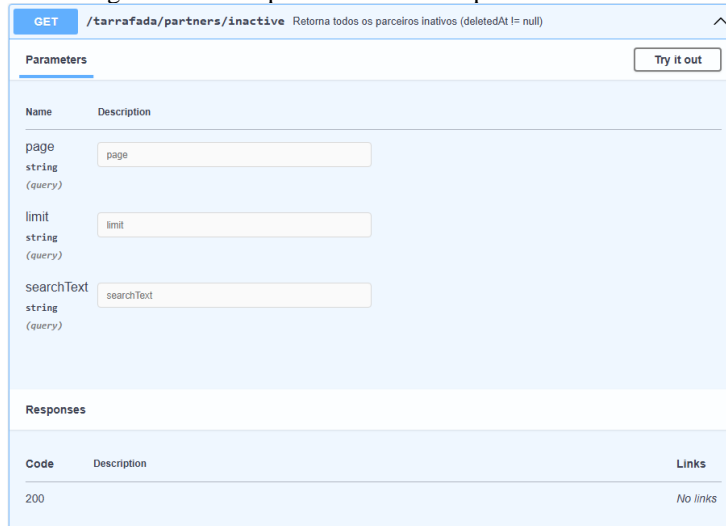
Code	Description	Links
200		No links

Fonte: Próprio autor, 2025

4.6.3. Rota para listar todas os parceiros inativos

A rota GET /tarrafada/partners/inactive retorna todos os parceiros inativos, podendo receber parâmetros opcionais na query para paginação (page, limit) e filtro por texto (searchText).

Figura 24 - Rota para listar todas os parceiros inativos

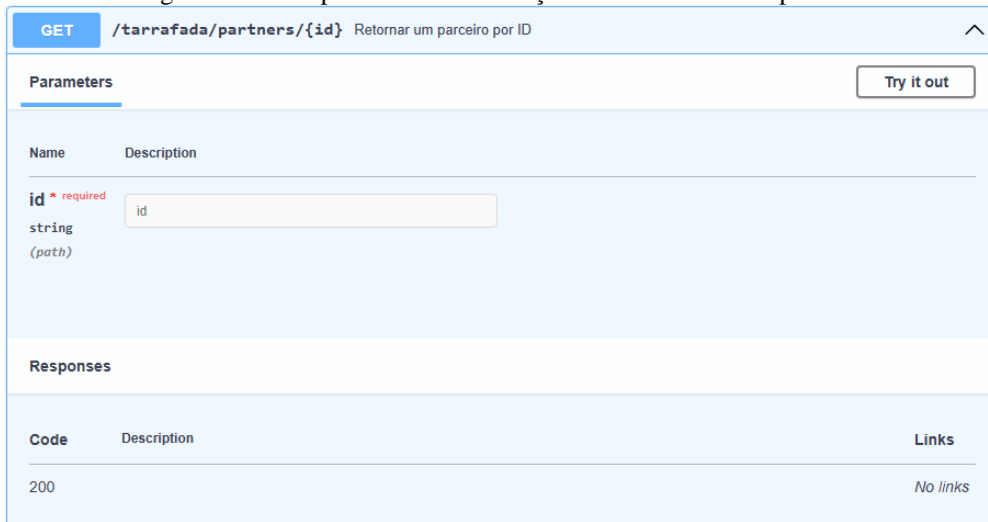


Fonte: Próprio autor, 2025

4.6.4. Rota para buscar informações detalhadas de um parceiro

A rota GET /tarrafada/partners/{id} retorna os dados de um parceiro específico, identificado pelo ID fornecido como parâmetro na URL.

Figura 25 - Rota para buscar informações detalhadas de um parceiro



Fonte: Próprio autor, 2025

4.6.5. Rota para atualizar um parceiro

A rota PATCH `/tarrafada/partners/{id}` atualiza os dados de um parceiro específico, identificado pelo ID fornecido como parâmetro na URL.

Figura 26 - Rota para atualizar um parceiro

The screenshot shows a REST client interface for a PATCH endpoint. The endpoint is `/tarrafada/partners/{id}` with the description "Atualiza um parceiro por ID". The interface is divided into several sections:

- Parameters:** A table with columns "Name" and "Description". It contains one parameter: `id` (required), type `string`, and location `(path)`. There is an input field for the value.
- Request body:** A section with a "required" label and a dropdown menu set to `application/json`. Below it, there are tabs for "Example Value" and "Schema". The "Example Value" tab shows an empty JSON object `{}`.
- Responses:** A table with columns "Code", "Description", and "Links". It shows a response with code `200` and description `No links`.

Fonte: Próprio autor, 2025

4.6.6. Rota para excluir um parceiro

A rota DELETE `/tarrafada/partners/{id}` exclui um parceiro específico, identificado pelo ID fornecido como parâmetro na URL.

Figura 27 - Rota para excluir um parceiro

The screenshot shows a REST client interface for a DELETE endpoint. The endpoint is `/tarrafada/partners/{id}` with the description "Exclui um parceiro por ID". The interface is divided into several sections:

- Parameters:** A table with columns "Name" and "Description". It contains one parameter: `id` (required), type `string`, and location `(path)`. There is an input field for the value.
- Responses:** A table with columns "Code", "Description", and "Links". It shows a response with code `200` and description `No links`.

Fonte: Próprio autor, 2025

4.6.7. Rota para gerar um relatório de parceiros

A rota GET /tarrafada/partners/report/pdf gera um relatório de parceiros em PDF, com base nos parâmetros obrigatórios date_in e date_out, e o parâmetro opcional status.

Figura 28 - Rota para gerar um relatório de parceiros

GET /tarrafada/partners/report/pdf Gera um relatório de parceiros em PDF

Parameters Try it out

Name	Description
date_in * required	date_in
string (query)	
date_out * required	date_out
string (query)	
status	status
string (query)	

Responses

Code	Description	Links
200		No links

Fonte: Próprio autor, 2025

4.6.8. Rota para ativar um parceiro

A rota PATCH /tarrafada/partners/active/{id} reativa um parceiro, definindo deletedAt como null. O parceiro é identificado pelo ID na URL.

Figura 29 - Rota para ativar um parceiro

PATCH /tarrafada/partners/active/{id} Ativa um parceiro (deletedAt = null)

Parameters Try it out

Name	Description
id * required	id
string (path)	

Responses

Code	Description	Links
200		No links

Fonte: Próprio autor, 2025

4.6.9. Rota para não aceitar o parceiro

A rota DELETE /tarrafada/partners/remove/{id} nega um parceiro específico, identificado pelo ID fornecido como parâmetro na URL.

Figura 30 - Rota para não aceitar o parceiro

DELETE /tarrafada/partners/remove/{id} Remove um parceiro

Try it out

Name	Description
id * required	
string (path)	id

Responses

Code	Description	Links
200		No links

Fonte: Próprio autor, 2025

4.6.10. Rota para gerar o dashboard de parceiro

A rota GET /tarrafada/partners/{id}/dashboard retorna dados do painel do parceiro, como total vendido, ticket médio, cupons vendidos, clientes e cupons mais vendidos. Requer o ID do parceiro na URL e aceita o parâmetro month na query (formato MM/YYYY ou 'Tudo').

Figura 31 - Rota para gerar o dashboard de parceiro

GET /tarrafada/partners/{id}/dashboard
Dashboard do parceiro: total vendido, ticket médio, cupons vendidos, clientes e cupons mais vendidos

Parameters Try it out

Name	Description
id * required	ID do parceiro
string (path)	<input type="text" value="123e4567-e89b-12d3-a456-426614174000"/>
month	Mês no formato MM/YYYY ou 'Tudo'
string (query)	<input type="text" value="month"/>

Responses

Code	Description	Links
200		No links

Fonte: Próprio autor, 2025

4.7. ROTAS DE ADMIN

4.7.1. Rota para cadastrar um administrador

A rota POST /tarrafada/admin/register cria um novo administrador, recebendo no corpo da requisição os dados do administrador.

Figura 32 - Rota para cadastrar um administrador

Admin ^

POST /tarrafada/admin/register ^

Parameters Try it out

No parameters

Request body required application/json

Example Value | Schema

```
{
  "name_admin": "João Silva",
  "email_admin": "joao.silva@empresa.com",
  "password_admin": "senha123"
}
```

Responses

Code	Description	Links
201		No links

Fonte: Próprio autor, 2025

4.7.2. Rota para gerar o dashboard de administrador

A rota GET /tarrafada/admin/dashboard retorna dados do painel do administrador, como total vendido, ticket médio, cupons vendidos, clientes e cupons mais vendidos. Requer o ID do parceiro na URL e aceita o parâmetro month na query (formato MM/YYYY ou 'Tudo').

Figura 33 - Rota para gerar o dashboard de administrador

Name	Description
month	Mês no formato MM/YYYY ou 'Tudo'

Code	Description	Links
200		No links

Fonte: Próprio autor, 2025

4.8. ROTAS DE ESTADO

4.8.1. Rota para listar todas os estados

A rota GET /tarrafada/states retorna todos os estados do sistema.

Figura 34 - Rota para listar todas os estados

Code	Description	Links
200		No links

Fonte: Próprio autor, 2025

4.8.2. Rota para buscar informações detalhadas de um estado

A rota GET /tarrafada/states/{id} retorna os dados de um estados específico, identificado pelo ID fornecido como parâmetro na URL.

Figura 35 - Rota para buscar informações detalhadas de um estado

The screenshot shows a REST client interface for the endpoint GET /tarrafada/states/{id}. The interface includes a 'Parameters' section with a table for defining path parameters. The parameter 'id' is defined as a required string (path). Below the parameters is a 'Responses' section with a table showing a 200 status code and no links.

Name	Description
id * required	
string	id
(path)	

Code	Description	Links
200		No links

Fonte: Próprio autor, 2025

4.9. ROTAS DE CIDADE

4.9.1. Rota para listar todas as cidades

A rota GET /tarrafada/cities retorna a lista de todas as cidades cadastradas no sistema.

Figura 36 - Rota para listar todas as cidades

The screenshot shows a REST client interface for the endpoint GET /tarrafada/cities. The interface includes a 'Parameters' section with the text 'No parameters'. Below is a 'Responses' section with a table showing a 200 status code and no links.

Code	Description	Links
200		No links

Fonte: Próprio autor, 2025

4.9.2. Rota para buscar informações detalhadas de uma cidade

A rota GET `/tarrafada/cities/{id}` retorna os dados de uma cidade específica, identificada pelo ID informado na URL.

Figura 37 - Rota para buscar informações detalhadas de uma cidade

The screenshot displays a REST client interface for the endpoint `GET /tarrafada/cities/{id}`. The interface includes a 'Try it out' button and a 'Parameters' section. The 'Parameters' section contains a table with the following data:

Name	Description
id * required	
string	id
(path)	

The 'Responses' section shows a table with the following data:

Code	Description	Links
200		No links

Fonte: Próprio autor, 2025

4.9.3. Rota para listar cidades de um estado

A rota GET `/tarrafada/cities/state/{id}` retorna a lista de cidades pertencentes ao estado identificado pelo ID informado na URL.

Figura 38 - Rota para listar cidades de um estado

The screenshot displays a REST client interface for the endpoint `GET /tarrafada/cities/state/{id}`. The interface includes a 'Try it out' button and a 'Parameters' section. The 'Parameters' section contains a table with the following data:

Name	Description
id * required	
string	id
(path)	

The 'Responses' section shows a table with the following data:

Code	Description	Links
200		No links

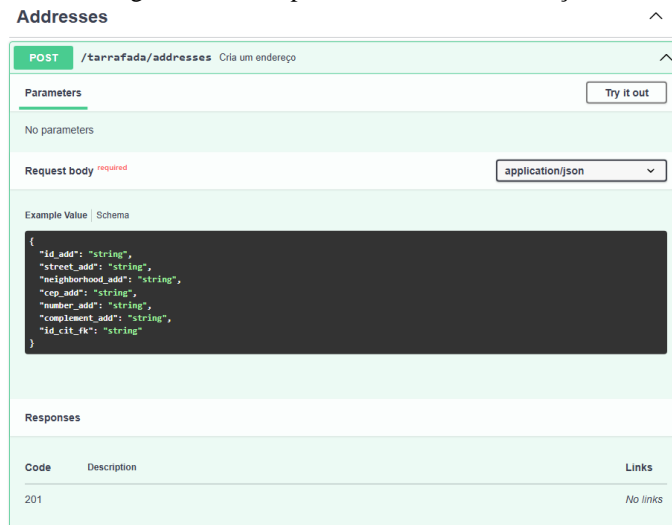
Fonte: Próprio autor, 2025

4.10. ROTAS DE ENDEREÇO

4.10.1. Rota para cadastrar um endereço

A rota POST `/tarrafada/addresses` cria um novo endereço, recebendo no corpo da requisição os dados do endereço.

Figura 39 - Rota para cadastrar um endereço

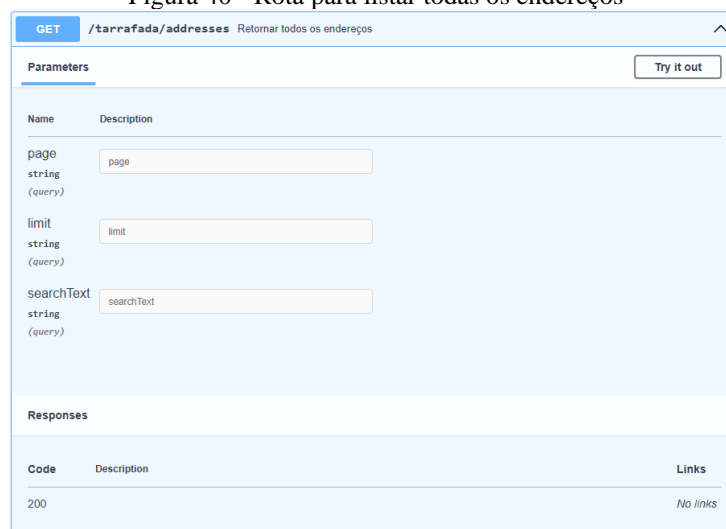


Fonte: Próprio autor, 2025

4.10.2. Rota para listar todas os endereços

A rota GET `/tarrafada/addresses` retorna todos os endereços, podendo receber parâmetros opcionais na query para paginação (`page`, `limit`) e filtro por texto (`searchText`).

Figura 40 - Rota para listar todas os endereços



Fonte: Próprio autor, 2025

4.10.3. Rota para buscar informações detalhadas de um endereço

A rota GET /tarrafada/addresses/{id} retorna os dados de um endereço específico, identificado pelo ID fornecido como parâmetro na URL.

Figura 41 - Rota para buscar informações detalhadas de um endereço

The screenshot shows a REST client interface for the endpoint GET /tarrafada/addresses/{id}. The interface includes a 'Parameters' section with a table for defining path parameters. The 'id' parameter is defined as a required string (path) with a text input field containing 'id'. Below the parameters is a 'Responses' section with a table showing a 200 status code and 'No links'.

Name	Description
id * required	
string	
(path)	

Code	Description	Links
200		No links

Fonte: Próprio autor, 2025

4.10.4. Rota para atualizar um endereço

A rota PATCH /tarrafada/addresses/{id} atualiza os dados de um endereço específico, identificado pelo ID fornecido como parâmetro na URL.

Figura 42 - Rota para atualizar um endereço

The screenshot shows a REST client interface for the endpoint PATCH /tarrafada/addresses/{id}. The interface includes a 'Parameters' section with a table for defining path parameters. The 'id' parameter is defined as a required string (path) with a text input field containing 'id'. Below the parameters is a 'Request body' section with a dropdown menu set to 'application/json' and a text area containing an empty JSON object '{}'. Below the request body is a 'Responses' section with a table showing a 200 status code and 'No links'.

Name	Description
id * required	
string	
(path)	

Request body required: application/json

```
{}
```


Code	Description	Links
200		No links

Fonte: Próprio autor, 2025

4.10.5. Rota para excluir um endereço

A rota DELETE /tarrafada/addresses/{id} exclui um endereço específico, identificado pelo ID fornecido como parâmetro na URL.

Figura 43 - Rota para excluir um endereço

Name	Description
id * required	id
string	(path)

Code	Description	Links
200	Exclui um endereço por ID	No links

Fonte: Próprio autor, 2025

4.11. ROTAS DE ENDEREÇO

4.11.1. Rota para cadastrar um cupom

A rota POST /tarrafada/coupons cria um novo cupom, recebendo no corpo da requisição os dados do cupom.

Figura 44 - Rota para cadastrar um cupom

```
{
  "id_cow": "string",
  "name_cow": "string",
  "description_cow": "string",
  "amount_cow": 0,
  "base_price_cow": 0,
  "percentage_cow": 0,
  "final_price_cow": 0,
  "images_cow": {},
  "additional_information_cow": "string",
  "active_cow": "string",
  "terms_cow": true,
  "id_cat_id": "string",
  "id_par_id": "string"
}
```

Code	Description	Links
201	Cria um novo cupom	No links

Fonte: Próprio autor, 2025

4.11.2. Rota para listar todas os cupons

A rota GET /tarrafada/coupons retorna todos os cupons, podendo receber parâmetros opcionais na query para paginação (page, limit) e filtro por texto (searchText).

Figura 45 - Rota para listar todas os cupons

The screenshot shows a REST client interface for the endpoint `GET /tarrafada/coupons` with the description "Retornar todos os cupons". The "Parameters" section contains three query parameters: `page` (string, query), `limit` (string, query), and `searchText` (string, query). The "Responses" section shows a table with one entry: a 200 status code with the description "No links".

Name	Description
page	page
limit	limit
searchText	searchText

Code	Description	Links
200		No links

Fonte: Próprio autor, 2025

4.11.3. Rota para adicionar imagem a um cupom

A rota PUT /tarrafada/coupons/images/{id} adiciona imagens a um cupom. Requer o ID do cupom na URL e os parâmetros `id_par` (parceiro) e `name_cou` (nome do cupom) na query.

Figura 46 - Rota para adicionar imagem a um cupom

The screenshot shows a REST client interface for the endpoint `PUT /tarrafada/coupons/images/{id}` with the description "Adicionar imagens a um cupom". The "Parameters" section contains three query parameters: `id` (string, path), `id_par` (string, query), and `name_cou` (string, query). The "Responses" section shows a table with one entry: a 200 status code with the description "No links".

Name	Description
id	id
id_par	id_par
name_cou	name_cou

Code	Description	Links
200		No links

Fonte: Próprio autor, 2025

4.11.4. Rota para listar o menu de cupons filtrados por categorias

A rota GET /tarrafada/coupons/menu retorna os itens do menu paginados, com informações dos cupons filtrados por categorias. Requer o parâmetro id_cus (ID do cliente) e aceita os parâmetros opcionais page, limit e searchText para paginação e busca.

Figura 47 - Rota para listar o menu de cupons filtrados por categorias

The screenshot displays a REST client interface for the endpoint `GET /tarrafada/coupons/menu`. The description of the endpoint is "Retorna itens do menu paginados com informações dos cupons filtrados por categorias".

Parameters:

Name	Description
page string (query)	page
limit string (query)	limit
searchText string (query)	searchText
id_cus * required string (query)	id_cus

Responses:

Code	Description	Links
200		No links

Fonte: Próprio autor, 2025

4.11.5. Rota para listar os cupons inativos

A rota GET /tarrafada/coupons/inactive retorna todos os cupons inativos (`active_cou = false`). Aceita os parâmetros opcionais page, limit e searchText para controle de paginação e filtro.

Figura 48 - Rota para listar os cupons inativos

The screenshot displays a REST client interface for the endpoint `GET /tarrafada/coupons/inactive`. The description of the endpoint is "Retorna todos os cupons inativos (active_cou = false)".

Parameters:

Name	Description
page string (query)	page
limit string (query)	limit
searchText string (query)	searchText

Responses:

Code	Description	Links
200		No links

Fonte: Próprio autor, 2025

4.11.6. Rota para buscar informações detalhadas de um cupom

A rota GET /tarrafada/coupons/{id} retorna os dados de um cupom específico, identificado pelo ID fornecido como parâmetro na URL.

Figura 49 - Rota para buscar informações detalhadas de um cupom

The screenshot shows a REST client interface for the endpoint GET /tarrafada/coupons/{id}. The interface is divided into several sections:

- Method and Path:** GET /tarrafada/coupons/{id} Retornar um cupom por ID
- Parameters:** A table with columns 'Name' and 'Description'. It lists a required parameter 'id' of type 'string (path)' with a text input field containing 'id'. A 'Try it out' button is visible in the top right.
- Responses:** A table with columns 'Code', 'Description', and 'Links'. It shows a response with code '200' and 'No links'.

Fonte: Próprio autor, 2025

4.11.7. Rota para atualizar um cupom

A rota PATCH /tarrafada/coupons/{id} atualiza os dados de um cupom específico, identificado pelo ID fornecido como parâmetro na URL.

Figura 50 - Rota para atualizar um cupom

The screenshot shows a REST client interface for the endpoint PATCH /tarrafada/coupons/{id}. The interface is divided into several sections:

- Method and Path:** PATCH /tarrafada/coupons/{id} Atualiza um cupom por ID
- Parameters:** A table with columns 'Name' and 'Description'. It lists a required parameter 'id' of type 'string (path)' with a text input field containing 'id'. A 'Try it out' button is visible in the top right.
- Request body:** A section labeled 'Request body required' with a dropdown menu set to 'application/json'.
- Example Value:** A section with tabs for 'Example Value' and 'Schema'. The 'Example Value' tab is active, showing a dark box with the JSON object {}.
- Responses:** A table with columns 'Code', 'Description', and 'Links'. It shows a response with code '200' and 'No links'.

Fonte: Próprio autor, 2025

4.11.8. Rota para excluir um cupom

A rota DELETE /tarrafada/coupons/{id} exclui um cupom específico, identificado pelo ID fornecido como parâmetro na URL.

Figura 51 - Rota para excluir um cupom

DELETE /tarrafada/coupons/{id} Exclui um cupom por ID

Try it out

Name	Description
id * required	id
string	
(path)	

Responses

Code	Description	Links
200		No links

Fonte: Próprio autor, 2025

4.11.9. Rota para retornar todos os cupons de um parceiro

A rota GET /tarrafada/coupons/partner/{id} retorna todos os cupons cadastrados por um parceiro, identificado pelo ID informado na URL.

Figura 52 - Rota para retornar todos os cupons de um parceiro

GET /tarrafada/coupons/partner/{id} Retorna todos os cupons do parceiro

Try it out

Name	Description
id * required	id
string	
(path)	

Responses

Code	Description	Links
200		No links

Fonte: Próprio autor, 2025

4.11.10. Rota para aceitar a solicitação e ativar um cupom

A rota PATCH `/tarrafada/coupons/active/{id}` ativa um cupom, definindo o campo `active_cou` como `true`. O cupom é identificado pelo ID informado na URL.

Figura 53 - Rota para aceitar a solicitação e ativar um cupom

The screenshot shows a REST client interface for a PATCH endpoint. The endpoint is `/tarrafada/coupons/active/{id}` with the description "Ativa um cupom (active_cou = true)". The parameters section shows a required path parameter `id` of type `string`. The responses section shows a 200 status code with the description "No links".

Fonte: Próprio autor, 2025

4.11.11. Rota para negar a solicitação e desativar o cupom

A rota DELETE `/tarrafada/coupons/remove/{id}` nega a solicitação de um cupom do sistema, identificado pelo ID informado na URL.

Figura 54 - Rota para negar a solicitação e desativar o cupom

The screenshot shows a REST client interface for a DELETE endpoint. The endpoint is `/tarrafada/coupons/remove/{id}` with the description "Remover um cupom". The parameters section shows a required path parameter `id` of type `string`. The request body section shows a required body of type `application/json` with an example value: `{ "motivo": "Motivo da remoção do cupom" }`. The responses section shows a 200 status code with the description "No links".

Fonte: Próprio autor, 2025

4.11.12. Rota para gerar um relatório de cupons

A rota GET /tarrafada/coupons/report/pdf gera um relatório de cupons em PDF, com base nos parâmetros obrigatórios date_in e date_out, e o parâmetros opcionais (status, category e partner).

Figura 55 - Rota para gerar um relatório de cupons

Name	Description
date_in * required	date_in
string (query)	
date_out * required	date_out
string (query)	
status	status
string (query)	
category	category
string (query)	
partner	partner
string (query)	

Code	Description	Links
200		No links

Fonte: Próprio autor, 2025

4.12. ROTAS DE AGENDA

4.12.1. Rota para cadastrar uma agenda

A rota POST /tarrafada/schedules cria uma nova agenda, recebendo no corpo da requisição os dados da agenda.

Figura 56 - Rota para cadastrar uma agenda

Parameters: No parameters

Request body required: application/json

```
{
  "id_sch": "string",
  "date_in_sch": "string",
  "date_out_sch": "string",
  "time_in_sch": "string",
  "time_out_sch": "string",
  "amount_sch": 0,
  "id_cou_fk": "string"
}
```


Code	Description	Links
201		No links

Fonte: Próprio autor, 2025

4.12.2. Rota para listar todas as agendas

A rota GET /tarrafada/schedules retorna todas as agendas, podendo receber parâmetros opcionais na query para paginação (page, limit) e filtro por texto (searchText).

Figura 57 - Rota para listar todas as agendas

The screenshot displays a REST client interface for the endpoint `GET /tarrafada/schedules` with the description "Retornar todas as agendas". The interface is divided into two main sections: "Parameters" and "Responses".

Parameters Section:

Name	Description
page	page
string (query)	
limit	limit
string (query)	
searchText	searchText
string (query)	

Responses Section:

Code	Description	Links
200		No links

Fonte: Próprio autor, 2025

4.12.3. Rota para buscar informações detalhadas de uma agenda

A rota GET /tarrafada/schedules/{id} retorna os dados de uma agenda específica, identificado pelo ID fornecido como parâmetro na URL.

Figura 58 - Rota para buscar informações detalhadas de uma agenda

The screenshot displays a REST client interface for the endpoint `GET /tarrafada/schedules/{id}` with the description "Retornar uma agenda por ID". The interface is divided into two main sections: "Parameters" and "Responses".

Parameters Section:

Name	Description
id * required	id
string (path)	

Responses Section:

Code	Description	Links
200		No links

Fonte: Próprio autor, 2025

4.12.4. Rota para atualizar uma agenda

A rota PATCH /tarrafada/schedules/{id} atualiza os dados de uma agenda específica, identificado pelo ID fornecido como parâmetro na URL.

Figura 59 - Rota para atualizar uma agenda

The screenshot shows a REST client interface for a PATCH endpoint. The URL is /tarrafada/schedules/{id} with the description 'Atualiza um cliente por ID'. The 'Parameters' section contains a required path parameter 'id' of type string. The 'Request body' is set to application/json and is currently empty. The 'Responses' section shows a 200 status code with no links.

Name	Description
id * required	id
string	(path)

Request body required: application/json

Example Value: {}

Code	Description	Links
200		No links

Fonte: Próprio autor, 2025

4.12.5. Rota para excluir uma agenda

A rota DELETE /tarrafada/schedules/{id} exclui um cupom específico, identificado pelo ID fornecido como parâmetro na URL.

Figura 60 - Rota para excluir uma agenda

The screenshot shows a REST client interface for a DELETE endpoint. The URL is /tarrafada/schedules/{id} with the description 'Exclui um cliente por ID'. The 'Parameters' section contains a required path parameter 'id' of type string. The 'Responses' section shows a 200 status code with no links.

Name	Description
id * required	id
string	(path)

Responses

Code	Description	Links
200		No links

Fonte: Próprio autor, 2025

4.12.6. Rota para gerar um relatório de agendas

A rota GET /tarrafada/schedules/report/pdf gera um relatório de agendas em PDF, com base nos parâmetros obrigatórios date_in e date_out, e o parâmetros opcionais (status, coupon e partner).

Figura 61 - Rota para gerar um relatório de agendas

Name	Description
date_in * required	date_in
string (query)	
date_out * required	date_out
string (query)	
status	status
string (query)	
coupon	coupon
string (query)	
partner	partner
string (query)	

Code	Description	Links
200		No links

Fonte: Próprio autor, 2025

4.13. ROTAS DE FAVORITO

4.13.1. Rota para cadastrar um favorito

A rota POST /tarrafada/favorites cria um favorito, recebendo no corpo da requisição os dados da agenda.

Figura 62 - Rota para cadastrar um favorito

Parameters: No parameters

Request body required: application/json

```
{
  "id_fav": "string",
  "favorite_fav": true,
  "id_coo_fk": "string",
  "id_cus_fk": "string"
}
```

Code	Description	Links
201		No links

Fonte: Próprio autor, 2025

4.13.2. Rota para retornar todos os favoritos por cliente

A rota GET /tarrafada/favorites/{id_cus} retorna todos os cupons favoritados por um cliente, identificado pelo id_cus (obrigatório) na URL. Aceita os parâmetros opcionais page, limit e searchText para controle de paginação e filtro.

Figura 63 - Rota para retornar todos os favoritos por cliente

The screenshot shows a REST client interface for the endpoint `GET /tarrafada/favorites/{id_cus}`. The interface is divided into two main sections: Parameters and Responses.

Parameters:

Name	Description
id_cus * required string (path)	id_cus
page string (query)	page
limit string (query)	limit
searchText string (query)	searchText

Responses:

Code	Description	Links
200		No links

Fonte: Próprio autor, 2025

4.13.3. Rota para excluir uma agenda

A rota DELETE /tarrafada/schedules/{id} exclui um cupom específico, identificado pelo ID fornecido como parâmetro na URL.

Figura 64 - Rota para excluir uma agenda

The screenshot shows a REST client interface for the endpoint `DELETE /tarrafada/schedules/{id}`. The interface is divided into two main sections: Parameters and Responses.

Parameters:

Name	Description
id * required string (path)	id

Responses:

Code	Description	Links
200		No links

Fonte: Próprio autor, 2025

4.13.4. Rota para gerar um relatório de favoritos

A rota GET /tarrafada/favorites/report/pdf gera um relatório de agendas em PDF, com base nos parâmetros obrigatórios date_in e date_out, e o parâmetros opcionais (status, coupon e partner).

Figura 65 - Rota para gerar um relatório de favoritos

Name	Description
date_in * required	date_in
string (query)	
date_out * required	date_out
string (query)	
customer	customer
string (query)	
partner	partner
string (query)	
coupon	coupon
string (query)	

Code	Description	Links
200		No links

Fonte: Próprio autor, 2025

4.14. ROTAS DE CARRINHO

4.14.1. Rota para adicionar cupom no carrinho

A rota POST /tarrafada/carts/add adiciona um cupom ao carrinho de compras. Recebe no corpo da requisição os campos id_cus (cliente), id_cou (cupom) e id_sch (horário/agendamento).

Figura 66 - Rota para adicionar cupom no carrinho

Cart

POST /tarrafada/carts/add Adicionar cupom ao carrinho

Parameters: No parameters

Request body required: application/json

Example Value

```
{
  "id_cus": "string",
  "id_cou": "string",
  "id_sch": "string"
}
```

Responses

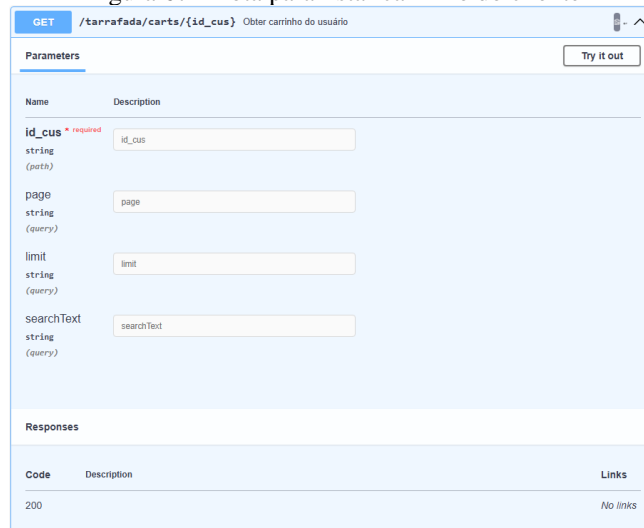
Code	Description	Links
201		No links

Fonte: Próprio autor, 2025

4.14.2. Rota para listar carrinho do cliente

A rota GET /tarrafada/carts/{id_cus} retorna os cupons presentes no carrinho de um cliente, identificado pelo id_cus na URL. Aceita os parâmetros opcionais page, limit e searchText para paginação e filtro.

Figura 67 - Rota para listar carrinho do cliente

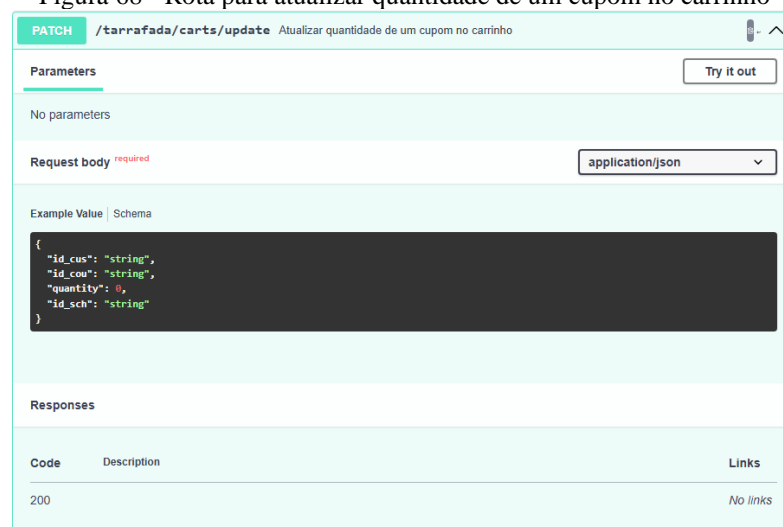


Fonte: Próprio autor, 2025

4.14.3. Rota para atualizar quantidade de um cupom no carrinho

A rota PATCH /tarrafada/carts/update atualiza a quantidade de um cupom no carrinho. Recebe no corpo da requisição os campos id_cus (cliente), id_cou (cupom), quantity e id_sch (horário/agendamento).

Figura 68 - Rota para atualizar quantidade de um cupom no carrinho



Fonte: Próprio autor, 2025

4.14.4. Rota para remover um cupom do carrinho

A rota DELETE /tarrafada/carts/remove remove um cupom do carrinho. Recebe no corpo da requisição os campos id_cus (cliente) e id_cou (cupom).

Figura 69 - Rota para remover um cupom do carrinho

DELETE /tarrafada/carts/remove Remover cupom do carrinho

Parameters Try it out

No parameters

Request body ^{required} application/json

Example Value | Schema

```
{
  "id_cus": "string",
  "id_cou": "string"
}
```

Responses

Code	Description	Links
200		No links

Fonte: Próprio autor, 2025

4.15. ROTAS DE VENDA

4.15.1. Rota para cadastrar uma venda

A rota POST /tarrafada/sales cria uma venda, recebendo no corpo da requisição os dados da venda.

Figura 70 - Rota para cadastrar uma venda

Sales

POST /tarrafada/sales Cria uma venda

Parameters Try it out

No parameters

Request body ^{required} application/json

Example Value | Schema

```
{
  "id_car_fk": "string",
  "id_payment_sal": "string",
  "total_sal": 0,
  "status_sal": "string",
  "status_details_sal": "string",
  "date_sal": "string",
  "time_sal": "string"
}
```

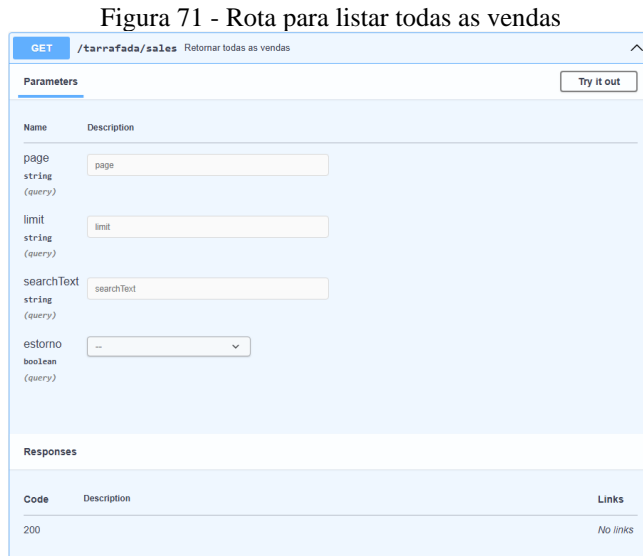
Responses

Code	Description	Links
201		No links

Fonte: Próprio autor, 2025

4.15.2. Rota para listar todas as vendas

A rota GET /tarrafada/sales retorna todas as vendas, podendo receber parâmetros opcionais na query para paginação (page, limit) e filtro por texto (searchText).



Fonte: Próprio autor, 2025

4.15.3. Rota para compra rápida de um cupom

A rota POST /tarrafada/sales/quick-buy realiza a compra rápida de um cupom por um cliente. Recebe no corpo da requisição os campos id_cus (cliente), id_cou (cupom), id_sch (horário/agendamento) e quantity.



Fonte: Próprio autor, 2025

4.15.4. Rota para gerar um relatório de vendas

A rota GET /tarrafada/sales/report/pdf gera um relatório de vendas em PDF, com base nos parâmetros obrigatórios date_in e date_out, e o parâmetros opcionais (status, customer e partner).

Figura 73 - Rota para gerar um relatório de vendas

The screenshot displays the Swagger UI for the endpoint GET /tarrafada/sales/report/pdf. The interface is divided into two main sections: Parameters and Responses. The Parameters section contains a table with columns for Name and Description. The parameters listed are: status (string, optional), date_in (string, required, format YYYY-MM-DD), date_out (string, required, format YYYY-MM-DD), partner (string, optional), and customer (string, optional). Each parameter has a corresponding input field. The Responses section shows a table with columns for Code, Description, and Links. A single response is listed with a 200 status code and 'No links'.

Name	Description
status	Status da venda
string (query)	status
date_in * required	Data inicial (AAAA-MM-DD)
string (query)	date_in
date_out * required	Data final (AAAA-MM-DD)
string (query)	date_out
partner	Parceiro
string (query)	partner
customer	Cliente
string (query)	customer

Code	Description	Links
200		No links

Fonte: Próprio autor, 2025

4.15.5. Rota para buscar venda pelo id do carrinho

A rota GET /tarrafada/sales/cart/{id} busca os detalhes de uma venda e os cupons comprados, a partir do ID do carrinho informado na URL.

Figura 74 - Rota para buscar venda pelo id do carrinho

The screenshot displays the Swagger UI for the endpoint GET /tarrafada/sales/cart/{id}. The interface is divided into two main sections: Parameters and Responses. The Parameters section contains a table with columns for Name and Description. A single parameter is listed: id (string, required, path). The Responses section shows a table with columns for Code, Description, and Links. A single response is listed with a 200 status code and 'No links'.

Name	Description
id * required	ID do carrinho
string (path)	id

Code	Description	Links
200		No links

Fonte: Próprio autor, 2025

4.15.6. Rota para buscar informações da venda

A rota GET /tarrafada/sales/{id} retorna os detalhes de uma venda, incluindo informações do carrinho e dos cupons, a partir do ID da venda informado na URL.

Figura 75 - Rota para buscar informações da venda

The screenshot shows a REST client interface for the endpoint GET /tarrafada/sales/{id}. The interface includes a 'Parameters' section with a table listing the 'id' parameter as a required string (path). Below this is a 'Responses' section with a table showing a 200 status code and 'No links'.

Name	Description
id * required	ID da venda
string (path)	id

Code	Description	Links
200		No links

Fonte: Próprio autor, 2025

4.16. ROTAS DE RECEBIMENTO

4.16.1. Rota para cadastrar um recebimento

A rota POST /tarrafada/receipts cria um recebimento, recebendo no corpo da requisição os dados do recebimento.

Figura 76 - Rota para cadastrar um recebimento

The screenshot shows a REST client interface for the endpoint POST /tarrafada/receipts. The interface includes a 'Request body' section with a dropdown menu set to 'application/json' and an 'Example Value' section displaying a JSON object. Below this is a 'Responses' section with a table showing a 201 status code and 'Recebimento criado com sucesso.'.

```
{
  "id_soa_fk": "string",
  "status_rec": "Pendente",
  "date_rec": "string",
  "time_rec": "string",
  "installment_rec": "string",
  "payment_method_rec": "string"
}
```


Code	Description	Links
201	Recebimento criado com sucesso.	No links

Fonte: Próprio autor, 2025

4.16.2. Rota para listar todos os recebimentos

A rota GET /tarrafada/receipts retorna todos os recebimentos, podendo receber parâmetros opcionais na query para paginação (page, limit), filtro por texto (searchText) e (status).

Figura 77 - Rota para listar todos os recebimentos

The screenshot shows a REST client interface for the endpoint GET /tarrafada/receipts. The interface is divided into two main sections: Parameters and Responses.

Parameters:

Name	Description
page	page
string (query)	
limit	limit
string (query)	
searchText	searchText
string (query)	
status	status
any (query)	

Responses:

Code	Description	Links
200		No links

Fonte: Próprio autor, 2025

4.16.3. Rota para buscar informações detalhadas de um recebimento

A rota GET /tarrafada/receipts/{id} retorna os dados de um recebimento específico, identificado pelo ID fornecido como parâmetro na URL.

Figura 78 - Rota para buscar informações detalhadas de um recebimento

The screenshot shows a REST client interface for the endpoint GET /tarrafada/receipts/{id}. The interface is divided into two main sections: Parameters and Responses.

Parameters:

Name	Description
id * required	id
string (path)	

Responses:

Code	Description	Links
200	Recebimento encontrado.	No links

Fonte: Próprio autor, 2025

4.16.4. Rota para gerar um relatório de recebimentos

A rota GET /tarrafada/receipts/report/pdf gera um relatório de recebimentos em PDF, com base nos parâmetros obrigatórios date_in e date_out, e o parâmetros opcionais (status, customer e partner).

Figura 79 - Rota para gerar um relatório de recebimentos

The screenshot displays a REST client interface for the endpoint `GET /tarrafada/receipts/report/pdf`. The interface is divided into two main sections: **Parameters** and **Responses**.

Parameters: This section lists five query parameters, each with a text input field:

- `status` (string, query): Input field with value "status".
- `date_in` (string, query, required): Input field with value "date_in".
- `date_out` (string, query, required): Input field with value "date_out".
- `partner` (string, query): Input field with value "partner".
- `customer` (string, query): Input field with value "customer".

Responses: This section shows a table with one row:

Code	Description	Links
200		No links

Fonte: Próprio autor, 2025

4.17. ROTAS DE PAGAMENTO (Mercado Pago)

4.17.1. Rota para gerar link de pagamento

A rota GET /tarrafada/payments/link gera um link de pagamento para um carrinho. Requer o parâmetro id_car_fk (ID do carrinho) na query.

Figura 80 - Rota para gerar link de pagamento

The screenshot displays a REST client interface for the endpoint `GET /tarrafada/payments/link`. The interface is divided into two main sections: **Parameters** and **Responses**.

Parameters: This section lists one query parameter with an input field:

- `id_car_fk` (string, query, required): Input field with value "id_car_fk".

Responses: This section shows a table with one row:

Code	Description	Links
200		No links

Fonte: Próprio autor, 2025

4.17.2. Rota para consultar uma ordem de compra

A rota GET `/tarrafada/payments/merchant-order/{id}` consulta os detalhes de uma ordem de pagamento, identificada pelo ID da ordem do comerciante informado na URL.

Figura 81 - Rota para consultar uma ordem de compra

GET `/tarrafada/payments/merchant-order/{id}` Consultar detalhes de uma ordem

Parameters Try it out

Name	Description
id * required	ID da ordem do comerciante
number (path)	<input type="text" value="id"/>

Responses

Code	Description	Links
200		No links

Fonte: Próprio autor, 2025

4.17.3. Rota para consultar detalhes do pagamento

A rota GET `/tarrafada/payments/details/{id}` consulta os detalhes de um pagamento, utilizando o ID da ordem do comerciante informado na URL.

Figura 82 - Rota para consultar detalhes do pagamento

GET `/tarrafada/payments/details/{id}` Consultar detalhes de uma pagamento

Parameters Try it out

Name	Description
id * required	ID da ordem do comerciante
string (path)	<input type="text" value="id"/>

Responses

Code	Description	Links
200		No links

Fonte: Próprio autor, 2025

4.17.4. Rota para consultar detalhes do pagamento

A rota POST `/tarrafada/payments/refund/{id_rec}` realiza o estorno de um pagamento, utilizando o ID do recibo (`id_rec`) informado na URL.

Figura 83 - Rota para consultar detalhes do pagamento

POST `/tarrafada/payments/refund/{id_rec}` Realizar estorno de pagamento

Parameters Try it out

Name	Description
id_rec * required	ID do recibo para estorno
string (path)	<input type="text" value="id_rec"/>

Responses

Code	Description	Links
201		No links

Fonte: Próprio autor, 2025

4.17.5. Rota para receber notificações de pagamento pelo webhook

A rota POST `/tarrafada/payments/webhook` recebe notificações de pagamento via webhook. Aceita os parâmetros de query `type` (tipo de notificação) e `id` (ID do pagamento).

Figura 84 - Rota para receber notificações de pagamento pelo webhook

POST `/tarrafada/payments/webhook` Webhook para receber notificações de pagamento

Parameters Try it out

Name	Description
type	Tipo de notificação
any (query)	<input type="text" value="type"/>
id	ID do pagamento
any (query)	<input type="text" value="id"/>

Responses

Code	Description	Links
201		No links

Fonte: Próprio autor, 2025

4.18. ROTAS DE VOUCHER

4.18.1. Rota para gerar um relatório de voucher

A rota GET `/tarrafada/voucher/report/pdf` gera um relatório de vouchers em PDF, com base nos parâmetros obrigatórios `date_in` e `date_out`, e o parâmetro opcional `status`.

Figura 85 - Rota para gerar um relatório de voucher

Voucher

GET /tarrafada/voucher/report/pdf Gera um relatório de vouchers em PDF

Parameters Try it out

Name	Description
date_in * required string (query)	date_in
date_out * required string (query)	date_out
status string (query)	status

Responses

Code	Description	Links
200		No links

Fonte: Próprio autor, 2025

4.18.2. Rota para gerar um voucher para a venda

A rota GET `/tarrafada/voucher/report/{saleId}` gera um voucher em PDF referente a uma venda, identificada pelo `saleId` informado na URL.

Figura 86 - Rota para gerar um voucher para a venda

GET /tarrafada/voucher/report/{saleId} Gera um voucher em PDF para a venda

Parameters Try it out

Name	Description
saleId * required string (path)	ID da venda saleId

Responses

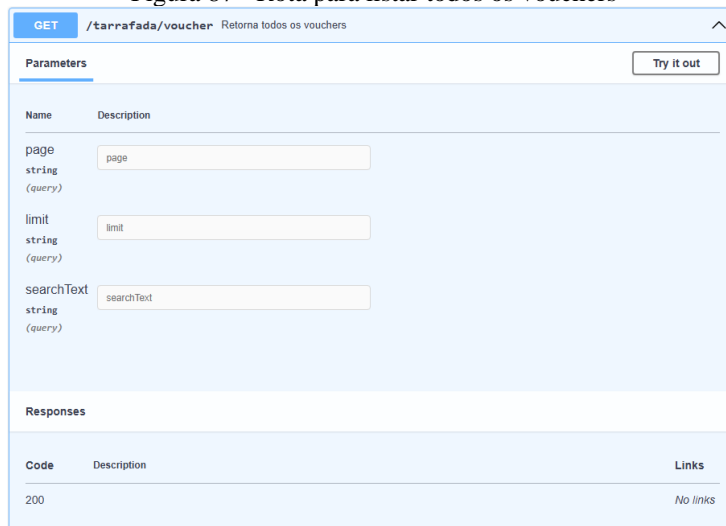
Code	Description	Links
200		No links

Fonte: Próprio autor, 2025

4.18.3. Rota para listar todos os vouchers

A rota GET /tarrafada/vouchers retorna todos os vouchers, podendo receber parâmetros opcionais na query para paginação (page, limit) e filtro por texto (searchText).

Figura 87 - Rota para listar todos os vouchers

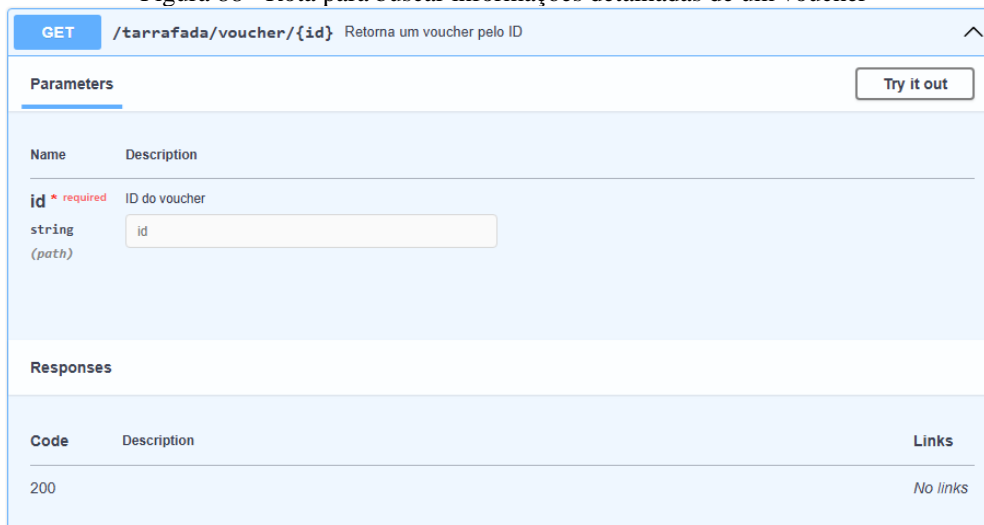


Fonte: Próprio autor, 2025

4.18.4. Rota para buscar informações detalhadas de um voucher

A rota GET /tarrafada/voucher/{id} retorna os dados de um voucher específico, identificado pelo ID fornecido como parâmetro na URL.

Figura 88 - Rota para buscar informações detalhadas de um voucher



Fonte: Próprio autor, 2025

4.18.5. Rota para validar o voucher

A rota POST /tarrafada/voucher/validate valida um voucher com base no CPF do cliente e no código do voucher, enviados no corpo da requisição.

Figura 89 - Rota para validar o voucher

The screenshot shows a Swagger UI interface for the endpoint POST /tarrafada/voucher/validate. The description is 'Valida um voucher pelo CPF do cliente e código do voucher'. The 'Parameters' section is empty. The 'Request body' is required and set to 'application/json'. An example value is shown as a JSON object: { "cpf": "12345678901", "code": "ABC12345" }. The 'Responses' section shows a 201 status code with no links.

Code	Description	Links
201		No links

Fonte: Próprio autor, 2025

As rotas apresentadas demonstram o funcionamento do back-end do sistema, evidenciando a estrutura organizada da API e sua capacidade de atender às principais necessidades do sistema. Essa abordagem orientada a serviços facilita a integração com o front-end e garante escalabilidade, segurança e clareza no desenvolvimento. A documentação via Swagger contribui ainda mais para a manutenção e evolução contínua da aplicação.

A aplicação dos conhecimentos adquiridos ao longo do curso resultou no desenvolvimento do back-end da API RESTful do sistema, voltado à venda de cupons. O sistema foi projetado para garantir escalabilidade, segurança e organização, utilizando boas práticas de desenvolvimento e tecnologias modernas.

Para evidenciar os resultados, foram incluídas imagens da documentação gerada automaticamente por meio do Swagger, que permite visualizar e testar as funcionalidades disponíveis. Cada rota corresponde a uma funcionalidade da aplicação, como autenticação, gerenciamento de cupons, carrinho de compras, geração de vouchers, entre outras.

A estrutura da API foi pensada para facilitar a integração com o front-end e permitir futuras expansões, como novas funcionalidades ou integração com sistemas de terceiros (ex.: gateways de pagamento, sistemas de CRM, etc.). Além disso, a documentação clara e o uso de parâmetros bem definidos contribuem para a manutenção do código e o trabalho colaborativo

em equipe.

Do ponto de vista científico e tecnológico, este produto contribui para a discussão sobre a aplicação prática de conceitos de engenharia de software, arquitetura orientada a serviços (SOA), segurança de APIs e desenvolvimento back-end em ambientes reais. O sistema se apresenta como um caso aplicável a outras plataformas que demandam funcionalidades similares, como e-commerces, marketplaces e sistemas de vouchers.

Por fim, os testes práticos demonstraram a robustez e funcionalidade do back-end, validando os requisitos propostos inicialmente e evidenciando o sucesso no desenvolvimento da solução.

5. CONCLUSÃO

Esta pesquisa teve como objetivo principal o desenvolvimento do back-end de um sistema de e-commerce voltado à comercialização de cupons promocionais, com foco inicial em Florianópolis (SC) e potencial de alcance nacional. Esse objetivo foi plenamente alcançado por meio da implementação de uma arquitetura baseada em serviços (SOA - *Service-Oriented Architecture*), utilizando tecnologias modernas e práticas consolidadas de desenvolvimento de software.

Os objetivos específicos também foram atendidos com êxito. A estruturação das rotas RESTful foi realizada com organização e clareza, garantindo a escalabilidade e a manutenção do sistema. Práticas de segurança foram implementadas, incluindo autenticação e controle de acesso, assegurando a proteção dos dados trafegados pela API. A documentação automatizada da API foi gerada com o uso do Swagger, o que facilitou os testes, a validação e a comunicação com o front-end. A arquitetura modular adotada permitiu uma integração fluida com a interface e estabeleceu uma base sólida para futuras evoluções do sistema.

O processo de desenvolvimento, conduzido com base na metodologia Scrum, proporcionou uma gestão eficiente do tempo e das funcionalidades. A integração com o front-end e os testes realizados demonstraram a robustez das funcionalidades desenvolvidas, como o gerenciamento de cupons, o carrinho de compras, a autenticação de usuários e a emissão de vouchers, além da integração com serviços externos como Mercado Pago e ViaCEP.

Como resultado, o sistema apresenta-se como uma solução funcional, segura e adaptável, capaz de apoiar micro e pequenos empreendedores locais na divulgação e venda de cupons promocionais. Do ponto de vista científico e tecnológico, este trabalho contribui para a prática acadêmica ao apresentar uma aplicação concreta dos conceitos de APIs RESTful, segurança da informação e arquitetura de software moderna no contexto de e-commerces.

Para trabalhos futuros, prevê-se a implantação do sistema em ambiente de produção, com parcerias comerciais locais, permitindo a validação do sistema em uso real. Essa etapa possibilitará a coleta de feedbacks dos usuários, a identificação de melhorias e a expansão funcional da plataforma, visando o aperfeiçoamento contínuo do produto e sua consolidação como ferramenta de apoio ao comércio regional.

6. REFERÊNCIAS

AGÊNCIA GOV. **E-commerce no Brasil cresce 4% e alcança R\$ 196 bi em 2023**. Agência Gov, 2024. Disponível em: <https://agenciagov.ebc.com.br/noticias/202409/e-commerce-no-brasil-cresce-4-e-alcanca-r-196-bi-em-2023>. Acesso em: 27 jun. 2025.

AMAZON WEB SERVICES. **O que é Scrum?**. AWS, 2024. Disponível em: <https://aws.amazon.com/pt/what-is/scrum>. Acesso em: 17 set. 2024.

ANTONIO, N., FORNAZIN, M., ARAUJO, R. (2018). **Metodologia de Pesquisa de Estudo de Caso em Sistemas de Informação**. V Escola Regional de Sistemas de Informação do Rio de Janeiro, Vol. 1, p. 67, Dezembro de 2018.

CUNHA, Fernando. **Sistema Web: o que é e como funciona?**. Mestre da Web, 2023. Disponível em: <https://www.mestresdawe.com.br/tecnologias/sistema-web-o-que-e-e-como-funciona>. Acesso em: 30 ago. 2024.

FOCUSKEYWORDFINDER. **Marketing Digital: Os Principais Benefícios e Vantagens para Pequenas Empresas**. Focus Keyword Finder, 2023. Disponível em: <https://focuskeywordfinder.com/pt/blog/marketing-digital/marketing-digital-os-principais-beneficios-e-vantagens-para-pequenas-empresas>. Acesso em: 27 jun. 2025.

GADO, Wesley. **Guia do framework NestJS**. Treina Web, 2024. Disponível em: <https://www.treinaweb.com.br/blog/guia-do-framework-nestjs>. Acesso em: 29 ago. 2024.

GOIÁS.GOV. **Comércio eletrônico representa 9% do varejo brasileiro**. Governo de Goiás, 2024. Disponível em: <https://www.goias.gov.br>. Acesso em: 27 jun. 2025.

HARVESTO. **Importância do marketing digital para pequenas empresas**. Harvesto, 2024. Disponível em: <https://harvesto.com.br/importancia-do-marketing-digital-para-pequenas-empresas/>. Acesso em: 27 jun. 2025.

MATA, K. **E-Commerce: Análise de dados sobre o comércio eletrônico no Brasil**. PUC Goiás, Goiânia, 2021. Disponível em: <https://repositorio.pucgoias.edu.br/jspui/bitstream/123456789/1761/1/E-commerce%20-%20An%C3%A1lise%20de%20Dados%20sobre%20o%20Com%C3%A9rcio%20Eletr%C3%B4nico%20no%20Brasil.pdf>. Acesso em: 20 ago. 2024.

MENDONÇA, H. E-commerce. **IPTEC – Revista Inovação, Projetos e Tecnologias**, Rio de Janeiro, Vol. 4, p. 251, Outubro de 2016.

MENEZES, Pedro. **Estudo de Caso: o que é, exemplos e como fazer**. Enciclopédia Significados, 2024. Disponível em: <https://www.significados.com.br/estudo-de-caso/>. Acesso em 17 de set. 2024.

MERCADO PAGO. **Documentação oficial da API**. Mercado Pago Developers, 2024. Disponível em: <https://www.mercadopago.com.br/developers/pt>. Acesso em: 10 jan. 2025.

MINDMASTER. **Scrum: A Metodologia Ágil Explicada de forma Definitiva**. MindMaster, 2024. Disponível em: <https://mindmaster.com.br/scrum/>. Acesso em: 17 set. 2024.

MO4 WEB. **5 vantagens do marketing digital: entenda sobre a estratégia**. MO4 Web, 2024. Disponível em: <https://mo4web.com.br/blog/5-vantagens-do-marketing-digital-entenda-sobre-a-estrategia/>. Acesso em: 27 jun. 2025.

MZCLICK. **Benefícios do Marketing Digital**. MZClick, 2024. Disponível em: <https://napoleon.mzclick.com.br/beneficios-do-marketing-digital/>. Acesso em: 27 jun. 2025.

MYSQL. **Why MySQL**. Oracle, 2024. Disponível em: <https://www.mysql.com/why-mysql/>. Acesso em: 29 ago. 2024.

NESTJS. **Official Documentation**. NestJS, 2024. Disponível em: <https://docs.nestjs.com/>. Acesso em: 25 ago. 2024.

NIELSENIQ. **A trajetória ascendente do e-commerce brasileiro**. NielsenIQ, 2024. Disponível em: <https://nielseniq.com/global/pt/insights/education/2024/a-trajetoria-ascendente-do-e-commerce-brasileiro/>. Acesso em: 27 jun. 2025.

OPENAPI INITIATIVE. **Swagger Tools**. Swagger, 2024. Disponível em: <https://swagger.io/tools/>. Acesso em: 10 jan. 2025.

ORACLE CORPORATION. **MySQL Documentation**. Oracle, 2024. Disponível em: <https://dev.mysql.com/doc/>. Acesso em: 25 ago. 2024.

PODER360. **Faturamento do e-commerce brasileiro chega a R\$ 196 bi em 2023**. Poder360, 2024. Disponível em: <https://www.poder360.com.br>. Acesso em: 27 jun. 2025.

PRISMA. **Prisma ORM – Next-generation Node.js and TypeScript ORM**. Prisma, 2024. Disponível em: <https://www.prisma.io>. Acesso em: 21 jun. 2025.

SEBRAE. **Marketing digital para iniciantes: o que é e como funciona**. Sebrae, 2024. Disponível em: <https://sebrae.com.br/sites/PortalSebrae/artigos/marketing-digital-para-iniciantes-o-que-e-e-como-funciona,9f1ac3399eeab810VgnVCM1000001b00320aRCRD>. Acesso em: 30 ago. 2024.

SEBRAE. **Como estruturar um e-commerce local eficiente**. Sebrae, 2024. Disponível em: <https://digital.sebraers.com.br/blog/marketing-e-vendas/como-estruturar-um-e-commerce-local-eficiente/>. Acesso em: 27 jun. 2025.

VIACEP. **API gratuita para consulta de CEPs**. ViaCEP, 2024. Disponível em: <https://viacep.com.br/>. Acesso em: 10 jan. 2025.

APÊNDICE A: Projeto do Software (**Obrigatório**)

7. ANEXO I: Título do Anexo (Opcional)

**INSTITUTO FEDERAL DE EDUCAÇÃO, CIÊNCIAS E TECNOLOGIA DE
RONDÔNIA – CAMPUS JI-PARANÁ**

**CURSO SUPERIOR DE TECNOLOGIA EM ANÁLISE E DESENVOLVIMENTO DE
SISTEMAS**

SISTEMA E-COMMERCE DE VENDA DE CUPONS

Mateus França da Cunha
Tiago Rafael Oliveira das Virgens

Ji-Paraná - RO, 2024

Mateus França da Cunha
Tiago Rafael Oliveira das Virgens

SISTEMA E-COMMERCE DE VENDA DE CUPONS

Projeto de Conclusão de Curso SUPERIOR EM ANÁLISE E DESENVOLVIMENTO DE SISTEMAS, como parte dos requisitos necessários para a aprovação em Tecnólogo em Análise e Desenvolvimento de Sistemas. Área de Concentração: Ciências Exatas e da Terra.

Orientador (a): Prof. Ms. Jackson Henrique da Silva Bezerra

LISTA DE FIGURAS

Figura 1 - Requisitos de Qualidade Interna e Externa.....	11
Figura 2 - Pilares do scrum.....	18
Figura 3 - Papéis no scrum.	19
Figura 4 - Modelo de sprint.	20
Figura 5 - Ciclo de Vida do Scrum.....	20
Figura 6 – Diagrama de Caso de Uso.....	80
Figura 7 - Diagrama de Sequência - Acessar Login Cliente	121
Figura 8 - Diagrama de Sequência - Recuperar Senha Cliente	121
Figura 9 - Diagrama de Sequência - Redefinir Senha Cliente.....	122
Figura 10 - Diagrama de Sequência - Alterar Senha Cliente	122
Figura 11 - Diagrama de Sequência - Cadastrar Categorias.....	122
Figura 12 - Diagrama de Sequência - Consultar Categorias	123
Figura 13 - Diagrama de Sequência - Acessar Menu Inicial Cliente	123
Figura 14 - Diagrama de Sequência - Comprar Cupons.....	124
Figura 15 - Diagrama de Sequência - Emitir Relatório de Categorias	124

LISTA DE TABELAS

Tabela 1 - Custo por Mês.	16
Tabela 2 - Custo Total do Projeto.....	16
Tabela 3 - Itens do Product Backlog.....	22

LISTA DE QUADROS

Quadro 1 - Papeis e atribuições da equipe de desenvolvimento.	17
Quadro 2 – Descrição das tarefas do projeto.....	22
Quadro 3 – Reuniões das Sprints.	30
Quadro 4 - Acessar Login Cliente.....	37
Quadro 5 - Acessar Login Parceiro.	37
Quadro 6 - Acessar Login Administrador.	38
Quadro 7 - Redefinir Senha Cliente.	39
Quadro 8 - Redefinir Senha Parceiro.	40
Quadro 9 - Redefinir Senha Administrador.	40
Quadro 10 - Recuperar Senha Cliente.....	41
Quadro 11 - Recuperar Senha Parceiro.	42
Quadro 12 - Recuperar Senha Administrador.....	42
Quadro 13 - Alterar Senha Cliente.....	43
Quadro 14 - Alterar Senha Parceiro.	44
Quadro 15 - Alterar Senha Administrador.	45
Quadro 16 - Acessar Tela Inicial Cliente.....	46
Quadro 17 - Acessar Menu Inicial Parceiro.....	46
Quadro 18 - Acessar Menu Inicial Administrador.....	47
Quadro 19 - Cadastrar Categorias.....	47
Quadro 20 - Cadastrar Clientes.....	48
Quadro 21 - Cadastrar Parceiros.....	49
Quadro 22 - Cadastrar Cupons.....	50
Quadro 23 - Cadastrar Agendas.....	51
Quadro 24 - Favoritar Cupons.....	52
Quadro 25 -Comprar Cupons.....	52
Quadro 26 - Receber Vendas.....	53
Quadro 27 - Emitir Voucher.....	54
Quadro 28 - Validar Voucher.....	54
Quadro 29 - Estornar Venda.....	55
Quadro 30 - Consultar Favoritos.....	56
Quadro 31 - Consultar Categorias.....	57
Quadro 32 -Consultar Clientes.....	58

Quadro 33 -Consultar Parceiros.....	59
Quadro 34 -Consultar Cupons.....	60
Quadro 35 - Consultar Agendas.....	60
Quadro 36 - Consultar Vendas.....	61
Quadro 37 -Consultar Recebimentos.....	62
Quadro 38 -Consultar Vouchers.....	63
Quadro 39 - Consultar Solicitação de Cupons.....	64
Quadro 40 - Consultar Solicitação de Parceiros.....	65
Quadro 41 - Autorizar Solicitação de Parceiros.....	66
Quadro 42 - Autorizar Solicitação de Cupons.....	67
Quadro 43 - Emitir Relatório de Vendas.....	68
Quadro 44 - Emitir Relatório de Vouchers.....	69
Quadro 45 - Emitir Relatório de Categorias.....	69
Quadro 46 - Emitir Relatório de Clientes.....	70
Quadro 47 - Emitir Relatório de Parceiros.....	71
Quadro 48 -Emitir Relatório de Cupons.....	72
Quadro 49 - Emitir Relatório de Agendas.....	73
Quadro 50 - Emitir Relatório de Favoritos.....	74
Quadro 51 - Emitir Relatório de Recebimentos.....	75
Quadro 52 - Gerar Dashboard de Parceiros.....	76
Quadro 53 - Gerar Dashboard de Administrador.....	76
Quadro 54 - Agendar.....	77

SUMÁRIO

1. DOCUMENTO DE ESPECIFICAÇÃO DO PROJETO.....	10
1.1 Dados da Empresa Cliente.....	10
1.2 Sobre o Produto de <i>Software</i>	10
1.3 Requisitos de Qualidade	11
1.4 Aspectos Técnicos do Produto	12
1.4.1 Arquitetura de Rede.....	13
1.4.2 Tecnologias <i>Front End</i>	14
1.4.3 Tecnologias Back End	15
1.5 Objetivos Gerenciais do Projeto	15
1.6 Equipe de Desenvolvimento	17
1.7 Metodologia de Desenvolvimento.....	18
1.7.1 Sobre a Metodologia de Pesquisa.....	18
1.7.2. Ciclo de Vida da Metodologia no Projeto do Sistema	21
1.8 Entrevistas	32
2 DOCUMENTO DE REQUISITOS.....	37
3 PROTÓTIPO DO SISTEMA	79
4 CASO DE USO	80
4.1 Diagrama de Caso de Uso	80
4.2 Caso de Uso Expandido.....	81
4.2.1Caso de Uso – Acessar Login Cliente	81
4.2.2 Caso de Uso – Acessar Login Parceiro	81
4.2.3 Caso de Uso – Acessar Login Administrador	82
4.2.4 Caso de Uso – Redefinir Senha Cliente	82
4.2.5 Caso de Uso – Redefinir Senha Parceiro.....	82
4.2.6 Caso de Uso – Redefinir Senha Administrador.....	83
4.2.7 Caso de Uso – Recuperar Senha Cliente	83

4.2.8 Caso de Uso – Recuperar Senha Parceiro	84
4.2.9 Caso de Uso – Recuperar Senha Administrador	84
4.2.10 Caso de Uso – Alterar Senha Cliente	85
4.2.11 Caso de Uso – Alterar Senha Parceiro.....	85
4.2.12 Caso de Uso – Alterar Senha Administrador.....	86
4.2.13 Caso de Uso – Acessar Tela Inicial Cliente	86
4.2.14 Caso de Uso – Acessar Menu Inicial Parceiro	87
4.2.15 Caso de Uso – Acessar Menu Inicial Administrador	87
4.2.16 Caso de Uso – Cadastrar Categorias	87
4.2.17 Caso de Uso – Cadastrar Clientes	88
4.2.18 Caso de Uso – Cadastrar Parceiros.....	89
4.2.19 Caso de Uso – Cadastrar Cupons	89
4.2.20 Caso de Uso – Cadastrar Agendas.....	90
4.2.21 Caso de Uso – Favoritar Cupons	90
4.2.22 Caso de Uso – Comprar Cupons.....	91
4.2.23 Caso de Uso – Receber Vendas	91
4.2.24 Caso de Uso – Emitir Voucher	92
4.2.25 Caso de Uso – Validar Vouchers.....	93
4.2.26 Caso de Uso – Estornar Vendas	93
4.2.27 Caso de Uso – Consultar Favoritos	94
4.2.28 Caso de Uso – Consultar Categorias	94
4.2.29 Caso de Uso – Consultar Clientes	95
4.2.30 Caso de Uso – Consultar Parceiros.....	96
4.2.31 Caso de Uso – Consultar Cupons	98
4.2.32 Caso de Uso – Consultar Agendas	99
4.2.33 Caso de Uso – Consultar Vendas	100
4.2.34 Caso de Uso – Consultar Recebimentos.....	101

4.2.35 Caso de Uso – Consultar Vouchers	101
4.2.36 Caso de Uso – Consultar Solicitação de Cupons.....	102
4.2.37 Caso de Uso – Consultar Solicitação de Parceiros	103
4.2.38 Caso de Uso – Autorizar Solicitação de Parceiros	104
4.2.39 Caso de Uso – Autorizar Solicitação de Cupons	105
4.2.40 Caso de Uso – Emitir Relatório de Vendas	106
4.2.41 Caso de Uso – Emitir Relatório de Vouchers.....	107
4.2.42 Caso de Uso – Emitir Relatório de Categorias	109
4.2.43 Caso de Uso – Emitir Relatório de Clientes	110
4.2.44 Caso de Uso – Emitir Relatório de Parceiros	111
4.2.45 Caso de Uso – Emitir Relatório de Cupons.....	112
4.2.46 Caso de Uso – Emitir Relatório de Agendas	114
4.2.47 Caso de Uso – Emitir Relatório de Favoritos	115
4.2.48 Caso de Uso – Emitir Relatório de Recebimentos	116
4.2.49 Caso de Uso – Gerar Dashboard de Parceiro	118
4.2.50 Caso de Uso – Gerar Dashboard de Administrador	118
4.2.51 Caso de Uso – Agendar	118
5 DIAGRAMA DE CLASSES	120
6 DIAGRAMAS DE SEQUÊNCIA.....	121
REFERÊNCIAS	125

1 DOCUMENTO DE ESPECIFICAÇÃO DO PROJETO

1.1 Dados da Empresa Cliente

Nome Fantasia: Sombrero Turismo.

Razão Social: Sombrero Viagens e Lazer LTDA

CNPJ: 11.123.123/0001-23

Endereço: Servidão João Januário do Nascimento, 103 - Barra da Lagoa, Florianópolis - SC, 88061-120

Telefones: (48) 99683-5238.

Cliente/Master: Lucas Gabriel Reis Lopes.

Cliente/Usuários: Público em geral.

Descrição da Empresa/Comércio: A Sombrero Turismo é uma empresa que tem como objetivo incentivar o turismo na região onde atua. Ela oferece vários serviços para seus clientes, como restaurante, pousada e bar, e agora está se expandindo para incluir vendas de cupons na internet. O objetivo é aumentar ainda mais o turismo na região e tornar a experiência dos turistas ainda mais agradável. Para isso, a Sombrero Turismo está desenvolvendo um sistema que permita a venda de cupons aos usuários e ajudará a empresa a ter um melhor controle dos seus serviços, incluindo pousadas, banquetes e reservas. Possui um total de 8 funcionários. Atualmente a empresa não se encontra informatizada. Em resumo, a Sombrero Turismo está trabalhando para tornar a região mais atraente para os turistas e oferecer a eles uma experiência incrível e inesquecível.

1.2 Sobre o Produto de *Software*

Será desenvolvido um *software* para automatização do processo de negócio de uma empresa de turismo. O *software* prestará atendimento virtual. Para a primeira versão, o *software* terá o nome de **Tarrafada de Ofertas 1.0**.

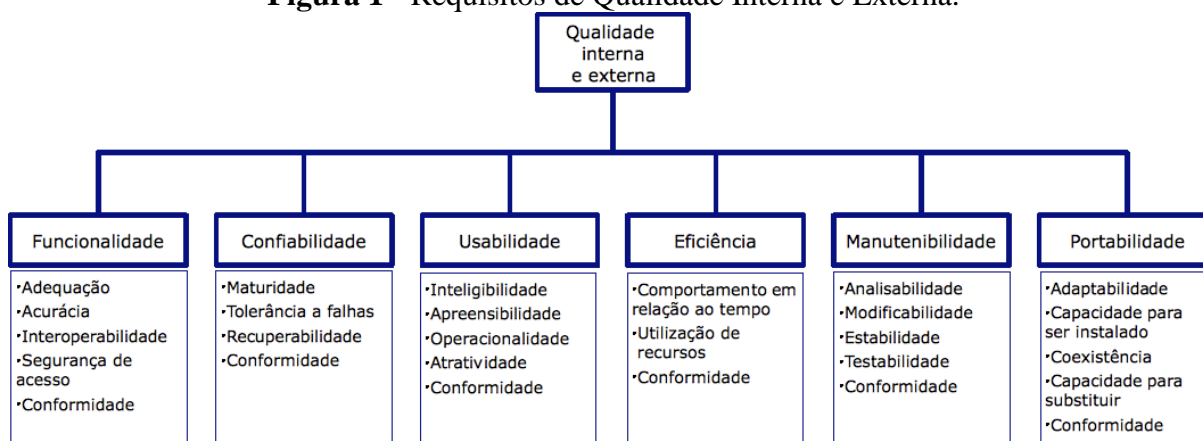
Tarrafada de Ofertas 1.0 é um sistema web de venda de cupons para restaurantes e locais turísticos em Florianópolis, com o objetivo de fomentar o turismo na região. Com o Tarrafada, os usuários podem facilmente acessar uma ampla variedade de opções de gastronomia e lazer na região, oferecendo descontos atrativos em suas escolhas. O sistema web do Tarrafada foi desenvolvido para ser acessível de diversos dispositivos, sejam desktops ou

smartphones, bastando apenas conexão à internet e acesso a um navegador web. Além disso, a interface do usuário foi desenvolvida para ser intuitiva, possibilitando uma navegação rápida e fácil para encontrar e adquirir os cupons de seu interesse. Com o Tarrafada de Ofertas, você pode ter acesso a uma grande seleção de opções turísticas e gastronômicas em Florianópolis com descontos exclusivos, tornando sua experiência ainda mais inesquecível e econômica.

1.3 Requisitos de Qualidade

A ISO nº 9126 determina uma série de requisitos de qualidade de *software* que um *software* a ser construído pode abranger. Esses requisitos de qualidade abrangem aspectos internos do *software* como linguagens, códigos, classes, etc. Assim como os aspectos externos do *software* como a interface da função, botões, cores, entre outras. A seguir a Figura 01 exemplificados quais são os requisitos de qualidade do *software* divididos por áreas.

Figura 1 - Requisitos de Qualidade Interna e Externa.



Fonte: ISO 9126.

A partir da ISO 9126 o produto deverá atender aos seguintes requisitos de qualidade:

- O uso será realizado por meio de uma interface gráfica moderna, seguindo os padrões de design atuais;
- A utilização do sistema será intuitiva e rápida, visando oferecer uma melhor agilidade no atendimento ao cliente. Ele contará com ícones de fácil acesso e fontes dimensionadas adequadamente para cada usuário, além de possuir uma navegação simples e intuitiva, o que tornará o processo de atendimento ainda mais ágil e eficiente;
- O sistema deve ser compatível com todos os dispositivos, proporcionando uma experiência suave e eficiente para usuários em desktop, tablet ou smartphone.

Ajustável a diferentes tamanhos de tela e resoluções, oferecendo interface intuitiva e fácil de navegar, a adaptabilidade do sistema é crucial para garantir a melhor experiência para os usuários a qualquer momento;

- O sistema irá sincronizar com as administradoras de cartões de crédito e débito para realizar as transações de cartão na hora do pagamento do cliente;
- A interface deve ser projetada com acessibilidade para ser usada por todos. Deve incluir opções de cor e tamanho de fonte, descrições de áudio, navegação por teclado e suporte a leitor de tela;
- O *software* será mantido de forma remota e contínua para garantir funcionamento otimizado e seguro, incluindo correções de bugs, atualizações de segurança, entre outros ajustes;
- O sistema terá função de salvamento automático segura, protegendo informações de usuários e empresas em caso de interrupções, com medidas de segurança rigorosas garantindo privacidade e confidencialidade;
- Entregar soluções no prazo com qualidade e funcionalidade é fundamental para a satisfação dos usuários e eficiência dos negócios. Desenvolvedores se responsabiliza pelo desempenho e manutenção do sistema para garantir a disponibilidade e confiabilidade das informações;
- Testar o *software* é crucial para garantir sua qualidade, funcionalidade, usabilidade e segurança para usuários. Corrige erros antes do lançamento e garante solução robusta e eficiente. Testar é essencial para garantir a satisfação dos usuários;
- O site deve ser atualizado constantemente para garantir a melhor experiência do usuário. Atualizações imediatas asseguram a funcionalidade e usabilidade do site, aprimorando sua performance e proporcionando uma navegação mais agradável aos usuários;
- A publicidade das ofertas no site preserva a imagem e credibilidade da empresa anunciada, bem como será responsável por divulgar com precisão e zelo as ofertas;
- A equipe de suporte do site estará disponível para solucionar dúvidas sobre as ofertas, bem como outras perguntas que possam surgir, buscando sempre melhorar a experiência do usuário.

1.4 Aspectos Técnicos do Produto

O *software* será construído utilizando a linguagem de programação JavaScript, que será a base para as demais tecnologias que vamos usar, como Vue.js e Tailwind para o front-end, além de Node.js e Nest.js para back-end. utilizando a ferramenta *Visual Studio Code* para o desenvolvimento da aplicação e o SGBD (Sistema de Gerenciamento de Banco de Dados) MySQL Workbench para a modelagem e construção do banco de dados do *software*.

Durante o processo de análise e projeto do sistema também serão utilizadas as ferramentas Astah UML para a modelagem dos diagramas UML e a ferramenta Jira para o gerenciamento das tarefas da equipe.

Já na fase de projeto serão utilizadas as ferramentas Figma para a projeção de protótipos de interfaces.

1.4.1 Arquitetura de Rede

Devido ao *software* ser construído em plataforma web será utilizado a arquitetura cliente-servidor para o funcionamento do *software* em rede. A camada de aplicação será implementada em 3 instâncias em nuvem, sendo uma para o front-end e as API's para consultas ao back-end, outra para o banco de dados e por último uma com um *proxy* reverso que irá receber as requisições externas e redirecioná-las dentro da rede privada, para o servidor de aplicação e para o banco de dados. As três na mesma rede privada, porém apenas o *proxy* recebera requisições externas.

A configuração mínima prevista do servidor de rede será:

- Processador: Intel® Core™ i5-6500 Cache de 6M, até 3,60 GHz;
- Memória: 8 Gb;
- Disco Rígido: 50Gb para backup;
- Sistema Operacional: Ubuntu Server.

A configuração mínima prevista do computador cliente será:

- Processador: Intel® Core™ Intel i3-7100 3.1 GHz;
- Memória: 8 Gb;
- Disco Rígido;
- Sistema Operacional: Windows Vista, Windows 7, Windows 8, Windows 8.1, Windows 10.

1.4.2 Tecnologias *Front End*

Para o desenvolvimento da interface do usuário do *software* "Tarrafada de Ofertas 1.0", utilizaremos as tecnologias Vue.js e Tailwind CSS.

O Vue.js é um framework JavaScript progressivo que facilita a criação de interfaces interativas e dinâmicas. Com Vue.js, seremos capazes de construir componentes reutilizáveis e gerenciar o estado da aplicação de maneira eficiente, proporcionando uma experiência de usuário fluida e responsiva. O uso do Vue.js permitirá que a interface do "Tarrafada de Ofertas" seja intuitiva e fácil de navegar, possibilitando aos usuários encontrar e adquirir cupons de seu interesse com rapidez e facilidade.

Para o design e estilização, utilizaremos o Tailwind CSS, um framework CSS utilitário que oferece uma abordagem altamente personalizável e eficiente para a criação de *layouts* modernos. O Tailwind CSS permitirá aplicar estilos diretamente nos elementos HTML através de classes utilitárias, garantindo consistência visual e agilidade no desenvolvimento. Isso é essencial para criar uma interface gráfica moderna, que segue os padrões de design atuais, com ícones de fácil acesso e fontes dimensionadas adequadamente para cada usuário.

A integração entre Vue.js e Tailwind será essencial para construir uma interface de usuário coesa e intuitiva. O Vue.js gerenciará a lógica e a interação dos componentes, enquanto o Tailwind CSS cuidará da estilização e do layout, assegurando um design responsivo que se adapta a diferentes dispositivos e tamanhos de tela. Essa adaptabilidade é crucial para garantir a melhor experiência para os usuários, seja em desktop, tablet ou smartphone.

Durante a fase de projeto, utilizaremos o Figma para a projeção de protótipos de interfaces, assegurando que o design final atenda aos requisitos de qualidade estabelecidos, como acessibilidade, usabilidade e estética. Além disso, o desenvolvimento será realizado utilizando a ferramenta Visual Studio Code, que proporcionará um ambiente de desenvolvimento integrado e eficiente para o trabalho com JavaScript, Vue.js e Tailwind CSS.

A arquitetura adotada no front-end permitirá uma separação clara entre a lógica de apresentação e a lógica de negócios, facilitando a manutenção e a escalabilidade da aplicação. Os componentes Vue.js serão responsáveis por renderizar os dados recebidos das APIs desenvolvidas em Node.js e NestJS, aplicando os estilos definidos pelo Tailwind CSS para criar uma interface atraente e funcional.

Essa configuração permitirá um desenvolvimento ágil e eficiente, com foco tanto na performance quanto na experiência do usuário, assegurando que a aplicação web "Tarrafada de Ofertas" oferece uma navegação intuitiva e uma aparência moderna. Assim, o sistema estará preparado para fornecer uma experiência de uso agradável e eficiente, atendendo às necessidades dos usuários e promovendo o turismo na região de Florianópolis através de cupons atrativos e exclusivos.

1.4.3 Tecnologias Back End

O desenvolvimento do *software* será realizado utilizando Node.js com o framework NestJS, que proporcionará uma estrutura robusta e modular para a construção das APIs da aplicação. A integração com o banco de dados será facilitada pelo Prisma, um ORM (*Object-Relational Mapping*) moderno, que permitirá um mapeamento eficiente dos dados entre a aplicação e o banco de dados.

Para o gerenciamento dos dados, será utilizado o MySQL, um sistema de gerenciamento de banco de dados relacional confiável e de alto desempenho. A administração do banco de dados será realizada com o MySQL Workbench, que oferece uma interface gráfica intuitiva para modelagem, desenvolvimento e manutenção das estruturas de dados.

A arquitetura adotada será a cliente-servidor, onde a aplicação web se comunicará com as APIs desenvolvidas em Node.js e NestJS. As APIs serão responsáveis por processar as requisições dos clientes, executar a lógica de negócios e interagir com o banco de dados através do Prisma. Essa configuração permitirá uma eficiente sincronização e compartilhamento de dados entre os clientes, além de facilitar a manutenção e atualização das APIs no servidor.

1.5 Objetivos Gerenciais do Projeto

Entregas: Durante o processo de desenvolvimento do *software* uma versão do produto deverá ser disponibilizada ao cliente ao final de cada sprint para que as funções desenvolvidas até determinado ponto sejam avaliadas pelo usuário.

Prazo: O prazo máximo para o desenvolvimento do *software* será de 6 meses;

Custos: Os custos previstos do projeto serão descritos nas Tabela 01 e 02 a seguir.

Tabela 1 - Custo por Mês.

Item	Descrição	Quant	Valor Unt.	Valor Total
01	Remuneração Mensal por Membro da Equipe	02	R\$ 3.000,00	R\$ 6.000,00
02	Auxílio Home Office (Internet, Energia, etc.)	02	R\$ 200,00	R\$ 400,00
03	Benefícios e Outros Custos (Saúde, etc.)	02	R\$ 300,00	R\$ 600,00
04	Ferramentas de Desenvolvimento (Licenças)	02	R\$ 150,00	R\$ 300,00
05	Hardware (Manutenção/Depreciação)	02	R\$ 100,00	R\$ 200,00
06	Serviços de Hospedagem e Infraestrutura	01	R\$ 500,00	R\$ 500,00
07	Cursos e Treinamentos	02	R\$ 100,00	R\$ 200,00
08	Ferramentas de Colaboração (Jira, etc.)	01	R\$ 100,00	R\$ 100,00
			Total por Mês	R\$ 8.300,00

Fonte: Produzido pelo autor (2024).

Tabela 2 - Custo Total do Projeto.

Item	Descrição	Tempo Previsto	Valor Unt.	Valor Total
01	Custo Fixo Mensal	06 meses	R\$ 8.300,00	R\$ 49.800,00
02	Lucro Previsto no Projeto	01	R\$ 80.000,00	R\$ 80.000,00
			Total	R\$ 129.800,00

Fonte: Produzido pelo autor (2024).

Manutenção: O custo de manutenção mensal do *software* após sua conclusão e implantação no cliente será de 01 (um) salário-mínimo por mês;

Corretiva: Tem como objetivo resolver problemas de *software* não detectados durante a fase de testes;

Adaptativa: Aborda ajustes no *software* para atender às constantes mudanças nas necessidades;

Evolutiva: Tem como finalidade aprimorar a qualidade do *software*;

Preventiva: Objetiva aprimorar a confiabilidade, identificando potenciais fontes de erros futuros no *software*;

Contrato: Este acordo de prestação de serviços é firmado entre a equipe de desenvolvimento e o cliente, podendo ser revisto a qualquer momento por ambas as partes, desde que a outra seja comunicada em até 48 horas. Além de cobrir o tempo e os recursos necessários, o contrato também inclui uma avaliação dos riscos financeiros associados ao projeto. Este é um acordo de desenvolvimento incremental que visa priorizar e trazer valor ao negócio, reduzindo drasticamente as chances de fracasso. Para garantir a segurança do projeto, todas as partes

envolvidas deverão seguir rigorosamente as normas e procedimentos estabelecidos neste contrato. Isso inclui a obrigação de manter a confidencialidade das informações compartilhadas, bem como garantir a integridade dos dados e dos recursos utilizados. A equipe de desenvolvimento também se compromete a implementar medidas de segurança adequadas para proteger o *software* e os dados do cliente.

Comunicação: A fim de garantir uma comunicação efetiva e acompanhamento dos avanços do projeto ao longo do tempo, será utilizado o Taiga como ferramenta de gestão de projetos. Assim, será possível acompanhar as atividades semanais, quinzenais e mensais de maneira organizada e transparente entre todos os membros envolvidos. Para ampliar a colaboração e o acesso a informações importantes, o projeto será compartilhado na plataforma de compartilhamento de arquivos, como o Google Docs. Além disso, reuniões periódicas serão realizadas para avaliar a eficiência dos processos, definir prioridades e garantir que todos estejam alinhados. Essas reuniões podem ser presenciais ou online via *Discord*, de acordo com a preferência dos envolvidos. Ao longo do desenvolvimento do projeto, serão incentivados *feedbacks* contínuos, através da entrega frequente de partes do *software*;

1.6 Equipe de Desenvolvimento

A equipe de desenvolvimento será composta pelos seguintes papéis e atribuições, de acordo com a metodologia **ágil Scrum** estabelecida para este projeto:

Quadro 1 - Papeis e atribuições da equipe de desenvolvimento.

ID	Nome	Função	Atribuições
01	Tiago Rafael Oliveira das Virgens	Desenvolvedor back end	Desenvolver o projeto de <i>software</i> .
02	Mateus França da Cunha	Desenvolvedor front-end	Desenvolver o projeto de <i>software</i> .
03	Lucas Gabriel Reis Lopes	Product Owner	Representante do cliente que ajuda no entendimento do negócio.
04	Prof. Jackson Henrique	Especialista	Usuário especializado no domínio da informação que ajuda no entendimento do negócio.

Fonte: Produzido pelo autor (2024).

1.7 Metodologia de Desenvolvimento

A metodologia de desenvolvimento Ágil Scrum é uma boa escolha para o projeto Tarrafada 1.0. Ela é baseada na colaboração e na flexibilidade, permitindo que o time trabalhe juntamente para alcançar objetivos comuns. O Scrum possui uma estrutura bem definida e um processo de desenvolvimento que permite uma maior agilidade na resolução de problemas e na tomada de decisões. Além disso, o envolvimento de todos os membros do time é fundamental para o sucesso do projeto, pois todos são responsáveis pela entrega de valor ao cliente.

1.7.1 Sobre a Metodologia de Pesquisa

A metodologia Scrum foi criada por Ken Schwaber e Jeff Sutherland, inspirada nas ideias de adaptação e flexibilidade. Jeff Sutherland, um piloto de avião, notou que não existe uma forma fixa de pousar um avião devido aos vários fatores envolvidos, exigindo ajustes a cada vez. Ele aplicou essa ideia aos grandes projetos, onde há muitas atividades e pessoas diferentes envolvidas, e surpresas podem surgir ao longo do caminho. A partir daí, a metodologia Scrum se tornou muito popular no setor de tecnologia e, posteriormente, em outras áreas.

A Metodologia Scrum é uma abordagem ágil para o gerenciamento de projetos e desenvolvimento de produtos que tem como objetivo ajudar as equipes a trabalhar de forma mais eficiente e entregar valor ao cliente de forma mais rápida. Ela foi desenvolvida no final dos anos 80 e início dos anos 90 como uma solução para os desafios enfrentados pelas equipes de desenvolvimento de *software*, mas desde então tem sido amplamente utilizada em vários setores, incluindo gerenciamento de projetos, desenvolvimento de produtos e gerenciamento de equipes.

A metodologia Scrum é baseada em três pilares: transparência, inspeção e adaptação. Isso significa que a equipe deve ser transparente sobre seu progresso, inspecionar regularmente seu trabalho e se adaptar às mudanças de forma rápida e eficiente. O Scrum também define alguns papéis importantes, incluindo o Scrum Master, o Product Owner e os membros do time de desenvolvimento.

Figura 2 - Pilares do scrum.

SCRUM



Fonte: SCRUM.AS.

O Scrum Master é responsável por garantir que a equipe siga a metodologia Scrum e remove obstáculos que impedem o progresso da equipe. Ele também é responsável por facilitar as cerimônias Scrum e garantir que a equipe esteja seguindo as melhores práticas. O Product Owner é responsável por representar os interesses do cliente e garantir que o trabalho da equipe esteja alinhado aos objetivos do negócio. Ele também é responsável por garantir que o backlog do produto esteja priorizado e atualizado regularmente.

Os membros do time de desenvolvimento são responsáveis por planejar, desenvolver e entregar o produto final. Eles trabalham juntos para cumprir as metas da sprint e garantir que o produto esteja sendo desenvolvido de acordo com as expectativas do cliente. O time de desenvolvimento é responsável por tomar decisões técnicas e garantir que o trabalho esteja sendo realizado de forma eficiente e eficaz.

Figura 3 - Papéis no scrum.



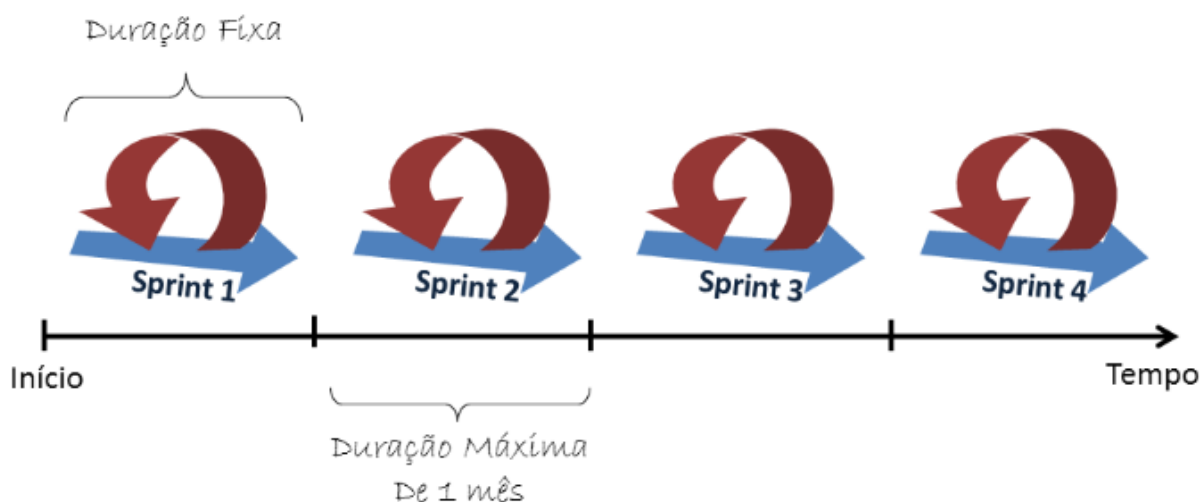
Fonte: Euax.

A metodologia Scrum também define algumas cerimônias importantes, incluindo o Daily Scrum, a Sprint Review e a Sprint Retrospective. O Daily Scrum é uma reunião diária

curta onde a equipe se reúne para revisar o progresso e planejar a próxima iteração. A Sprint Review é uma reunião no final de cada sprint onde a equipe apresenta o trabalho concluído ao cliente e discute as melhorias para a próxima sprint. A Sprint Retrospective é uma reunião no final de cada sprint onde a equipe se reúne para revisar o que funcionou bem e o que precisa ser melhorado na próxima sprint.

O Scrum também define o conceito de Sprint, que é um período de tempo fixo (geralmente de 1 a 4 semanas) durante o qual a equipe trabalha em conjunto para completar um conjunto de tarefas prioritizadas. A equipe deve se comprometer a entregar um trabalho funcional ao final de cada sprint e está sempre buscando melhorias contínuas.

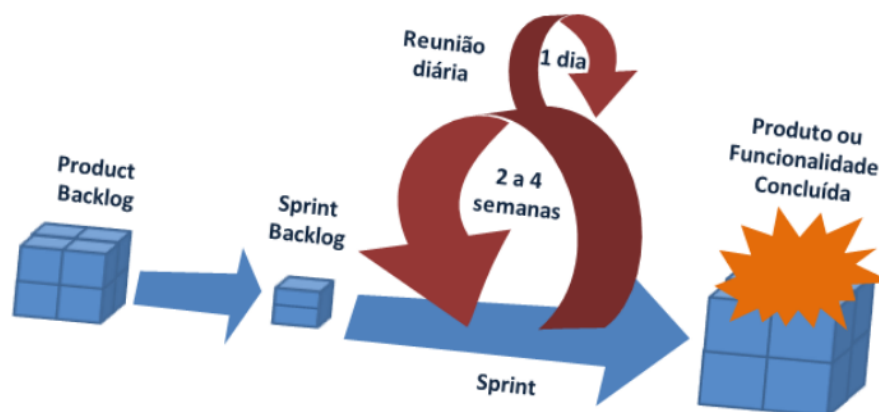
Figura 4 - Modelo de sprint.



Fonte: Master Mind.

O processo começa com o planejamento da sprint, onde o backlog do produto é revisado e as tarefas são selecionadas para o sprint backlog. Durante a sprint, a equipe trabalha diariamente em reuniões chamadas Daily Scrum para revisar o progresso. No final da sprint, há uma revisão da sprint para apresentar o incremento do produto e uma retrospectiva da sprint para identificar melhorias para o próximo ciclo.

Figura 5 - Ciclo de Vida do Scrum.



Fonte: Master Mind.

Em resumo, a metodologia Scrum é uma abordagem ágil que ajuda as equipes a entregar valor ao cliente de forma mais rápida e eficiente. É baseada em três pilares: transparência, inspeção e adaptação, e define alguns papéis importantes, cerimônias e conceitos que ajudam as equipes a trabalhar de forma colaborativa e eficaz. Se você está procurando uma maneira de gerenciar projetos e desenvolver produtos de forma mais ágil, a metodologia Scrum pode ser uma ótima opção para você.

1.7.2. Ciclo de Vida da Metodologia no Projeto do Sistema

Para o desenvolvimento do *software* será utilizado sprints com duração de 14 dias. Serão realizadas as reuniões de abertura, fechamento e revisão de sprint previstas no Scrum. A reunião de revisão de resultados será adaptada conforme as necessidades surgirem. Além disso, haverá reuniões diárias de acompanhamento (daily stand-ups) para garantir a comunicação contínua e o alinhamento da equipe, e sessões de planejamento detalhado no início de cada sprint para definir claramente as metas e prioridades.

1.7.2.1 Product Backlog

Neste tópico serão documentados todos os itens do Product Backlog criados ao longo do desenvolvimento do projeto. Destaca-se que, por se tratar de um projeto acadêmico cujo

principal produto gerado é o Projeto do *Software*, os itens do Product Backlog serão os artefatos do projeto gerados ao longo do processo de elaboração do projeto.

A tabela a seguir apresenta os itens do Product Backlog, descrevendo cada item, a estimativa de esforço necessária para sua realização e sua prioridade:

Tabela 3 - Itens do Product Backlog.

Item	Descrição	Estimativa de Esforço	Prioridade
1	Elaboração do Documento de Especificação do Projeto	30 dias	1°
2	Escrita do Documento de Requisitos	30 dias	2°
3	Prototipação do sistema	30 dias	3°
4	Criação do Diagrama de Caso de Uso	5 dias	4°
5	Escrita dos casos de uso expandidos	10 dias	5°
6	Criação do Diagrama de Classes	3 dias	6°
7	Criação do Diagrama de Sequência	7 dias	7°

Fonte: Produzido pelo autor (2024).

1.7.2.2 Sprints Backlog

Neste tópico serão documentados todos os itens das Sprints Backlog criados ao longo do desenvolvimento do projeto. Todas as tarefas do projeto foram gerenciadas com a ferramenta Jira.

Quadro 2 – Descrição das tarefas do projeto.

Item do PB	Nº e Período	Descrição da Tarefa	Responsável	Esforço
		1.1. Dados da Empresa Cliente	Mateus França da Cunha	8h
		1.2. Sobre o Produto de <i>Software</i>	Mateus França da Cunha	8h
		1.3. Requisitos de Qualidade	Mateus França da Cunha	16h
		1.4. Aspectos Técnicos do Produto	Tiago Rafael Oliveira das Virgens	8h
		1.4.1. Arquitetura de Rede	Tiago Rafael Oliveira das Virgens	8h
		1.4.2. Tecnologias Front End	Mateus França da Cunha	4h

1	Sprint 1: 15/03/2024 a 15/04/2024	1.4.3. Tecnologias Back End	Tiago Oliveira Virgens	Rafael das	4h
		1.5. Objetivos Gerenciais do Projeto	Tiago Oliveira Virgens	Rafael das	8h
		1.6. Equipe de Desenvolvimento	Mateus França da Cunha		1h
		1.7. Metodologia de Desenvolvimento	Mateus França da Cunha		8h
		1.7.1. Sobre a Metodologia de Pesquisa	Mateus França da Cunha		8h
		1.7.2. Ciclo de Vida da Metodologia no Projeto do Sistema	Tiago Oliveira Virgens	Rafael das	8h
		1.7.2.1. Product Backlog	Tiago Oliveira Virgens	Rafael das	1h
		1.7.2.2. Sprints Backlog	Mateus França da Cunha		8h
		1.7.2.3. Reuniões das Sprints	Mateus França da Cunha		8h
		1.8. Entrevistas	Mateus França da Cunha		
		2.1. Acessar Login	Tiago Oliveira Virgens	Rafael das	8h
		2.2. Acessar Parceiro	Tiago Oliveira Virgens	Rafael das	8h
		2.3. Acessar Login Administrador	Tiago Oliveira Virgens	Rafael das	8h
		2.4. Redefinir Senha Cliente	Tiago Oliveira Virgens	Rafael das	8h
		2.5. Redefinir Senha Parceiro	Tiago Oliveira Virgens	Rafael das	8h
		2.6. Redefinir Senha Administrador	Tiago Oliveira Virgens	Rafael das	8h

2	<i>Sprint 2:</i> 15/04/2024 a 15/05/2024	2.7. Recuperar Senha Cliente	Tiago Oliveira Virgens	Rafael das	8h
		2.8. Recuperar Senha Parceiro	Tiago Oliveira Virgens	Rafael das	8h
		2.9. Recuperar Senha Administrador	Tiago Oliveira Virgens	Rafael das	8h
		2.10. Alterar Senha Cliente	Tiago Oliveira Virgens	Rafael das	8h
		2.11. Alterar Senha Parceiro	Tiago Oliveira Virgens	Rafael das	8h
		2.12. Acessar Tela Inicial Cliente	Tiago Oliveira Virgens	Rafael das	8h
		2.13. Acessar Menu Inicial Parceiro	Tiago Oliveira Virgens	Rafael das	8h
		2.14. Acessar Menu Inicial Administrador	Tiago Oliveira Virgens	Rafael das	8h
		2.15. Cadastrar Categoria	Tiago Oliveira Virgens	Rafael das	8h
		2.16. Cadastrar Cliente	Tiago Oliveira Virgens	Rafael das	8h
		2.17. Cadastrar Parceiro	Tiago Oliveira Virgens	Rafael das	8h
		2.18. Cadastrar Cupons	Tiago Oliveira Virgens	Rafael das	8h
		2.19. Cadastrar Agendas	Tiago Oliveira Virgens	Rafael das	8h
2.20. Favoritar Cupons	Tiago Oliveira Virgens	Rafael das	8h		

	2.21. Comprar Cupons	Tiago Oliveira Virgens	Rafael das	8h
	2.22. Receber Vendas	Tiago Oliveira Virgens	Rafael das	8h
	2.23. Emitir Voucher	Tiago Oliveira Virgens	Rafael das	8h
	2.24. Validar Voucher	Mateus Cunha	França da	8h
	2.25. Estornar Vendas	Mateus Cunha	França da	8h
	2.26. Consultar Favoritos	Mateus Cunha	França da	8h
	2.27. Consultar Categorias	Mateus Cunha	França da	8h
	2.28. Consultar Clientes	Mateus Cunha	França da	8h
	2.29. Consultar Parceiros	Mateus Cunha	França da	8h
	2.30. Consultar Cupons	Mateus Cunha	França da	8h
	2.31. Consultar Agendas	Mateus Cunha	França da	8h
	2.32. Consultar Vendas	Mateus Cunha	França da	8h
	2.33. Consultar Recebimentos	Mateus Cunha	França da	8h
	2.34. Consultar Voucher	Mateus Cunha	França da	8h
	2.35. Consultar Solicitações de Cupons	Mateus Cunha	França da	8h
	2.36. Consultar Solicitações de Parceiros	Mateus Cunha	França da	8h
	2.37. Autorizar Solicitações de Parceiros	Mateus Cunha	França da	8h
	2.38. Autorizar Solicitações de Cupons	Mateus Cunha	França da	8h
	2.39. Emitir Relatório de Vendas	Mateus Cunha	França da	8h
	2.40. Emitir Relatório de Vouchers	Mateus Cunha	França da	8h

		2.41. Emitir Relatório de Categorias	Mateus França da Cunha	8h
		2.42. Emitir Relatório de Clientes	Mateus França da Cunha	8h
		2.43. Emitir Relatório de Parceiros	Mateus França da Cunha	8h
		2.44. Emitir Relatório de Cupons	Mateus França da Cunha	8h
		2.45. Emitir Relatório de Agendas	Mateus França da Cunha	8h
		2.46. Emitir Relatório de Favoritos	Mateus França da Cunha	8h
		2.47. Emitir Relatório de Recebimentos	Mateus França da Cunha	8h
		2.48. Gerar Dashboard de Parceiros	Mateus França da Cunha	8h
		2.49. Gerar Dashboard de Administrador	Mateus França da Cunha	8h
		2.50. Agendar	Mateus França da Cunha	8h
3	<i>Sprint 3: 15/05/2024 a 15/06/2024</i>	3. Protótipo do Sistema	Mateus França da Cunha	176h
		4. Diagrama de Caso de Uso	Tiago Oliveira Virgens Rafael das	8h
		4.1.1. Administrador	Tiago Oliveira Virgens Rafael das	8h
		4.1.2. Parceiro	Tiago Oliveira Virgens Rafael das	8h
		4.1.2. Cliente	Tiago Oliveira Virgens Rafael das	8h
		4.2. Caso de Uso Expandido	Tiago Oliveira Virgens Rafael das	8h
		4.2.1. Caso de Uso - Acessar Login Cliente	Tiago Oliveira Virgens Rafael das	8h

4	<i>Sprint 4: 15/06/2024 a 15/07/2024</i>	4.2.1. Caso de Uso - Acessar Login Parceiro	Tiago Oliveira Virgens	Rafael das	8h
		4.2.3. Caso de Uso - Acessar Login Administrador	Tiago Oliveira Virgens	Rafael das	8h
		4.2.4. Caso de Uso - Redefinir Senha Cliente	Tiago Oliveira Virgens	Rafael das	8h
		4.2.5. Caso de Uso - Redefinir Senha Parceiro	Tiago Oliveira Virgens	Rafael das	8h
		4.2.6. Caso de Uso - Redefinir Senha Administrador	Tiago Oliveira Virgens	Rafael das	8h
		4.2.7. Caso de Uso - Recuperar Senha Cliente	Tiago Oliveira Virgens	Rafael das	8h
		4.2.8. Caso de Uso - Recuperar Senha Parceiro	Tiago Oliveira Virgens	Rafael das	8h
		4.2.9. Caso de Uso - Recuperar Senha Administrador	Tiago Oliveira Virgens	Rafael das	8h
		4.2.10. Caso de Uso - Alterar Senha Cliente	Tiago Oliveira Virgens	Rafael das	8h
		4.2.11. Caso de Uso - Alterar Senha Parceiro	Tiago Oliveira Virgens	Rafael das	8h
		4.2.12. Caso de Uso - Alterar Senha Administrador	Tiago Oliveira Virgens	Rafael das	8h
		4.2.13. Caso de Uso - Acessar Tela Inicial Cliente	Tiago Oliveira Virgens	Rafael das	8h
		4.2.14. Caso de Uso - Acessar Menu Inicial Parceiro	Tiago Oliveira Virgens	Rafael das	8h
		4.2.15. Caso de Uso - Acessar Menu Inicial Administrador	Tiago Oliveira Virgens	Rafael das	8h

	4.2.16. Caso de Uso - Cadastrar Categorias	Tiago Oliveira Virgens	Rafael das	8h
	4.2.17. Caso de Uso - Cadastrar Cliente	Tiago Oliveira Virgens	Rafael das	8h
	4.2.18. Caso de Uso - Cadastrar Parceiros	Tiago Oliveira Virgens	Rafael das	8h
	4.2.19. Caso de Uso - Cadastrar Cupons	Tiago Oliveira Virgens	Rafael das	8h
	4.2.20. Caso de Uso - Cadastrar Agendas	Tiago Oliveira Virgens	Rafael das	8h
	4.2.21. Caso de Uso - Favoritar Cupons	Tiago Oliveira Virgens	Rafael das	8h
	4.2.22. Caso de Uso - Comprar Cupons	Tiago Oliveira Virgens	Rafael das	8h
	4.2.23. Caso de Uso - Receber Vendas	Tiago Oliveira Virgens	Rafael das	8h
	4.2.24. Caso de Uso - Emitir Voucher	Tiago Oliveira Virgens	Rafael das	8h
	4.2.25. Caso de Uso - Validar Vouchers	Mateus França da Cunha		8h
	4.2.26. Caso de Uso - Estornar Vendas	Mateus França da Cunha		8h
	4.2.27. Caso de Uso - Consultar Favoritos	Mateus França da Cunha		8h
	4.2.28. Caso de Uso - Consultar Clientes	Mateus França da Cunha		8h
	4.2.29. Caso de Uso - Consultar Parceiros	Mateus França da Cunha		8h
	4.2.30. Caso de Uso - Consultar Cupons	Mateus França da Cunha		8h
	4.2.31. Caso de Uso - Consultar Agendas	Mateus França da Cunha		8h
	4.2.32. Caso de Uso - Consultar Vendas	Mateus França da Cunha		8h

		4.2.33. Caso de Uso - Consultar Recebimentos	Mateus França da Cunha	8h
		4.2.34. Caso de Uso - Consultar Vouchers	Mateus França da Cunha	8h
		4.2.35. Caso de Uso - Consultar Solicitação de Cupons	Mateus França da Cunha	8h
		4.2.36. Caso de Uso - Consultar Solicitação de Parceiros	Mateus França da Cunha	8h
		4.2.37. Caso de Uso - Autorizar Solicitação de Parceiros	Mateus França da Cunha	8h
		4.2.38. Caso de Uso - Autorizar Solicitação de Cupons	Mateus França da Cunha	8h
		4.2.39. Caso de Uso - Emitir Relatório de Vendas	Mateus França da Cunha	8h
		4.2.40. Caso de Uso - Emitir Relatório de Vouchers	Mateus França da Cunha	8h
		4.2.41. Caso de Uso - Emitir Relatório de Categorias	Mateus França da Cunha	8h
		4.2.42. Caso de Uso - Emitir Relatório de Clientes	Mateus França da Cunha	8h
		4.2.43. Caso de Uso - Emitir Relatório de Parceiros	Mateus França da Cunha	8h
		4.2.44. Caso de Uso - Emitir Relatório de Cupons	Mateus França da Cunha	8h
		4.2.45. Caso de Uso - Emitir Relatório de Agendas	Mateus França da Cunha	8h
		4.2.46. Caso de Uso - Emitir Relatório de Favoritos	Mateus França da Cunha	8h
		4.2.47. Caso de Uso - Emitir Relatório de Recebimentos	Mateus França da Cunha	8h
		4.2.48. Caso de Uso - Gerar Dashboard de Parceiro	Mateus França da Cunha	8h
		4.2.49. Caso de Uso - Gerar Dashboard de Administrador	Mateus França da Cunha	8h
		4.2.50. Caso de Uso - Agendar	Mateus França da Cunha	8h
5	<i>Sprint 5: 15/07/2024 a 15/08/2024</i>	5. Diagrama de Classes	Tiago Oliveira Virgens Rafael das	8h
		6. Diagrama de Caso de Uso	Tiago Oliveira Virgens Rafael das	8h

		7. Diagrama de Sequência	Tiago Oliveira Virgens	Rafael das	8h
		7.1 Diagrama de Sequência - Login	Tiago Oliveira Virgens	Rafael das	8h

Fonte: Produzido pelo autor (2024).

1.7.2.3 Reuniões das Sprints

Documento aqui o tempo, data de início e fim da sprint, datas de reuniões, observações, problemas enfrentados e demais informações que julgarem necessário em cada uma das Sprints.

Quadro 3 – Reuniões das Sprints.

Sprint	Reunião	Resumo da Reunião	Data da Reunião
1	Reunião de Planejamento	O líder da equipe de projeto abriu a reunião com uma forte apelação ao comprometimento de cada membro com relação ao projeto. Ele então passou a coordenar a divisão de tarefas entre os participantes, com o objetivo de garantir que todos tivessem claro o que lhes cabia. Depois de algumas discussões produtivas, foi acordado que seria realizada uma reunião semanal para atualizarmos o progresso do projeto. O principal objetivo desta reunião seria incorporar as informações relevantes disponibilizadas pelo cliente ao documento oficial e chegar a um consenso geral sobre o que já foi escrito.	15/03/2024
	Reunião semanal	Durante a reunião semanal da equipe, o líder designou as tarefas específicas para cada membro do grupo, a serem realizadas durante a sprint 1 do projeto em desenvolvimento.	22/03/2024
	Reunião semanal	Durante a reunião semanal da equipe, foram realizadas verificações acerca do progresso das atividades até o momento. Ademais, foram promovidas discussões acerca da identificação de quaisquer funções do sistema que possam não ter sido ainda levantadas.	29/03/2024
	Reunião semanal	Durante a reunião semanal, foi discutido sobre a finalização das telas desenhadas no papel, as quais foram previamente delineadas, com o objetivo de promover a discussão sobre as mesmas.	05/04/2024

	Reunião de Retrospectiva	Durante a reunião de encerramento da interação, a equipe identificou alguns aspectos positivos e negativos. No que diz respeito aos aspectos positivos, foi observado que a comunicação interna do grupo está funcionando de maneira satisfatória. Por outro lado, como ponto negativo, a equipe destacou a ocorrência de uma subestimação do tempo necessário para concluir todas as atividades planejadas para a sprint em questão.	08/04/2024
2	Reunião semanal	Durante a reunião semanal da equipe, o líder designou as tarefas específicas para cada membro do grupo, a serem realizadas durante a sprint 2 do projeto em desenvolvimento.	15/04/2024
	Reunião semanal	Durante a reunião semanal da equipe, foram realizadas verificações acerca do progresso das atividades até o momento. Ademais, foram promovidas discussões acerca dos padrões a serem utilizados no sistema.	22/04/2024
	Reunião semanal	Durante a reunião semanal, foi discutido sobre o início das telas desenhadas no figma, as quais foram previamente delineadas, com o objetivo de promover a discussão sobre as mesmas.	29/04/2024
	Reunião semanal	Durante a reunião semanal, foi feito o compartilhamento entre todos acerca dos conhecimentos adquiridos na ferramenta figma, para que todos possam incluir interações nas suas telas, botões padronizados, etc.	06/05/2024
	Reunião semanal	Durante a reunião semanal, foi falado sobre quais telas foram finalizadas e foi analisado também sobre como iremos apresentar a sprint 2: Projeto de Interface para toda a sala no dia 19/04/2023.	13/05/2024
3	Reuniões diárias (15/05 - 30/05)	Durante a reunião diária, nos reunimos e falamos quais atividades fizemos durante as 24 horas do dia e o que planejamos para as próximas 24 horas.	15/05/2024
	Reunião semanal	Durante a reunião da semana da sprint, dividimos todas as atividades do projeto e marcamos para fazermos o diagrama de caso de uso em conjunto.	23/05/2024
4	Reuniões diárias (15/06 - 15/07)	Durante as reuniões diárias, discutimos o progresso no desenvolvimento do diagrama de caso de uso e do caso de uso expandido. Cada membro da equipe compartilhou as atividades realizadas nas últimas 24 horas e os planos para o próximo período.	15/06/2024
	Reunião semanal	Na reunião semanal da Sprint, revisamos os diagramas de caso de uso e discutimos o detalhamento dos casos de uso expandidos. Trabalhamos em conjunto para garantir que todos	15/07/2024

		os casos de uso fossem abrangentes e precisos, ajustando detalhes conforme necessário.	
5	Reuniões diárias (15/07 - 15/08)	Nas reuniões diárias, focamos na criação e revisão do diagrama de classes e do diagrama de sequência. Cada membro da equipe relatou o progresso feito nesses diagramas e os desafios encontrados, além de planejar as atividades para o dia seguinte.	15/07/2024
	Reunião semanal	Durante a reunião semanal da sprint, abordamos o desenvolvimento dos diagramas de classes e sequência, revisando e ajustando os detalhes para garantir a consistência e a clareza. Discutimos como esses diagramas interagem e influenciam o restante do projeto.	15/08/2024

Fonte: Produzido pelo autor (2024).

1.8 Entrevistas

Neste tópico serão documentadas todas as entrevistas, workshops ou observações realizadas ao longo do projeto.

Número Entrevista: 01

Cliente Entrevistado: Lucas Lopes

Data: 10/02/2023 e 11/02/2023

Perguntas:

- **Perguntas sobre a empresa:**
- Qual é o número de colaboradores da empresa?
- Há quanto tempo a empresa está em atividade?
- Como você está gerenciando seu negócio atualmente e quais são seus planos para o futuro?
- Qual tem sido sua experiência no setor ao longo do tempo?
- Como você gerencia o fluxo de clientes, desde a entrada até a saída?

- Como você organiza seus funcionários no estabelecimento?
- Como você administra os pedidos e quais materiais são utilizados?
- Como você processa as entregas e como essas são organizadas?
- Quais são as dificuldades que você enfrenta no dia a dia?
- **Perguntas para tentar entender o sistema desejado:**
- Já foram apresentadas outras ofertas para criar um sistema?
- Como vocês imaginam que o sistema personalizado para sua empresa funcione?
- Você está interessado em ter o sistema em web com uma opção de extensão mobile? Se sim, como você gostaria que isso funcionasse?
- Quais são as principais características que o sistema precisa ter?
- Qual é a maior preocupação da sua empresa no momento em relação ao site?
- Principais expectativas com o software
- **Síntese das Respostas:**

Durante a conversa com o Sr. Lucas Lopes, conseguimos entender melhor o sistema que ele quer implementar em sua empresa. Quando questionado sobre as características e funções centrais do sistema, ele explicou que precisa de um site fácil e intuitivo para usuários da terceira idade. Ele também espera estimular o comércio local e atrair novos clientes com preços mais acessíveis. O site de descontos gerará um código único para cada compra, que será apresentado ao pagar a conta em um estabelecimento que aceite o voucher.

O empresário de Florianópolis enfatizou que sua empresa tem uma história simples e humilde, formada principalmente por parentes e contando com oito funcionários que atuam

como atendentes, garçons, limpeza e caixa. O restaurante está localizado em uma área turística da cidade, com uma vista deslumbrante para a praia. Com quase duas décadas de existência, a empresa tem uma série de histórias e momentos inesquecíveis. O Sr. Lucas Lopes é apaixonado por seu negócio e está sempre procurando aperfeiçoá-lo e apoiar o comércio local, com o objetivo de atrair mais clientes para o seu restaurante e para outros estabelecimentos próximos à praia.

No momento, a gestão da empresa é feita manualmente, com finanças e agendamentos registrados em um bloco de notas. No entanto, o objetivo do Sr. Lucas Lopes é criar um site de descontos para atrair mais clientes, não apenas para si, mas também para outros comerciantes, especialmente durante a alta temporada turística, quando a região recebe um grande número de pessoas. Ele já tentou implementar esta ideia anteriormente e espera que no futuro possa ter uma versão para dispositivos móveis para atingir um público mais jovem.

O empresário quer que o site tenha uma aparência atrativa para se destacar da concorrência e ofereça conveniência na compra e um layout simples para facilitar a busca pelo produto desejado. Ele espera atingir o público-alvo, formado por pessoas da terceira idade, e ampliar sua base de clientes para os restaurantes locais.

Número Entrevista: 02

Cliente Entrevistado: Lucas Lopes

Data: 17/02/2023

Pergunta: Como funciona o voucher?

Síntese das Resposta:

O site de descontos gerará um código único para cada compra, que será apresentado ao pagar a conta em um estabelecimento que aceite o voucher. O cupom terá detalhes de localização, regulamento da oferta, preço, entre outras informações. O cliente poderá imprimir o cupom ou apresentá-lo no celular, facilitando o acesso às promoções. O Sr. Lucas Lopes enfatizou que a apresentação do voucher no celular é uma tendência cada vez mais forte, especialmente entre o público mais jovem. Aliás, ele também mostrou algumas telas de sites que já existem para exemplificar o que ele busca para o seu negócio. O empresário pretende que o site ofereça uma experiência agradável e prática para seus clientes, tanto na hora da compra quanto na apresentação do voucher.

Número Entrevista: 03

Cliente Entrevistado: Lucas Lopes

Data: 18/04/2023

Perguntas:

- **Perguntas sobre a empresa:**
- Quais são as cores principais do sistema e como elas foram escolhidas?
- Existe um esquema de cores específico usado em diferentes partes do sistema, como menus, botões ou ícones?
- Como as cores são usadas para indicar estados diferentes, como avisos, erros ou sucesso?
- O sistema usa cores personalizadas ou segue uma paleta de cores padrão?
- A acessibilidade das cores foi levada em consideração durante o desenvolvimento da interface?
- As cores da interface foram testadas com usuários antes do lançamento do sistema?
- Quais são as cores principais do sistema e como elas foram escolhidas?
- **Síntese das Respostas:**

O Sr. Lucas Lopes, expressou sua preferência por um design minimalista e objetivo, complementado pela cor azul, que, segundo ele, transmite seriedade e confiança, características importantes para o sucesso do projeto. Além disso, enfatizou a importância dos ícones para melhorar a experiência do usuário, tornando a navegação mais intuitiva e fácil de usar.

Número da Entrevista: 04

Cliente Entrevistado: Lucas Lopes

Data: 22/05/2023

Perguntas:

- **Perguntas sobre os recursos do sistema:**

- Quais são os recursos principais que você espera que o sistema ofereça aos usuários?
- Existe alguma funcionalidade específica que seja considerada um diferencial em relação a outros sistemas similares?
- Os usuários terão a opção de personalizar suas preferências ou configurações dentro do sistema?
- O sistema terá integrações com outras plataformas ou serviços externos?
- Haverá suporte para múltiplos idiomas no sistema?
- **Síntese das Respostas:**

O Sr. Lucas Lopes destacou que os principais recursos que ele espera que o sistema ofereça aos usuários são a facilidade de busca por descontos, a possibilidade de filtrar ofertas por categoria e a conveniência de visualizar os detalhes do voucher, incluindo o regulamento da oferta e o preço. Ele também mencionou que um diferencial importante do sistema será a função de recomendação personalizada de promoções com base nos interesses e histórico de compras dos usuários.

Quanto às configurações personalizáveis, o Sr. Lucas Lopes afirmou que os usuários terão a opção de personalizar suas preferências, como selecionar as categorias de ofertas de seu interesse e escolher receber notificações sobre promoções específicas.

Em relação a integrações, ele mencionou que está considerando a possibilidade de integrar o sistema com redes sociais populares para facilitar o compartilhamento de ofertas com amigos e familiares.

Quanto ao suporte a múltiplos idiomas, o Sr. Lucas Lopes expressou interesse em tornar o sistema acessível a um público mais amplo, incluindo a possibilidade de oferecer suporte a diferentes idiomas no futuro.

2. DOCUMENTO DE REQUISITOS

Após a fase de levantamento de requisitos, foram identificados e coletados os seguintes requisitos para o software:

Quadro 4 - Acessar Login Cliente.

Requisito Funcional		
Nome:	Acessar Login Cliente	Código: F1
Descrição:	O sistema deve validar o login do cliente.	
Estimativa de Esforço: 8h	Prioridade: 100 pontos	
Requisitos Não funcionais		
ID NF	Descrição	Categoria
1.1	O tamanho da tela deve se adaptar aos diferentes tamanhos de telas dos dispositivos eletrônicos, sejam eles com uso mobile ou desktop.	Portabilidade
1.2	É necessário o uso do sistema operacional Android ou IOS para executar o requisito funcional em sistemas mobile.	Portabilidade
1.3	O tamanho da fonte será de fácil visualização para o público com deficiência visual.	Usabilidade
1.4	O “Enter” deverá ser o atalho para o botão Entrar.	Usabilidade
1.5	O site contará com acessibilidade para pessoas com deficiência auditiva.	Usabilidade
1.6	Caso o usuário digite errado a senha ou o e-mail, o sistema deverá exibir um pop-up dizendo: “Usuário ou senha incorretos!”.	Segurança
1.7	Caso o usuário digite a senha incorreta 3 vezes seguidas, o sistema deverá enviar um email de segurança notificando a possível invasão da conta.	Segurança
1.8	No campo senha quando o usuário digitar deverá possuir a senha com caracteres de “password” (*****).	Segurança
1.9	O usuário terá atalhos para o Cadastro (caso ele não tenha) e para o Esqueceu sua Senha (caso ele esqueça) através da Tela de Login.	Usabilidade
1.10	A função deverá possuir as cores seguindo o mesmo padrão do sistema.	Interface
1.11	O campo onde o usuário vai digitar contará com um placeholder para facilitar o seu entendimento do dado necessário a ser informado.	Usabilidade
1.12	O botão deve mudar de cor quando o usuário passar o mouse em cima.	Interface
1.13	O sistema deverá ter uma resposta rápida quando o usuário clicar no botão Entrar.	Desempenho
1.14	Caso o usuário não tenha preenchido todos os campos obrigatórios (*) será exibida a mensagem de erro “Campo obrigatório vazio”.	Segurança
1.15	Após o usuário efetuar o login, será armazenado token de acesso e o refresh token para que somente os usuários autorizados possam trafegar pelo site livremente.	Segurança

Fonte: Produzido pelo autor (2024).

Quadro 5 - Acessar Login Parceiro.

Requisito Funcional		
Nome:	Acessar Login Parceiro	Código: F2

Descrição:	O sistema deve validar o login do parceiro.	
Estimativa de Esforço: 8h	Prioridade: 100 pontos	
Requisitos Não funcionais		
ID NF	Descrição	Categoria
1.1	O tamanho da tela deve se adaptar aos diferentes tamanhos de telas dos dispositivos eletrônicos, sejam eles com uso mobile ou desktop.	Portabilidade
1.2	É necessário o uso do sistema operacional Android ou IOS para executar o requisito funcional em sistemas mobile.	Portabilidade
1.3	O tamanho da fonte será de fácil visualização para o público com deficiência visual.	Usabilidade
1.4	O “Enter” deverá ser o atalho para o botão Entrar.	Usabilidade
1.5	O site contará com acessibilidade para pessoas com deficiência auditiva.	Usabilidade
1.6	Caso o usuário digite errado a senha ou o e-mail, o sistema deverá exibir um pop-up dizendo: “Usuário ou senha incorretos!”.	Segurança
1.7	Caso o usuário digite a senha incorreta 3 vezes seguidas, o sistema deverá enviar um email de segurança notificando a possível invasão da conta.	Segurança
1.8	No campo senha quando o usuário digitar deverá possuir a senha com caracteres de “password” (*****).	Segurança
1.9	O usuário terá atalhos para o Cadastro (caso ele não tenha) e para o Esqueceu sua Senha (caso ele esqueça) através da Tela de Login.	Usabilidade
1.10	A função deverá possuir as cores seguindo o mesmo padrão do sistema.	Interface
1.11	O campo onde o usuário vai digitar contará com um placeholder para facilitar o seu entendimento do dado necessário a ser informado.	Usabilidade
1.12	O botão deve mudar de cor quando o usuário passar o mouse em cima.	Interface
1.13	O sistema deverá ter uma resposta rápida quando o usuário clicar no botão Entrar.	Desempenho
1.14	Caso o usuário não tenha preenchido todos os campos obrigatórios (*) será exibida a mensagem de erro “Campo obrigatório vazio”.	Segurança
1.15	Após o usuário efetuar o login, será armazenado token de acesso e o refresh token para que somente os usuários autorizados possam trafegar pelo site livremente.	Segurança

Fonte: Produzido pelo autor (2024).

Quadro 6 - Acessar Login Administrador.

Requisito Funcional		
Nome:	Acessar Login Administrador	Código: F3
Descrição:	O sistema deve validar o login do administrador.	
Estimativa de Esforço: 8h	Prioridade: 100 pontos	
Requisitos Não funcionais		
ID NF	Descrição	Categoria
1.1	O tamanho da tela deve se adaptar aos diferentes tamanhos de telas dos dispositivos eletrônicos, sejam eles com uso mobile ou desktop.	Portabilidade
1.2	É necessário o uso do sistema operacional Android ou IOS para executar o requisito funcional em sistemas mobile.	Portabilidade
1.3	O tamanho da fonte será de fácil visualização para o público com deficiência visual.	Usabilidade

1.4	O “Enter” deverá ser o atalho para o botão Entrar.	Usabilidade
1.5	O site contará com acessibilidade para pessoas com deficiência auditiva.	Usabilidade
1.6	Caso o usuário digite errado a senha ou o e-mail, o sistema deverá exibir um pop-up dizendo: “Usuário ou senha incorretos!”.	Segurança
1.7	Caso o usuário digite a senha incorreta 3 vezes seguidas, o sistema deverá enviar um email de segurança notificando a possível invasão da conta.	Segurança
1.8	No campo senha quando o usuário digitar deverá possuir a senha com caracteres de “password” (*****).	Segurança
1.9	O usuário terá atalhos para o Cadastro (caso ele não tenha) e para o Esqueceu sua Senha (caso ele esqueça) através da Tela de Login.	Usabilidade
1.10	A função deverá possuir as cores seguindo o mesmo padrão do sistema.	Interface
1.11	O campo onde o usuário vai digitar contará com um placeholder para facilitar o seu entendimento do dado necessário a ser informado.	Usabilidade
1.12	O botão deve mudar de cor quando o usuário passar o mouse em cima.	Interface
1.13	O sistema deverá ter uma resposta rápida quando o usuário clicar no botão Entrar.	Desempenho
1.14	Caso o usuário não tenha preenchido todos os campos obrigatórios (*) será exibida a mensagem de erro “Campo obrigatório vazio”.	Segurança
1.15	Após o usuário efetuar o login, será armazenado token de acesso e o refresh token para que somente os usuários autorizados possam trafegar pelo site livremente.	Segurança

Fonte: Produzido pelo autor (2024).

Quadro 7 - Redefinir Senha Cliente.

Requisito Funcional		
Nome:	Redefinir Senha Cliente	Código: F4
Descrição:	O sistema deve redefinir a senha do cliente.	
Estimativa de Esforço: 8h	Prioridade: 100 pontos	
Requisitos Não funcionais		
ID NF	Descrição	Categoria
1.1	O tamanho da tela deve se adaptar aos diferentes tamanhos de telas dos dispositivos eletrônicos, sejam eles com uso mobile ou desktop.	Portabilidade
1.2	É necessário o uso do sistema operacional Android ou IOS para executar o requisito funcional em sistemas mobile.	Portabilidade
1.3	O tamanho da fonte será de fácil visualização para o público com deficiência visual.	Usabilidade
1.4	O “Enter” deverá ser o atalho para o botão Enviar.	Usabilidade
1.5	O site contará com acessibilidade para pessoas com deficiência auditiva.	Usabilidade
1.6	Caso o usuário digite um e-mail que não esteja cadastrado, o sistema deverá exibir um pop-up dizendo: “Usuário não cadastrado no sistema!”.	Segurança
1.7	A função deverá possuir as cores seguindo o mesmo padrão do sistema.	Interface
1.8	O campo onde o usuário vai digitar contará com um placeholder para facilitar o seu entendimento do dado necessário a ser informado.	Usabilidade
1.9	O botão deve mudar de cor quando o usuário passar o mouse em cima.	Interface

1.10	O sistema deverá ter uma resposta rápida quando o usuário clicar no botão Enviar.	Desempenho
1.11	Caso o usuário não tenha preenchido todos os campos obrigatórios (*) será exibida a mensagem de erro “Campo obrigatório vazio”.	Segurança
1.12	Após o usuário digitar um e-mail válido e clicar em Enviar, enviar um e-mail para o usuário redefinir sua senha.	Usabilidade

Fonte: Produzido pelo autor (2024).

Quadro 8 - Redefinir Senha Parceiro.

Requisito Funcional		
Nome:	Redefinir Senha Parceiro	Código: F5
Descrição:	O sistema deve redefinir a senha do parceiro.	
Estimativa de Esforço: 8h	Prioridade: 100 pontos	
Requisitos Não funcionais		
ID NF	Descrição	Categoria
1.1	O tamanho da tela deve se adaptar aos diferentes tamanhos de telas dos dispositivos eletrônicos, sejam eles com uso mobile ou desktop.	Portabilidade
1.2	É necessário o uso do sistema operacional Android ou IOS para executar o requisito funcional em sistemas mobile.	Portabilidade
1.3	O tamanho da fonte será de fácil visualização para o público com deficiência visual.	Usabilidade
1.4	O “Enter” deverá ser o atalho para o botão Enviar.	Usabilidade
1.5	O site contará com acessibilidade para pessoas com deficiência auditiva.	Usabilidade
1.6	Caso o usuário digite um e-mail que não esteja cadastrado, o sistema deverá exibir um pop-up dizendo: “Usuário não cadastrado no sistema!”.	Segurança
1.7	A função deverá possuir as cores seguindo o mesmo padrão do sistema.	Interface
1.8	O campo onde o usuário vai digitar contará com um placeholder para facilitar o seu entendimento do dado necessário a ser informado.	Usabilidade
1.9	O botão deve mudar de cor quando o usuário passar o mouse em cima.	Interface
1.10	O sistema deverá ter uma resposta rápida quando o usuário clicar no botão Enviar.	Desempenho
1.11	Caso o usuário não tenha preenchido todos os campos obrigatórios (*) será exibida a mensagem de erro “Campo obrigatório vazio”.	Segurança
1.12	Após o usuário digitar um e-mail válido e clicar em Enviar, enviar um e-mail para o usuário redefinir sua senha.	Usabilidade

Fonte: Produzido pelo autor (2024).

Quadro 9 - Redefinir Senha Administrador.

Requisito Funcional		
Nome:	Redefinir Senha Administrador	Código: F6
Descrição:	O sistema deve redefinir a senha do administrador.	
Estimativa de Esforço: 8h	Prioridade: 100 pontos	
Requisitos Não funcionais		
ID NF	Descrição	Categoria

1.1	O tamanho da tela deve se adaptar aos diferentes tamanhos de telas dos dispositivos eletrônicos, sejam eles com uso mobile ou desktop.	Portabilidade
1.2	É necessário o uso do sistema operacional Android ou IOS para executar o requisito funcional em sistemas mobile.	Portabilidade
1.3	O tamanho da fonte será de fácil visualização para o público com deficiência visual.	Usabilidade
1.4	O “Enter” deverá ser o atalho para o botão Enviar.	Usabilidade
1.5	O site contará com acessibilidade para pessoas com deficiência auditiva.	Usabilidade
1.6	Caso o usuário digite um e-mail que não esteja cadastrado, o sistema deverá exibir um pop-up dizendo: “Usuário não cadastrado no sistema!”.	Segurança
1.7	A função deverá possuir as cores seguindo o mesmo padrão do sistema.	Interface
1.8	O campo onde o usuário vai digitar contará com um placeholder para facilitar o seu entendimento do dado necessário a ser informado.	Usabilidade
1.9	O botão deve mudar de cor quando o usuário passar o mouse em cima.	Interface
1.10	O sistema deverá ter uma resposta rápida quando o usuário clicar no botão Enviar.	Desempenho
1.11	Caso o usuário não tenha preenchido todos os campos obrigatórios (*) será exibida a mensagem de erro “Campo obrigatório vazio”.	Segurança
1.12	Após o usuário digitar um e-mail válido e clicar em Enviar, enviar um e-mail para o usuário redefinir sua senha.	Usabilidade

Fonte: Produzido pelo autor (2024).

Quadro 10 - Recuperar Senha Cliente.

Requisito Funcional		
Nome:	Recuperar Senha Cliente	Código: F7
Descrição:	O sistema deve recuperar a senha do cliente.	
Estimativa de Esforço: 8h	Prioridade: 100 pontos	
Requisitos Não funcionais		
ID NF	Descrição	Categoria
1.1	O tamanho da tela deve se adaptar aos diferentes tamanhos de telas dos dispositivos eletrônicos, sejam eles com uso mobile ou desktop.	Portabilidade
1.2	É necessário o uso do sistema operacional Android ou IOS para executar o requisito funcional em sistemas mobile.	Portabilidade
1.3	O tamanho da fonte será de fácil visualização para o público com deficiência visual.	Usabilidade
1.4	O “Enter” deverá ser o atalho para o botão Salvar.	Usabilidade
1.5	O site contará com acessibilidade para pessoas com deficiência auditiva.	Usabilidade
1.6	Caso o usuário digite uma senha muito fraca, o sistema deve informar para ele colocar uma senha composta por caracteres especiais, números e no mínimo 8 caracteres.	Segurança
1.7	A função deverá possuir as cores seguindo o mesmo padrão do sistema.	Interface
1.8	O campo onde o usuário vai digitar contará com um placeholder para facilitar o seu entendimento do dado necessário a ser informado.	Usabilidade
1.9	O botão deve mudar de cor quando o usuário passar o mouse em cima.	Interface

1.10	O sistema deverá ter uma resposta rápida quando o usuário clicar no botão Salvar.	Desempenho
1.11	Caso o usuário não tenha preenchido todos os campos obrigatórios (*) será exibida a mensagem de erro “Campo obrigatório vazio”.	Segurança
1.12	O usuário deverá escolher senhas que sejam iguais nos dois campos, caso não sejam senhas iguais, o sistema deve informar "As senhas não coincidem!"	Usabilidade
1.13	Após o usuário digitar as senhas e elas coincidirem e serem senhas fortes o sistema deverá salvar a nova senha e direcionar para o menu com o login efetuado.	Usabilidade

Fonte: Produzido pelo autor (2024).

Quadro 11 - Recuperar Senha Parceiro.

Requisito Funcional		
Nome:	Recuperar Senha Parceiro	Código: F8
Descrição:	O sistema deve redefinir a senha do parceiro.	
Estimativa de Esforço: 8h	Prioridade: 100 pontos	
Requisitos Não funcionais		
ID NF	Descrição	Categoria
1.1	O tamanho da tela deve se adaptar aos diferentes tamanhos de telas dos dispositivos eletrônicos, sejam eles com uso mobile ou desktop.	Portabilidade
1.2	É necessário o uso do sistema operacional Android ou IOS para executar o requisito funcional em sistemas mobile.	Portabilidade
1.3	O tamanho da fonte será de fácil visualização para o público com deficiência visual.	Usabilidade
1.4	O “Enter” deverá ser o atalho para o botão Salvar.	Usabilidade
1.5	O site contará com acessibilidade para pessoas com deficiência auditiva.	Usabilidade
1.6	Caso o usuário digite uma senha muito fraca, o sistema deve informar para ele colocar uma senha composta por caracteres especiais, números e no mínimo 8 caracteres.	Segurança
1.7	A função deverá possuir as cores seguindo o mesmo padrão do sistema.	Interface
1.8	O campo onde o usuário vai digitar contará com um placeholder para facilitar o seu entendimento do dado necessário a ser informado.	Usabilidade
1.9	O botão deve mudar de cor quando o usuário passar o mouse em cima.	Interface
1.10	O sistema deverá ter uma resposta rápida quando o usuário clicar no botão Salvar.	Desempenho
1.11	Caso o usuário não tenha preenchido todos os campos obrigatórios (*) será exibida a mensagem de erro “Campo obrigatório vazio”.	Segurança
1.12	O usuário deverá escolher senhas que sejam iguais nos dois campos, caso não sejam senhas iguais, o sistema deve informar "As senhas não coincidem!"	Usabilidade
1.13	Após o usuário digitar as senhas e elas coincidirem e serem senhas fortes o sistema deverá salvar a nova senha e direcionar para o menu com o login efetuado.	Usabilidade

Fonte: Produzido pelo autor (2024).

Quadro 12 - Recuperar Senha Administrador

Requisito Funcional

Nome:	Recuperar Senha Administrador	Código:	F9
Descrição:	O sistema deve redefinir a senha do administrador.		
Estimativa de Esforço: 8h	Prioridade: 100 pontos		
Requisitos Não funcionais			
ID NF	Descrição	Categoria	
1.1	O tamanho da tela deve se adaptar aos diferentes tamanhos de telas dos dispositivos eletrônicos, sejam eles com uso mobile ou desktop.	Portabilidade	
1.2	É necessário o uso do sistema operacional Android ou IOS para executar o requisito funcional em sistemas mobile.	Portabilidade	
1.3	O tamanho da fonte será de fácil visualização para o público com deficiência visual.	Usabilidade	
1.4	O “Enter” deverá ser o atalho para o botão Salvar.	Usabilidade	
1.5	O site contará com acessibilidade para pessoas com deficiência auditiva.	Usabilidade	
1.6	Caso o usuário digite uma senha muito fraca, o sistema deve informar para ele colocar uma senha composta por caracteres especiais, números e no mínimo 8 caracteres.	Segurança	
1.7	A função deverá possuir as cores seguindo o mesmo padrão do sistema.	Interface	
1.8	O campo onde o usuário vai digitar contará com um placeholder para facilitar o seu entendimento do dado necessário a ser informado.	Usabilidade	
1.9	O botão deve mudar de cor quando o usuário passar o mouse em cima.	Interface	
1.10	O sistema deverá ter uma resposta rápida quando o usuário clicar no botão Salvar.	Desempenho	
1.11	Caso o usuário não tenha preenchido todos os campos obrigatórios (*) será exibida a mensagem de erro “Campo obrigatório vazio”.	Segurança	
1.12	O usuário deverá escolher senhas que sejam iguais nos dois campos, caso não sejam senhas iguais, o sistema deve informar "As senhas não coincidem!"	Usabilidade	
1.13	Após o usuário digitar as senhas e elas coincidirem e serem senhas fortes o sistema deverá salvar a nova senha e direcionar para o menu com o login efetuado.	Usabilidade	

Fonte: Produzido pelo autor (2024).

Quadro 13 - Alterar Senha Cliente.

Requisito Funcional			
Nome:	Alterar Senha Cliente	Código:	F10
Descrição:	O sistema deve alterar a senha do cliente.		
Estimativa de Esforço: 8h	Prioridade: 100 pontos		
Requisitos Não funcionais			
ID NF	Descrição	Categoria	
1.1	O tamanho da tela deve se adaptar aos diferentes tamanhos de telas dos dispositivos eletrônicos, sejam eles com uso mobile ou desktop.	Portabilidade	
1.2	É necessário o uso do sistema operacional Android ou IOS para executar o requisito funcional em sistemas mobile.	Portabilidade	
1.3	O tamanho da fonte será de fácil visualização para o público com deficiência visual.	Usabilidade	
1.4	O “Enter” deverá ser o atalho para o botão Salvar.	Usabilidade	

1.5	O site contará com acessibilidade para pessoas com deficiência auditiva.	Usabilidade
1.6	Caso o usuário digite uma senha muito fraca, o sistema deve informar para ele colocar uma senha composta por caracteres especiais, números e no mínimo 8 caracteres.	Segurança
1.7	A função deverá possuir as cores seguindo o mesmo padrão do sistema.	Interface
1.8	O campo onde o usuário vai digitar contará com um placeholder para facilitar o seu entendimento do dado necessário a ser informado.	Usabilidade
1.9	O botão deve mudar de cor quando o usuário passar o mouse em cima.	Interface
1.10	O sistema deverá ter uma resposta rápida quando o usuário clicar no botão Salvar.	Desempenho
1.11	Caso o usuário não tenha preenchido todos os campos obrigatórios (*) será exibida a mensagem de erro "Campo obrigatório vazio".	Segurança
1.12	O usuário deverá digitar a sua senha atual, caso a senha não seja igual a atual, o sistema não deverá efetuar a troca de senha e informar o usuário "Sua senha não foi alterada!"	Usabilidade
1.13	O usuário deverá escolher senhas que sejam iguais nos dois campos, caso não sejam senhas iguais, o sistema deve informar "As senhas não coincidem!"	Usabilidade
1.14	Após o usuário digitar as senhas e elas coincidirem e serem senhas fortes e a senha atual coincidir, o sistema deverá salvar a nova senha e direcionar para o menu com o login efetuado.	Usabilidade

Fonte: Produzido pelo autor (2024).

Quadro 14 - Alterar Senha Parceiro.

Requisito Funcional		
Nome:	Alterar Senha Parceiro	Código: F11
Descrição:	O sistema deve alterar a senha do parceiro.	
Estimativa de Esforço: 8h	Prioridade: 100 pontos	
Requisitos Não funcionais		
ID NF	Descrição	Categoria
1.1	O tamanho da tela deve se adaptar aos diferentes tamanhos de telas dos dispositivos eletrônicos, sejam eles com uso mobile ou desktop.	Portabilidade
1.2	É necessário o uso do sistema operacional Android ou IOS para executar o requisito funcional em sistemas mobile.	Portabilidade
1.3	O tamanho da fonte será de fácil visualização para o público com deficiência visual.	Usabilidade
1.4	O "Enter" deverá ser o atalho para o botão Salvar.	Usabilidade
1.5	O site contará com acessibilidade para pessoas com deficiência auditiva.	Usabilidade
1.6	Caso o usuário digite uma senha muito fraca, o sistema deve informar para ele colocar uma senha composta por caracteres especiais, números e no mínimo 8 caracteres.	Segurança
1.7	A função deverá possuir as cores seguindo o mesmo padrão do sistema.	Interface
1.8	O campo onde o usuário vai digitar contará com um placeholder para facilitar o seu entendimento do dado necessário a ser informado.	Usabilidade
1.9	O botão deve mudar de cor quando o usuário passar o mouse em cima.	Interface

1.10	O sistema deverá ter uma resposta rápida quando o usuário clicar no botão Salvar.	Desempenho
1.11	Caso o usuário não tenha preenchido todos os campos obrigatórios (*) será exibida a mensagem de erro “Campo obrigatório vazio”.	Segurança
1.12	O usuário deverá digitar a sua senha atual, caso a senha não seja igual a atual, o sistema não deverá efetuar a troca de senha e informar o usuário "Sua senha não foi alterada!"	Usabilidade
1.13	O usuário deverá escolher senhas que sejam iguais nos dois campos, caso não sejam senhas iguais, o sistema deve informar "As senhas não coincidem!"	Usabilidade
1.14	Após o usuário digitar as senhas e elas coincidirem e serem senhas fortes e a senha atual coincidir, o sistema deverá salvar a nova senha e direcionar para o menu com o login efetuado.	Usabilidade

Fonte: Produzido pelo autor (2024).

Quadro 15 - Alterar Senha Administrador.

Requisito Funcional		
Nome:	Alterar Senha Administrador	Código: F12
Descrição:	O sistema deve alterar a senha do administrador.	
Estimativa de Esforço: 8h	Prioridade: 100 pontos	
Requisitos Não funcionais		
ID NF	Descrição	Categoria
1.1	O tamanho da tela deve se adaptar aos diferentes tamanhos de telas dos dispositivos eletrônicos, sejam eles com uso mobile ou desktop.	Portabilidade
1.2	É necessário o uso do sistema operacional Android ou IOS para executar o requisito funcional em sistemas mobile.	Portabilidade
1.3	O tamanho da fonte será de fácil visualização para o público com deficiência visual.	Usabilidade
1.4	O “Enter” deverá ser o atalho para o botão Salvar.	Usabilidade
1.5	O site contará com acessibilidade para pessoas com deficiência auditiva.	Usabilidade
1.6	Caso o usuário digite uma senha muito fraca, o sistema deve informar para ele colocar uma senha composta por caracteres especiais, números e no mínimo 8 caracteres.	Segurança
1.7	A função deverá possuir as cores seguindo o mesmo padrão do sistema.	Interface
1.8	O campo onde o usuário vai digitar contará com um placeholder para facilitar o seu entendimento do dado necessário a ser informado.	Usabilidade
1.9	O botão deve mudar de cor quando o usuário passar o mouse em cima.	Interface
1.10	O sistema deverá ter uma resposta rápida quando o usuário clicar no botão Salvar.	Desempenho
1.11	Caso o usuário não tenha preenchido todos os campos obrigatórios (*) será exibida a mensagem de erro “Campo obrigatório vazio”.	Segurança
1.12	O usuário deverá digitar a sua senha atual, caso a senha não seja igual a atual, o sistema não deverá efetuar a troca de senha e informar o usuário "Sua senha não foi alterada!"	Usabilidade
1.13	O usuário deverá escolher senhas que sejam iguais nos dois campos, caso não sejam senhas iguais, o sistema deve informar "As senhas não coincidem!"	Usabilidade
1.14	Após o usuário digitar as senhas e elas coincidirem e serem senhas fortes e a senha atual coincidir, o sistema deverá salvar a nova senha e direcionar para o menu com o login efetuado.	Usabilidade

Fonte: Produzido pelo autor (2024).

Quadro 16 - Acessar Tela Inicial Cliente.

Requisito Funcional		
Nome:	Acessar Tela Inicial Cliente	Código: F13
Descrição:	Será possível ter acesso à tela inicial do site, onde terão os cupons, carousel com itens, ofertas e muito mais.	
Estimativa de Esforço: 8h	Prioridade: Alta	
Requisitos Não funcionais		
ID NF	Descrição	Categoria
1.1	O tamanho da tela deve se adaptar aos diferentes tamanhos de telas dos dispositivos eletrônicos, sejam eles com uso mobile ou desktop.	Portabilidade
1.2	É necessário o uso do sistema operacional Android ou IOS para executar o requisito funcional em sistemas mobile	Portabilidade
1.3	A função deverá possuir as cores seguindo o mesmo padrão do sistema.	Interface
1.4	O tamanho da fonte será de fácil visualização para o público com deficiência visual.	Usabilidade
1.5	O site contará com acessibilidade para pessoas com deficiência auditiva.	Usabilidade
1.6	Capacidade do requisito funcional de recuperar dados após uma falha do sistema ou queda de energia;	Confiabilidade
1.7	O requisito funcional deverá estar disponível 24h por dia para o usuário.	Confiabilidade
1.8	A função deverá possuir imagens interativas com boa qualidade, com um carrossel de itens com ofertas especiais.	Interface
1.9	Os preços com ofertas deverão possuir cores diferentes e chamativas, para atrair clientes do site.	Interface
1.10	A tela apresentará uma barra de rolagem vertical para permitir a visualização de todos os cupons e ofertas do site.	Interface
1.11	O sistema deverá ter uma resposta rápida ao acessar itens do carrossel.	Interface

Fonte: Produzido pelo autor (2024).

Quadro 17 - Acessar Menu Inicial Parceiro.

Requisito Funcional		
Nome:	Acessar Menu Inicial Parceiro	Código: F14
Descrição:	Será possível ter acesso ao menu principal do parceiro, onde terão opções de consultas, cadastros, consultas, relatórios e dashboard para os parceiros.	
Estimativa de Esforço: 8h	Prioridade: Alta	
Requisitos Não funcionais		
ID NF	Descrição	Categoria
1.1	O tamanho da tela deve se adaptar aos diferentes tamanhos de telas dos dispositivos eletrônicos, sejam eles com uso mobile ou desktop.	Portabilidade
1.2	É necessário o uso do sistema operacional Android ou IOS para executar o requisito funcional em sistemas mobile	Portabilidade
1.3	A função deverá possuir as cores seguindo o mesmo padrão do sistema.	Interface
1.4	O tamanho da fonte será de fácil visualização para o público com deficiência visual.	Usabilidade

1.5	O site contará com acessibilidade para pessoas com deficiência auditiva.	Usabilidade
1.6	Capacidade do requisito funcional de recuperar dados após uma falha do sistema ou queda de energia;	Confiabilidade
1.7	O requisito funcional deverá estar disponível 24h por dia para o usuário.	Confiabilidade
1.8	A função deverá possuir imagens interativas com boa qualidade, com um carrossel de itens com ofertas especiais.	Interface
1.9	O menu deverá ter acesso a todas as permissões do parceiro, sendo elas cadastros, consultas, relatórios e dashboard.	Interface

Fonte: Produzido pelo autor (2024).

Quadro 18 - Acessar Menu Inicial Administrador.

Requisito Funcional		
Nome:	Acessar Menu Inicial Administrador	Código: F15
Descrição:	Será possível ter acesso ao menu principal do administrador, onde terão opções de consultas, cadastros, relatórios e dashboard para os parceiros.	
Estimativa de Esforço: 8h	Prioridade: Alta	
Requisitos Não funcionais		
ID NF	Descrição	Categoria
1.1	O tamanho da tela deve se adaptar aos diferentes tamanhos de telas dos dispositivos eletrônicos, sejam eles com uso mobile ou desktop.	Portabilidade
1.2	É necessário o uso do sistema operacional Android ou IOS para executar o requisito funcional em sistemas mobile	Portabilidade
1.3	A função deverá possuir as cores seguindo o mesmo padrão do sistema.	Interface
1.4	O tamanho da fonte será de fácil visualização para o público com deficiência visual.	Usabilidade
1.5	O site contará com acessibilidade para pessoas com deficiência auditiva.	Usabilidade
1.6	Capacidade do requisito funcional de recuperar dados após uma falha do sistema ou queda de energia;	Confiabilidade
1.7	O requisito funcional deverá estar disponível 24h por dia para o usuário.	Confiabilidade
1.8	A função deverá possuir imagens interativas com boa qualidade, com um carrossel de itens com ofertas especiais.	Interface
1.9	O menu deverá ter acesso a todas as permissões do parceiro, sendo elas cadastros, consultas, relatórios e dashboard.	Interface

Fonte: Produzido pelo autor (2024).

Quadro 19 - Cadastrar Categorias.

Requisito Funcional		
Nome:	Cadastrar Categorias	Código: F16
Descrição:	O administrador poderá cadastrar a categoria do seu negócio.	
Estimativa de Esforço: 5h	Prioridade: Média	
Requisitos Não funcionais		
ID NF	Descrição	Categoria
1.1	As informações da categoria que deverão ser armazenadas pela função no banco de dados são: categoria e descrição.	Especificação

1.2	O sistema deve registrar e responder de forma rápida ao usuário na sua interface sobre o andamento do cadastro.	Desempenho
1.3	O campo onde o usuário vai digitar contará com um placeholder para facilitar o seu entendimento do dado necessário a ser informado.	Usabilidade
1.4	O único botão presente na tela será o “Cadastrar”.	Interface
1.5	O botão da funcionalidade terá cores que o destacam do seu fundo.	Interface
1.6	Os botões devem mudar de cor quando o usuário passar o mouse em cima.	Interface
1.7	O tamanho da fonte será de fácil visualização para o público com deficiência visual.	Usabilidade
1.8	A interface contará com acessibilidade para pessoas com deficiência auditiva.	Usabilidade
1.9	O campo de texto de descrição contará com no máximo 300 caracteres, que podem ser informados pelo usuário para sabermos mais sobre sua categoria.	Interface
1.10	Caso o usuário não tenha preenchido todos os campos obrigatórios (*) será exibida a mensagem de erro “Campo obrigatório vazio”.	Usabilidade
1.11	O campo de descrição não será obrigatório para o usuário informar.	Usabilidade
1.12	O tamanho da tela deve se adaptar aos diferentes tamanhos de telas dos dispositivos eletrônicos, sejam eles com uso mobile ou desktop.	Portabilidade
1.13	A função deverá possuir as cores seguindo o mesmo padrão do sistema.	Interface
1.14	O “Enter” deverá ser o atalho para o botão “Cadastrar”.	Interface
1.15	Capacidade do requisito funcional de recuperar dados após uma falha do sistema ou queda de energia.	Confiabilidade

Fonte: Produzido pelo autor (2024).

Quadro 20 - Cadastrar Clientes.

Requisito Funcional		
Nome:	Cadastrar Clientes	Código: F17
Descrição:	O sistema deve registrar todas as informações possíveis sobre o cliente.	
Estimativa de Esforço: 8h	Prioridade: Alta	
Requisitos Não funcionais		
ID NF	Descrição	Categoria
1.1	O tamanho da tela deve se adaptar aos diferentes tamanhos de telas dos dispositivos eletrônicos, sejam eles com uso mobile ou desktop.	Portabilidade
1.2	É necessário o uso do sistema operacional Android ou IOS para executar o requisito funcional em sistemas mobile.	Portabilidade
1.3	Caso o email ou senha digitado para a cadastrar o cliente não esteja correto aparecerá uma pop-up de erro na tela.	Interface
1.4	O único botão presente na tela será o “Cadastrar”, tendo cores que se distinguem do seu fundo para o destacá-lo.	Interface
1.5	O botão deve mudar de cor quando o usuário passar o mouse em cima.	Interface
1.6	A função deverá possuir as cores seguindo o mesmo padrão do sistema.	Interface
1.7	O tamanho da fonte será de fácil visualização para o público com deficiência visual.	Usabilidade
1.8	O site contará com acessibilidade para pessoas com deficiência auditiva.	Usabilidade

1.9	O "Enter" deverá ser o atalho para o botão Cadastrar.	Usabilidade
1.10	O campo onde o usuário vai digitar contará com um placeholder para facilitar o seu entendimento do dado necessário a ser informado.	Usabilidade
1.11	Será obrigatório senhas com comprimento maior ou igual a 8 caracteres e caracteres especiais.	Segurança
1.12	Os dados inseridos pelo usuário no requisito funcional devem estar criptografados.	Segurança
1.13	Dos dados armazenados pela função, serão obrigatórios os seguintes dados: nome, email, senha, data de nascimento, telefone, CPF e as informações de endereço (cep, estado, cidade, bairro, logradouro, número).	Segurança
1.14	O sistema deverá ter uma resposta rápida ao cadastrar.	Desempenho
1.15	Garantir que os dados registrados pelo sistema não serão duplicados.	Confiabilidade
1.16	Capacidade do requisito funcional de recuperar dados após uma falha do sistema ou queda de energia;	Confiabilidade
1.17	O requisito funcional deverá estar disponível 24h por dia para o usuário.	Confiabilidade
1.18	As informações do cliente que deverão ser armazenadas pela função no banco de dados são: id, nome, email, senha, data de nascimento, CPF, telefone, status e as informações do endereço.	Especificação

Fonte: Produzido pelo autor (2024).

Quadro 21 - Cadastrar Parceiros

Requisito Funcional		
Nome:	Cadastrar Parceiros	Código: F18
Descrição:	O sistema deve registrar todas as informações possíveis sobre o parceiro.	
Estimativa de Esforço: 8h	Prioridade: Alta	
Requisitos Não funcionais		
ID NF	Descrição	Categoria
1.1	O tamanho da tela deve se adaptar aos diferentes tamanhos de telas dos dispositivos eletrônicos, sejam eles com uso mobile ou desktop.	Portabilidade
1.2	É necessário o uso do sistema operacional Android ou IOS para executar o requisito funcional em sistemas mobile.	Portabilidade
1.3	Caso o email ou senha digitado para a cadastrar o parceiro não esteja correto aparecerá uma pop-up de erro na tela.	Interface
1.4	O único botão presente na tela será o "Cadastrar", tendo cores que se distinguem do seu fundo para o destacá-lo.	Interface
1.5	O botão deve mudar de cor quando o usuário passar o mouse em cima.	Interface
1.6	A função deverá possuir as cores seguindo o mesmo padrão do sistema.	Interface
1.7	O tamanho da fonte será de fácil visualização para o público com deficiência visual.	Usabilidade
1.8	O site contará com acessibilidade para pessoas com deficiência auditiva.	Usabilidade
1.9	O "Enter" deverá ser o atalho para o botão Cadastrar.	Usabilidade
1.10	O campo onde o usuário vai digitar contará com um placeholder para facilitar o seu entendimento do dado necessário a ser informado.	Usabilidade

1.11	Será obrigatório senhas com comprimento maior ou igual a 8 caracteres e caracteres especiais.	Segurança
1.12	Os dados inseridos pelo usuário no requisito funcional devem estar criptografados.	Segurança
1.13	Dos dados armazenados pela função, serão obrigatórios os seguintes dados: CNPJ, razão social, nome fantasia, telefone, e-mail e as informações de endereço (cep, estado, cidade, bairro, logradouro, número).	Segurança
1.14	O sistema deverá ter uma resposta rápida ao cadastrar.	Desempenho
1.15	Garantir que os dados registrados pelo sistema não serão duplicados.	Confiabilidade
1.16	Capacidade do requisito funcional de recuperar dados após uma falha do sistema ou queda de energia;	Confiabilidade
1.17	O requisito funcional deverá estar disponível 24h por dia para o usuário.	Confiabilidade
1.18	As informações do parceiro que deverão ser armazenadas pela função no banco de dados são: id, CNPJ, razão social, nome fantasia, telefone, e-mail e as informações de endereço.	Especificação

Fonte: Produzido pelo autor (2024).

Quadro 22 - Cadastrar Cupons.

Requisito Funcional		
Nome:	Cadastrar Cupons	Código: F19
Descrição:	Registrar os cupons fornecidos pelos parceiros.	
Estimativa de Esforço: 5h	Prioridade: Alta	
Requisitos Não funcionais		
ID NF	Descrição	Categoria
1.1	Garantir que os dados registrados pelo sistema não serão duplicados.	Confiabilidade
1.2	O cadastro de cupom deve estar disponível para registro 24h por dia.	Desempenho
1.3	A confirmação do registro do cupom deve ser em segundos.	Desempenho
1.4	O campo onde o usuário vai digitar contará com um placeholder para facilitar o seu entendimento do dado necessário a ser informado.	Usabilidade
1.5	A função deverá possuir as cores seguindo o mesmo padrão do sistema.	Interface
1.6	O único botão presente na tela será o “Cadastrar”, tendo cores que se distinguem do seu fundo para o destacá-lo.	Interface
1.7	Os dados salvos pela função serão: nome do cupom, quantidade, empresa, unidade, imagens, valor sem desconto, porcentagem de desconto, status, valor de venda, descrição, categoria e informações adicionais.	Interface
1.8	A imagem do cupom poderá ser importada para a funcionalidade.	Usabilidade
1.9	O campo de texto de descrição contará com no máximo 300 caracteres, que podem ser informados pelo usuário para sabermos mais sobre sua categoria.	Interface
1.10	Todos os campos informados corretamente aparecerá um pop - up de cadastros realizado com sucesso.	Interface
1.11	Caso o usuário não tenha preenchido todos os campos obrigatórios (*) será exibida a mensagem de erro “Campo obrigatório vazio”.	Especificação
1.12	Garantir que os dados registrados pelo sistema não serão duplicados.	Confiabilidade
1.13	O “Enter” deverá ser o atalho para o botão Cadastrar.	Usabilidade

1.14	É necessário o uso do sistema operacional Android ou IOS para executar o requisito funcional em sistemas mobile.	Portabilidade
1.15	O tamanho da tela deve se adaptar aos diferentes tamanhos de telas dos dispositivos eletrônicos, sejam eles com uso mobile ou desktop.	Portabilidade
1.16	O tamanho da fonte será de fácil visualização para o público com deficiência visual.	Usabilidade
1.17	O site contará com acessibilidade para pessoas com deficiência auditiva.	Usabilidade

Fonte: Produzido pelo autor (2024).

Quadro 23 - Cadastrar Agendas.

Requisito Funcional		
Nome:	Cadastrar Agendas	Código: F20
Descrição:	O sistema deve registrar todas as informações possíveis de uma agenda.	
Estimativa de Esforço: 8h	Prioridade: Alta	
Requisitos Não funcionais		
ID NF	Descrição	Categoria
1.1	As informações do parceiro que deverão ser armazenadas pela função no banco de dados são: id, data inicial, data final, horário inicial, horário final, cupom, parceiro e quantidade.	Especificação
1.2	Caso o usuário não tenha preenchido todos os campos obrigatórios (*) será exibida a mensagem de erro “Campo obrigatório vazio”.	Especificação
1.3	O campo de inserção da data contará com um ícone de calendário ao lado que funcionará como um atalho para preencher a data desejada.	Especificação
1.4	O sistema deve possuir uma interface gráfica com ícones representativos para facilitar a sua utilização.	Interface
1.5	O requisito funcional deverá estar disponível 24h por dia para o usuário.	Confiabilidade
1.6	A função deverá possuir as cores seguindo o mesmo padrão do sistema.	Interface
1.7	A função deverá apresentar o botão: Cadastrar.	Interface
1.8	O “Enter” deverá ser o atalho para o botão Cadastrar.	Usabilidade
1.9	Garantir que os dados registrados pelo sistema não serão duplicados.	Confiabilidade
1.10	O tamanho da fonte será de fácil visualização para o público com deficiência visual.	Usabilidade
1.11	O site contará com acessibilidade para pessoas com deficiência auditiva.	Usabilidade
1.12	Caso as informações digitadas para a cadastrar a agenda não esteja correta aparecerá uma pop-up de efetuado com sucesso ou se houve um erro.	Segurança
1.13	Dos dados armazenados pela função, serão obrigatórios os seguintes dados: data inicial, data final, horário inicial, horário final, cupom, parceiro e quantidade.	Segurança
1.14	O campo onde o usuário vai digitar contará com um placeholder para facilitar o seu entendimento do dado necessário a ser informado.	Usabilidade
1.15	A função deverá possuir as cores seguindo o mesmo padrão do sistema.	Interface
1.16	O sistema deverá ter uma resposta rápida ao cadastro das agendas.	Desempenho

Fonte: Produzido pelo autor (2024).

Quadro 24 - Favoritar Cupons.

Requisito Funcional		
Nome:	Favoritar Cupons	Código: F21
Descrição:	O sistema deve armazenar os cupons favoritados.	
Estimativa de Esforço: 8h	Prioridade: Alta	
Requisitos Não funcionais		
ID NF	Descrição	Categoria
1.1	O sistema deve possuir uma interface gráfica com ícones representativos para facilitar a sua utilização, no caso ao lado do cupom na tela inicial aparecer um coração não preenchido.	Interface
1.2	Caso o usuário clique no coração no cupom, o cupom deverá ser favoritado.	Usabilidade
1.3	O requisito funcional deverá estar disponível 24h por dia para o usuário.	Confiabilidade
1.4	A função deverá possuir as cores seguindo o mesmo padrão do sistema.	Interface
1.5	O tamanho da fonte será de fácil visualização para o público com deficiência visual.	Usabilidade
1.6	O site contará com acessibilidade para pessoas com deficiência auditiva.	Usabilidade
1.7	Após o usuário clicar em favoritar o cupom, o coração deverá ser preenchido.	Usabilidade

Fonte: Produzido pelo autor (2024).

Quadro 25 - Comprar Cupons.

Requisito Funcional		
Nome:	Comprar Cupons	Código: F22
Descrição:	Tela responsável por trazer as informações e cupons adicionados ao carrinho, possuindo o resumo da compra.	
Estimativa de Esforço: 10h	Prioridade: Alta	
Requisitos Não funcionais		
ID NF	Descrição	Categoria
1.1	O tamanho da fonte será de fácil visualização para o público com deficiência visual.	Usabilidade
1.2	O site contará com acessibilidade para pessoas com deficiência auditiva.	Usabilidade
1.3	A função deverá possuir as cores seguindo o mesmo padrão do sistema.	Interface
1.4	As informações da venda que deverão ser armazenadas pela função no banco de dados são: id, data, horário, parceiro, cliente e cupons.	Especificação
1.5	A função deverá apresentar o botão: Ir ao Pagamento.	Interface
1.6	O "Enter" deverá ser o atalho para o botão Ir ao Pagamento.	Interface
1.7	A GRID apresentará uma barra de rolagem vertical para permitir a visualização de todas as linhas da GRID.	Interface
1.8	O sistema deve possuir uma interface gráfica com ícones representativos para facilitar a sua utilização.	Interface
1.9	O cliente poderá manipular os cupons do seu carrinho, podendo remover os cupons e adicionar quando quiser.	Usabilidade
1.10	O cliente conseguirá alterar a quantidade de cupons que quer comprar no resumo da compra.	Usabilidade

1.11	O sistema irá conseguir calcular os valores de subtotal e total a pagar do cliente, conseguindo calcular em tempo real, conforme a quantidade que o cliente desejar.	Desempenho
1.12	É necessário o uso do sistema operacional Android ou IOS para executar o requisito funcional em sistemas mobile;	Portabilidade
1.13	Capacidade do requisito funcional de recuperar dados após uma falha do sistema ou queda de energia.	Confiabilidade
1.14	O botão deve mudar de cor quando o usuário passar o mouse em cima.	Interface

Fonte: Produzido pelo autor (2024).

Quadro 26 - Receber Vendas.

Requisito Funcional		
Nome:	Receber Vendas	Código: F23
Descrição:	Receber vendas realizadas pelo sistema.	
Estimativa de Esforço: 5h	Prioridade: Alta	
Requisitos Não funcionais		
ID NF	Descrição	Categoria
1.1	O requisito funcional deve estar disponível 24h por dia para o usuário.	Desempenho
1.2	Garantir que os dados registrados pelo sistema não serão duplicados.	Confiabilidade
1.3	O tamanho da tela deve se adaptar aos diferentes tamanhos de telas dos dispositivos eletrônicos, sejam eles com uso mobile ou desktop.	Portabilidade
1.4	Capacidade do requisito funcional de recuperar dados após uma falha do sistema ou queda de energia.	Confiabilidade
1.5	O sistema deverá ter uma resposta rápida ao envio da confirmação do recebimento de vendas.	Desempenho
1.6	Os dados armazenados pela função serão: número do cartão, código de segurança, data de validade, ano, CPF do titular do cartão, código do cupom, nome escrito no cartão, telefone e parcelas, caso seja pix, armazenar as informações de código.	Usabilidade
1.7	Caso o usuário não tenha preenchido todos os campos obrigatórios (*) será exibida a mensagem de erro “Campo obrigatório vazio”.	Especificação
1.8	A funcionalidade contará com o botão: Efetuar pagamento.	Interface
1.9	O botão deve mudar de cor quando o usuário passar o mouse em cima.	Interface
1.10	A função deverá possuir as cores seguindo o mesmo padrão do sistema.	Interface
1.11	O tamanho da fonte será de fácil visualização para o público com deficiência visual.	Usabilidade
1.12	O site contará com acessibilidade para pessoas com deficiência auditiva.	Usabilidade
1.13	O “Enter” deverá ser o atalho para o botão Efetuar pagamento.	Usabilidade
1.14	O campo onde o usuário vai digitar contará com um placeholder para facilitar o seu entendimento do dado necessário a ser informado.	Usabilidade
1.15	A funcionalidade contará com a opção de pagamento por pix, um pop - up aparecerá para o usuário poder usar o QR code ou copiar seu código.	Usabilidade
1.16	Caso o pagamento esteja correto aparecerá um pop-up “Transação realizada com sucesso”.	Interface

1.17	É necessário o uso do sistema operacional Android ou IOS para executar o requisito funcional em sistemas mobile;	Portabilidade
1.18	O campo de “data de validade” contará com um atalho, um ícone de agenda, para o usuário preencher o campo. Não podendo informar uma data menor da atual.	Usabilidade
1.19	O campo de “Ano” contará com um atalho, o ícone de uma seta. Não podendo ser informado um ano menor do atual.	Usabilidade

Fonte: Produzido pelo autor (2024).

Quadro 27 - Emitir Voucher.

Requisito Funcional		
Nome:	Emitir Voucher	Código: F24
Descrição:	A função tem como objetivo gerar voucher.	
Estimativa de Esforço: 10h	Prioridade: Alta	
Requisitos Não funcionais		
ID NF	Descrição	Categoria
1.1	Capacidade do requisito funcional de recuperar dados após uma falha do sistema ou queda de energia.	Confiabilidade
1.2	O requisito funcional deve estar disponível 24h por dia para o usuário.	Desempenho
1.3	O voucher será gerado automaticamente após a confirmação do pagamento.	Usabilidade
1.4	Os botões presentes na funcionalidade: Download e Imprimir.	Usabilidade
1.5	A função imprimir contará com o atalho Ctrl + P.	Usabilidade
1.6	A função deverá possuir as cores seguindo o mesmo padrão do sistema.	Interface
1.7	O site contará com acessibilidade para pessoas com deficiência visual.	Usabilidade
1.8	O site contará com acessibilidade para pessoas com deficiência auditiva.	Usabilidade
1.9	O tamanho da tela deve se adaptar aos diferentes tamanhos de telas dos dispositivos eletrônicos, sejam eles com uso mobile ou desktop.	Portabilidade
1.10	É necessário o uso do sistema operacional Android ou IOS para executar o requisito funcional em sistemas mobile;	Portabilidade
1.11	O botão deve mudar de cor quando o usuário passar o mouse em cima.	Interface
1.12	As informações que o voucher deve ter: número do voucher, código do voucher, validade, comprador, data de compra, valor pago, cupons compostos, regulamento da oferta e também a localização do estabelecimento de consumo.	Especificação

Fonte: Produzido pelo autor (2024).

Quadro 28 - Validar Voucher.

Requisito Funcional		
Nome:	Validar Voucher	Código: F25
Descrição:	Responsável por validar voucher para os parceiros possuírem o controle de vouchers.	
Estimativa de Esforço: 10h	Prioridade: Alta	
Requisitos Não funcionais		
ID NF	Descrição	Categoria
1.1	Capacidade do requisito funcional de recuperar dados após uma falha do sistema ou queda de energia.	Confiabilidade

1.2	O requisito funcional deve estar disponível 24h por dia para o usuário.	Desempenho
1.3	A função deverá apresentar o botão: Validar Voucher.	Usabilidade
1.4	O usuário do sistema terá que completar o código do voucher e também colocar o CPF do responsável pela compra, para segurança do voucher.	Segurança
1.5	Os dois campos requeridos neste requisito serão obrigatórios, caso o cliente não digite, irá aparecer uma mensagem dizendo: “Por favor, digite todos os campos”.	Usabilidade
1.6	O “Enter” deverá ser o atalho para o botão “Validar Voucher”.	Usabilidade
1.7	O campo onde o usuário vai digitar contará com um placeholder para facilitar o seu entendimento do dado necessário a ser informado.	Usabilidade
1.8	A confirmação da validação de um voucher deve ser em segundos.	Desempenho
1.9	A função deverá possuir as cores seguindo o mesmo padrão do sistema.	Interface
1.10	O site contará com acessibilidade para pessoas com deficiência visual.	Usabilidade
1.11	O site contará com acessibilidade para pessoas com deficiência auditiva.	Usabilidade
1.12	O tamanho da tela deve se adaptar aos diferentes tamanhos de telas dos dispositivos eletrônicos, sejam eles com uso mobile ou desktop.	Portabilidade
1.13	É necessário o uso do sistema operacional Android ou IOS para executar o requisito funcional em sistemas mobile;	Portabilidade
1.14	O botão deve mudar de cor quando o usuário passar o mouse em cima.	Interface
1.15	As informações que o voucher deve ter: número do voucher, código do voucher, validade, comprador, data de compra, valor pago, produtos compostos, regulamento da oferta e também a localização do estabelecimento de consumo.	Especificação
1.16	Os dados inseridos pelo usuário no requisito funcional devem estar criptografados.	Segurança

Fonte: Produzido pelo autor (2024).

Quadro 29 - Estornar Venda.

Requisito Funcional		
Nome:	Estornar Venda	Código: F26
Descrição:	Responsável por estornar a venda.	
Estimativa de Esforço: 6h	Prioridade: Alta	
Requisitos Não funcionais		
ID NF	Descrição	Categoria
1.1	Capacidade do requisito funcional de recuperar dados após uma falha do sistema ou queda de energia.	Confiabilidade
1.2	O requisito funcional deve estar disponível 24h por dia para o usuário.	Desempenho
1.3	O único botão presente na tela será o “Estornar”, tendo cores que se distinguem do seu fundo para o destacá-lo.	Interface
1.4	O campo onde o usuário vai digitar contará com um placeholder para facilitar o seu entendimento do dado necessário a ser informado.	Usabilidade
1.5	A função deverá possuir as cores seguindo o mesmo padrão do sistema.	Interface
1.6	O site contará com acessibilidade para pessoas com deficiência visual.	Usabilidade

1.7	O site contará com acessibilidade para pessoas com deficiência auditiva.	Usabilidade
1.8	O tamanho da tela deve se adaptar aos diferentes tamanhos de telas dos dispositivos eletrônicos, sejam eles com uso mobile ou desktop.	Portabilidade
1.9	É necessário o uso do sistema operacional Android ou IOS para executar o requisito funcional em sistemas mobile;	Portabilidade
1.10	O botão deve mudar de cor quando o usuário passar o mouse em cima.	Interface
1.11	Os dados inseridos pelo usuário no requisito funcional devem estar criptografados.	Segurança
1.12	O “Enter” deverá ser o atalho para o botão Pedir Estornar.	Usabilidade
1.13	Garantir que os dados registrados pelo sistema não serão duplicados.	Confiabilidade
1.14	As informações do parceiro que deverão ser armazenadas pela função no banco de dados são: motivo, venda.	Especificação

Fonte: Produzido pelo autor (2024).

Quadro 30 - Consultar Favoritos.

Requisito Funcional		
Nome:	Consultar Favoritos	Código: F27
Descrição:	Consultar dados dos favoritos armazenados no sistema.	
Estimativa de Esforço: 6h	Prioridade: Média	
Requisitos Não funcionais		
ID NF	Descrição	Categoria
1.1	Capacidade do requisito funcional de recuperar dados após uma falha do sistema ou queda de energia.	Confiabilidade
1.2	O requisito funcional deve estar disponível 24h por dia para o usuário.	Desempenho
1.3	O “Enter” deverá ser o atalho para a pesquisa de um favorito específico.	Usabilidade
1.4	O campo onde o usuário vai digitar contará com um placeholder para facilitar o seu entendimento do dado necessário a ser informado.	Usabilidade
1.5	A função deverá possuir as cores seguindo o mesmo padrão do sistema.	Interface
1.6	O site contará com acessibilidade para pessoas com deficiência visual.	Usabilidade
1.7	O site contará com acessibilidade para pessoas com deficiência auditiva.	Usabilidade
1.8	O tamanho da tela deve se adaptar aos diferentes tamanhos de telas dos dispositivos eletrônicos, sejam eles com uso mobile ou desktop.	Portabilidade
1.9	É necessário o uso do sistema operacional Android ou IOS para executar o requisito funcional em sistemas mobile;	Portabilidade
1.10	O botão deve mudar de cor quando o usuário passar o mouse em cima.	Interface
1.11	Os dados inseridos pelo usuário no requisito funcional devem estar criptografados.	Segurança
1.12	Garantir que os dados registrados pelo sistema não serão duplicados.	Confiabilidade
1.13	Os resultados da consulta serão mostrados em GRID: nome, parceiro, produto, ações.	Usabilidade
1.14	A GRID apresentará uma barra de rolagem vertical e horizontal para permitir a visualização de todas as linhas e colunas da grade.	Interface

1.15	O sistema deverá ter uma atualização rápida dos favoritos excluídos na GRID.	Desempenho
1.16	O sistema apresentará uma tela de confirmação antes de excluir o registro.	segurança
1.17	As consultas serão efetuadas pelos campos: nome (Título Produto favoritado);	Usabilidade

Fonte: Produzido pelo autor (2024).

Quadro 31 - Consultar Categorias.

Requisito Funcional		
Nome:	Consultar Categorias	Código: F28
Descrição:	Consultar dados das categorias armazenadas no sistema.	
Estimativa de Esforço: 6h	Prioridade: Média	
Requisitos Não funcionais		
ID NF	Descrição	Categoria
1.1	Capacidade do requisito funcional de recuperar dados após uma falha do sistema ou queda de energia.	Confiabilidade
1.2	O requisito funcional deve estar disponível 24h por dia para o usuário.	Desempenho
1.3	O “Enter” deverá ser o atalho para a funcionalidade pesquisa de uma categoria.	Usabilidade
1.4	O campo onde o usuário vai digitar contará com um placeholder para facilitar o seu entendimento do dado necessário a ser informado.	Usabilidade
1.5	A função deverá possuir as cores seguindo o mesmo padrão do sistema.	Interface
1.6	O site contará com acessibilidade para pessoas com deficiência visual.	Usabilidade
1.7	O site contará com acessibilidade para pessoas com deficiência auditiva.	Usabilidade
1.8	O tamanho da tela deve se adaptar aos diferentes tamanhos de telas dos dispositivos eletrônicos, sejam eles com uso mobile ou desktop.	Portabilidade
1.9	É necessário o uso do sistema operacional Android ou IOS para executar o requisito funcional em sistemas mobile;	Portabilidade
1.10	O botão deve mudar de cor quando o usuário passar o mouse em cima.	Interface
1.11	Os dados inseridos pelo usuário no requisito funcional devem estar criptografados.	Segurança
1.12	O sistema não poderá constar na sua consulta dados duplicados.	Confiabilidade
1.13	O campo de busca terá um ícone de lupa o mesmo com um atalho e um placeholder escrito “Pesquisar: ”.	Interface
1.14	Os resultados da consulta serão mostrados em GRID: Nome da categoria, descrição e status.	Usabilidade
1.15	Os botões presentes na GRID são: edição e excluir, ambos representados por um ícone de um lápis e outro por uma lixeira, respectivamente.	Interface
1.16	A GRID apresentará uma barra de rolagem vertical e horizontal para permitir a visualização de todas as linhas e colunas da grade.	Interface
1.17	O sistema apresentará uma tela de confirmação antes de excluir o registro.	segurança
1.18	O botão (ícone) de editar abrirá um fluxo alternativo onde poderá ser modificado os dados cadastrados.	Usabilidade

1.19	O sistema deverá ter uma atualização rápida das categorias excluídas na GRID.	Desempenho
1.20	As consultas serão efetuadas pelos campos: nome.	Usabilidade

Fonte: Produzido pelo autor (2024).

Quadro 32 - Consultar Clientes.

Requisito Funcional		
Nome:	Consultar Clientes	Código: F29
Descrição:	Consultar dados dos clientes armazenados no sistema.	
Estimativa de Esforço: 6h	Prioridade: Média	
Requisitos Não funcionais		
ID NF	Descrição	Categoria
1.1	Capacidade do requisito funcional de recuperar dados após uma falha do sistema ou queda de energia.	Confiabilidade
1.2	O requisito funcional deve estar disponível 24h por dia para o usuário.	Desempenho
1.3	O “Enter” deverá ser o atalho para a pesquisa de um cliente específico.	Usabilidade
1.4	O campo onde o usuário vai digitar contará com um placeholder para facilitar o seu entendimento do dado necessário a ser informado.	Usabilidade
1.5	A função deverá possuir as cores seguindo o mesmo padrão do sistema.	Interface
1.6	O site contará com acessibilidade para pessoas com deficiência visual.	Usabilidade
1.7	O site contará com acessibilidade para pessoas com deficiência auditiva.	Usabilidade
1.8	O tamanho da tela deve se adaptar aos diferentes tamanhos de telas dos dispositivos eletrônicos, sejam eles com uso mobile ou desktop.	Portabilidade
1.9	É necessário o uso do sistema operacional Android ou IOS para executar o requisito funcional em sistemas mobile;	Portabilidade
1.10	O botão deve mudar de cor quando o usuário passar o mouse em cima.	Interface
1.11	Os dados inseridos pelo usuário no requisito funcional devem estar criptografados.	Segurança
1.12	O sistema não poderá constar na sua consulta dados duplicados.	Confiabilidade
1.13	O campo de busca terá um ícone de lupa o mesmo com um atalho e um placeholder escrito “Pesquisar: ”.	Interface
1.14	Os resultados da consulta serão mostrados em GRID: nome, CPF, email, telefone, status e ações.	Usabilidade
1.15	Os botões presentes na GRID são: edição e excluir, ambos representados por um ícone de um lápis e outro por uma lixeira, respectivamente.	Interface
1.16	A GRID apresentará uma barra de rolagem vertical e horizontal para permitir a visualização de todas as linhas e colunas da grade.	Interface
1.17	O sistema apresentará uma tela de confirmação antes de excluir o registro.	segurança
1.18	O botão (ícone) de editar abrirá um fluxo alternativo onde poderá ser modificado os dados cadastrados.	Usabilidade
1.19	O sistema deverá ter uma atualização rápida dos clientes excluídos na GRID.	Desempenho

1.20	As consultas serão efetuadas pelos campos: cpf e nome.	Usabilidade
------	--	-------------

Fonte: Produzido pelo autor (2024).

Quadro 33 - Consultar Parceiros.

Requisito Funcional		
Nome:	Consultar Parceiros	Código: F30
Descrição:	Consultar dados dos parceiros armazenados no sistema.	
Estimativa de Esforço: 6h	Prioridade: Média	
Requisitos Não funcionais		
ID NF	Descrição	Categoria
1.1	Capacidade do requisito funcional de recuperar dados após uma falha do sistema ou queda de energia.	Confiabilidade
1.2	O requisito funcional deve estar disponível 24h por dia para o usuário.	Desempenho
1.3	O “Enter” deverá ser o atalho para a funcionalidade pesquisa de uma parceiro.	Usabilidade
1.4	O campo onde o usuário vai digitar contará com um placeholder para facilitar o seu entendimento do dado necessário a ser informado.	Usabilidade
1.5	A função deverá possuir as cores seguindo o mesmo padrão do sistema.	Interface
1.6	O site contará com acessibilidade para pessoas com deficiência visual.	Usabilidade
1.7	O site contará com acessibilidade para pessoas com deficiência auditiva.	Usabilidade
1.8	O tamanho da tela deve se adaptar aos diferentes tamanhos de telas dos dispositivos eletrônicos, sejam eles com uso mobile ou desktop.	Portabilidade
1.9	É necessário o uso do sistema operacional Android ou IOS para executar o requisito funcional em sistemas mobile;	Portabilidade
1.10	O botão deve mudar de cor quando o usuário passar o mouse em cima.	Interface
1.11	Os dados inseridos pelo usuário no requisito funcional devem estar criptografados.	Segurança
1.12	O sistema não poderá constar na sua consulta dados duplicados.	Confiabilidade
1.13	O campo de busca terá um ícone de lupa o mesmo com um atalho e um placeholder escrito “Pesquisar: ”.	Interface
1.14	Os resultados da consulta serão mostrados em GRID: Nome Fantasia, CNPJ, telefone e ações.	Usabilidade
1.15	Os botões presentes na GRID são: edição e excluir, ambos representados por um ícone de um lápis e outro por uma lixeira, respectivamente.	Interface
1.16	A GRID apresentará uma barra de rolagem vertical e horizontal para permitir a visualização de todas as linhas e colunas da grade.	Interface
1.17	O sistema apresentará uma tela de confirmação antes de excluir o registro.	segurança
1.18	O botão (ícone) de editar abrirá um fluxo alternativo onde poderá ser modificado os dados cadastrados.	Usabilidade
1.19	O sistema deverá ter uma atualização rápida das parceiro excluídas na GRID.	Desempenho
1.20	As consultas serão efetuadas pelos campos: cnpj e nome fantasia	Usabilidade

Fonte: Produzido pelo autor (2024).

Quadro 34 - Consultar Cupons.

Requisito Funcional		
Nome:	Consultar Cupons	Código: F31
Descrição:	Consultar dados dos cupons armazenados no sistema.	
Estimativa de Esforço: 6h	Prioridade: Média	
Requisitos Não funcionais		
ID NF	Descrição	Categoria
1.1	Capacidade do requisito funcional de recuperar dados após uma falha do sistema ou queda de energia.	Confiabilidade
1.2	O requisito funcional deve estar disponível 24h por dia para o usuário.	Desempenho
1.3	O “Enter” deverá ser o atalho para a funcionalidade pesquisa de um cupom.	Usabilidade
1.4	O campo onde o usuário vai digitar contará com um placeholder para facilitar o seu entendimento do dado necessário a ser informado.	Usabilidade
1.5	A função deverá possuir as cores seguindo o mesmo padrão do sistema.	Interface
1.6	O site contará com acessibilidade para pessoas com deficiência visual.	Usabilidade
1.7	O site contará com acessibilidade para pessoas com deficiência auditiva.	Usabilidade
1.8	O tamanho da tela deve se adaptar aos diferentes tamanhos de telas dos dispositivos eletrônicos, sejam eles com uso mobile ou desktop.	Portabilidade
1.9	É necessário o uso do sistema operacional Android ou IOS para executar o requisito funcional em sistemas mobile;	Portabilidade
1.10	O botão deve mudar de cor quando o usuário passar o mouse em cima.	Interface
1.11	Os dados inseridos pelo usuário no requisito funcional devem estar criptografados.	Segurança
1.12	O sistema não poderá constar na sua consulta dados duplicados.	Confiabilidade
1.13	O campo de busca terá um ícone de lupa o mesmo com um atalho e um placeholder escrito “Pesquisar: ”.	Interface
1.14	Os resultados da consulta serão mostrados em GRID: nome, parceiro, categoria, status e ação.	Usabilidade
1.15	Os botões presentes na GRID são: visualizar, representado por um ícone de olho.	Interface
1.16	A GRID apresentará uma barra de rolagem vertical e horizontal para permitir a visualização de todas as linhas e colunas da grade.	Interface
1.17	O sistema apresentará uma tela de confirmação antes de excluir o registro.	segurança
1.18	O botão (ícone) de visualizar abrirá um fluxo alternativo onde poderá visualizar os dados cadastrados.	Usabilidade
1.19	O sistema deverá ter uma atualização rápida das cupom excluídas na GRID.	Desempenho
1.20	As consultas serão efetuadas pelos campos: nome do cupom.	Usabilidade

Fonte: Produzido pelo autor (2024).

Quadro 35 - Consultar Agendas

Requisito Funcional

Nome:	Consultar Agendas	Código:	F32
Descrição:	Consultar dados das categorias armazenadas no sistema.		
Estimativa de Esforço: 6h	Prioridade: Média		
Requisitos Não funcionais			
ID NF	Descrição	Categoria	
1.1	Capacidade do requisito funcional de recuperar dados após uma falha do sistema ou queda de energia.	Confiabilidade	
1.2	O requisito funcional deve estar disponível 24h por dia para o usuário.	Desempenho	
1.3	O “Enter” deverá ser o atalho para a funcionalidade de pesquisa de uma agenda.	Usabilidade	
1.4	O campo onde o usuário vai digitar contará com um placeholder para facilitar o seu entendimento do dado necessário a ser informado.	Usabilidade	
1.5	A função deverá possuir as cores seguindo o mesmo padrão do sistema.	Interface	
1.6	O site contará com acessibilidade para pessoas com deficiência visual.	Usabilidade	
1.7	O site contará com acessibilidade para pessoas com deficiência auditiva.	Usabilidade	
1.8	O tamanho da tela deve se adaptar aos diferentes tamanhos de telas dos dispositivos eletrônicos, sejam eles com uso mobile ou desktop.	Portabilidade	
1.9	É necessário o uso do sistema operacional Android ou IOS para executar o requisito funcional em sistemas mobile;	Portabilidade	
1.10	O botão deve mudar de cor quando o usuário passar o mouse em cima.	Interface	
1.11	Os dados inseridos pelo usuário no requisito funcional devem estar criptografados.	Segurança	
1.12	O sistema não poderá constar na sua consulta dados duplicados.	Confiabilidade	
1.13	O campo de busca terá um ícone de lupa o mesmo com um atalho e um placeholder escrito “Pesquisar: ”.	Interface	
1.14	Os resultados da consulta serão mostrados em GRID: Nome, parceiro, data e hora, status e ações.	Usabilidade	
1.15	Os botões presentes na GRID são: edição e excluir, ambos representados por um ícone de um lápis e outro por uma lixeira, respectivamente.	Interface	
1.16	A GRID apresentará uma barra de rolagem vertical e horizontal para permitir a visualização de todas as linhas e colunas da grade.	Interface	
1.17	O sistema apresentará uma tela de confirmação antes de excluir o registro.	segurança	
1.18	O botão (ícone) de editar abrirá um fluxo alternativo onde poderá ser modificado os dados cadastrados.	Usabilidade	
1.19	O sistema deverá ter uma atualização rápida das agendas excluídas na GRID.	Desempenho	
1.20	As consultas serão efetuadas pelos campos: data da agenda.	Usabilidade	

Fonte: Produzido pelo autor (2024).

Quadro 36 - Consultar Vendas.

Requisito Funcional			
Nome:	Consultar Vendas	Código:	F33
Descrição:	Consultar vendas realizadas pelo sistema.		

Estimativa de Esforço: 6h		Prioridade: Média
Requisitos Não funcionais		
ID NF	Descrição	Categoria
1.1	Capacidade do requisito funcional de recuperar dados após uma falha do sistema ou queda de energia.	Confiabilidade
1.2	O requisito funcional deve estar disponível 24h por dia para o usuário.	Desempenho
1.3	O “Enter” deverá ser o atalho para a funcionalidade pesquisa de vendas.	Usabilidade
1.4	O campo onde o usuário vai digitar contará com um placeholder para facilitar o seu entendimento do dado necessário a ser informado.	Usabilidade
1.5	A função deverá possuir as cores seguindo o mesmo padrão do sistema.	Interface
1.6	O site contará com acessibilidade para pessoas com deficiência visual.	Usabilidade
1.7	O site contará com acessibilidade para pessoas com deficiência auditiva.	Usabilidade
1.8	O tamanho da tela deve se adaptar aos diferentes tamanhos de telas dos dispositivos eletrônicos, sejam eles com uso mobile ou desktop.	Portabilidade
1.9	É necessário o uso do sistema operacional Android ou IOS para executar o requisito funcional em sistemas mobile;	Portabilidade
1.10	O botão deve mudar de cor quando o usuário passar o mouse em cima.	Interface
1.11	Os dados inseridos pelo usuário no requisito funcional devem estar criptografados.	Segurança
1.12	O sistema não poderá constar na sua consulta dados duplicados.	Confiabilidade
1.13	O campo de busca terá um ícone de lupa o mesmo com um atalho e um placeholder escrito “Pesquisar: ”.	Interface
1.14	Os resultados da consulta serão mostrados em GRID: código, cliente, parceiro, status e ação.	Usabilidade
1.15	O botão presentes na GRID é: visualizar, representado por um ícone de olho.	Interface
1.16	A GRID apresentará uma barra de rolagem vertical e horizontal para permitir a visualização de todas as linhas e colunas da grade.	Interface
1.17	O sistema apresentará uma tela de confirmação antes de excluir o registro.	segurança
1.18	O botão (ícone) de visualizar abrirá um fluxo alternativo onde poderá visualizar os dados cadastrados.	Usabilidade
1.19	O sistema deverá ter uma atualização rápida das vendas excluídas na GRID.	Desempenho
1.20	As consultas serão efetuadas pelos campos: código da venda.	Usabilidade

Fonte: Produzido pelo autor (2024).

Quadro 37 - Consultar Recebimentos.

Requisito Funcional		
Nome:	Consultar recebimentos	Código: F34
Descrição:	Consultar dados dos recebimentos armazenados no sistema.	
Estimativa de Esforço: 6h	Prioridade: Média	
Requisitos Não funcionais		
ID NF	Descrição	Categoria

1.1	Capacidade do requisito funcional de recuperar dados após uma falha do sistema ou queda de energia.	Confiabilidade
1.2	O requisito funcional deve estar disponível 24h por dia para o usuário.	Desempenho
1.3	O “Enter” deverá ser o atalho para a pesquisa de um recebimento específico.	Usabilidade
1.4	O campo onde o usuário vai digitar contará com um placeholder para facilitar o seu entendimento do dado necessário a ser informado.	Usabilidade
1.5	A função deverá possuir as cores seguindo o mesmo padrão do sistema.	Interface
1.6	O site contará com acessibilidade para pessoas com deficiência visual.	Usabilidade
1.7	O site contará com acessibilidade para pessoas com deficiência auditiva.	Usabilidade
1.8	O tamanho da tela deve se adaptar aos diferentes tamanhos de telas dos dispositivos eletrônicos, sejam eles com uso mobile ou desktop.	Portabilidade
1.9	É necessário o uso do sistema operacional Android ou IOS para executar o requisito funcional em sistemas mobile;	Portabilidade
1.10	O botão deve mudar de cor quando o usuário passar o mouse em cima.	Interface
1.11	Os dados inseridos pelo usuário no requisito funcional devem estar criptografados.	Segurança
1.12	O sistema não poderá constar na sua consulta dados duplicados.	Confiabilidade
1.13	O campo de busca terá um ícone de lupa o mesmo com um atalho e um placeholder escrito “Pesquisar: ”.	Interface
1.14	Os resultados da consulta serão mostrados em GRID: código, cliente, valor, forma de pagamentos, status e ações.	Usabilidade
1.15	O botão presentes na GRID é: visualizar, representado por um ícone de olho.	Interface
1.16	A GRID apresentará uma barra de rolagem vertical e horizontal para permitir a visualização de todas as linhas e colunas da grade.	Interface
1.17	O sistema apresentará uma tela de confirmação antes de excluir o registro.	segurança
1.18	O botão (ícone) de visualizar abrirá um fluxo alternativo onde poderá visualizar os dados cadastrados.	Usabilidade
1.19	O sistema deverá ter uma atualização rápida dos recebimentos excluídos na GRID.	Desempenho
1.20	As consultas serão efetuadas pelos campos: código do recebimento.	Usabilidade

Fonte: Produzido pelo autor (2024).

Quadro 38 - Consultar Vouchers.

Requisito Funcional		
Nome:	Consultar Vouchers	Código: F35
Descrição:	Consultar dados dos vouchers armazenados no sistema.	
Estimativa de Esforço: 6h	Prioridade: Média	
Requisitos Não funcionais		
ID NF	Descrição	Categoria
1.1	Capacidade do requisito funcional de recuperar dados após uma falha do sistema ou queda de energia.	Confiabilidade
1.2	O requisito funcional deve estar disponível 24h por dia para o usuário.	Desempenho

1.3	O “Enter” deverá ser o atalho para a funcionalidade pesquisa de um voucher.	Usabilidade
1.4	O campo onde o usuário vai digitar contará com um placeholder para facilitar o seu entendimento do dado necessário a ser informado.	Usabilidade
1.5	A função deverá possuir as cores seguindo o mesmo padrão do sistema.	Interface
1.6	O site contará com acessibilidade para pessoas com deficiência visual.	Usabilidade
1.7	O site contará com acessibilidade para pessoas com deficiência auditiva.	Usabilidade
1.8	O tamanho da tela deve se adaptar aos diferentes tamanhos de telas dos dispositivos eletrônicos, sejam eles com uso mobile ou desktop.	Portabilidade
1.9	É necessário o uso do sistema operacional Android ou IOS para executar o requisito funcional em sistemas mobile;	Portabilidade
1.10	O botão deve mudar de cor quando o usuário passar o mouse em cima.	Interface
1.11	Os dados inseridos pelo usuário no requisito funcional devem estar criptografados.	Segurança
1.12	O sistema não poderá constar na sua consulta dados duplicados.	Confiabilidade
1.13	O campo de busca terá um ícone de lupa o mesmo com um atalho e um placeholder escrito “Pesquisar: ”.	Interface
1.14	Os resultados da consulta serão mostrados em GRID: código, cliente, data de expiração, status e ação.	Usabilidade
1.15	O botão presentes na GRID é: visualizar, representado por um ícone de olho.	Interface
1.16	A GRID apresentará uma barra de rolagem vertical e horizontal para permitir a visualização de todas as linhas e colunas da grade.	Interface
1.17	O sistema apresentará uma tela de confirmação antes de excluir o registro.	segurança
1.18	O botão (ícone) de visualizar abrirá um fluxo alternativo onde poderá visualizar os dados cadastrados.	Usabilidade
1.19	O sistema deverá ter uma atualização rápida das categorias excluídas na GRID.	Desempenho
1.20	As consultas serão efetuadas pelos campos: número do voucher.	Usabilidade

Fonte: Produzido pelo autor (2024).

Quadro 39 - Consultar Solicitação de Cupons.

Requisito Funcional		
Nome:	Consultar solicitação de cupons	Código: F36
Descrição:	Consultar dados das solicitações de cupons armazenados no sistema.	
Estimativa de Esforço: 6h	Prioridade: Média	
Requisitos Não funcionais		
ID NF	Descrição	Categoria
1.1	Capacidade do requisito funcional de recuperar dados após uma falha do sistema ou queda de energia.	Confiabilidade e
1.2	O requisito funcional deve estar disponível 24h por dia para o usuário.	Desempenho
1.3	O “Enter” deverá ser o atalho para a funcionalidade pesquisa de uma solicitação de cupons.	Usabilidade
1.4	O campo onde o usuário vai digitar contará com um placeholder para facilitar o seu entendimento do dado necessário a ser informado.	Usabilidade

1.5	A função deverá possuir as cores seguindo o mesmo padrão do sistema.	Interface
1.6	O site contará com acessibilidade para pessoas com deficiência visual.	Usabilidade
1.7	O site contará com acessibilidade para pessoas com deficiência auditiva.	Usabilidade
1.8	O tamanho da tela deve se adaptar aos diferentes tamanhos de telas dos dispositivos eletrônicos, sejam eles com uso mobile ou desktop.	Portabilidade
1.9	É necessário o uso do sistema operacional Android ou IOS para executar o requisito funcional em sistemas mobile;	Portabilidade
1.10	O botão deve mudar de cor quando o usuário passar o mouse em cima.	Interface
1.11	Os dados inseridos pelo usuário no requisito funcional devem estar criptografados.	Segurança
1.12	O sistema não poderá constar na sua consulta dados duplicados.	Confiabilidade e
1.13	O campo de busca terá um ícone de lupa o mesmo com um atalho e um placeholder escrito “Pesquisar: ”.	Interface
1.14	Os resultados da consulta serão mostrados em GRID: nome do cupom, parceiro, motivo, status e ação.	Usabilidade
1.15	O botão presentes na GRID é: visualizar, representado por um ícone de olho.	Interface
1.16	A GRID apresentará uma barra de rolagem vertical e horizontal para permitir a visualização de todas as linhas e colunas da grade.	Interface
1.17	O sistema apresentará uma tela de confirmação antes de excluir o registro.	segurança
1.18	O botão (ícone) de visualizar abrirá um fluxo alternativo onde poderá visualizar os dados cadastrados.	Usabilidade
1.19	O sistema deverá ter uma atualização rápida das solicitações de cupons excluídos na GRID.	Desempenho
1.20	As consultas serão efetuadas pelos campos: nome ou número da solicitação.	Usabilidade

Fonte: Produzido pelo autor (2024).

Quadro 40 - Consultar Solicitação de Parceiros.

Requisito Funcional		
Nome:	Consultar solicitação de parceiros	Código: F37
Descrição:	Consultar dados das solicitações de parceiros armazenados no sistema.	
Estimativa de Esforço: 6h	Prioridade: Média	
Requisitos Não funcionais		
ID NF	Descrição	Categoria
1.1	Capacidade do requisito funcional de recuperar dados após uma falha do sistema ou queda de energia.	Confiabilidade
1.2	O requisito funcional deve estar disponível 24h por dia para o usuário.	Desempenho
1.3	O “Enter” deverá ser o atalho para a funcionalidade pesquisa de uma solicitação de parceiros.	Usabilidade
1.4	O campo onde o usuário vai digitar contará com um placeholder para facilitar o seu entendimento do dado necessário a ser informado.	Usabilidade
1.5	A função deverá possuir as cores seguindo o mesmo padrão do sistema.	Interface

1.6	O site contará com acessibilidade para pessoas com deficiência visual.	Usabilidade
1.7	O site contará com acessibilidade para pessoas com deficiência auditiva.	Usabilidade
1.8	O tamanho da tela deve se adaptar aos diferentes tamanhos de telas dos dispositivos eletrônicos, sejam eles com uso mobile ou desktop.	Portabilidade
1.9	É necessário o uso do sistema operacional Android ou IOS para executar o requisito funcional em sistemas mobile;	Portabilidade
1.10	O botão deve mudar de cor quando o usuário passar o mouse em cima.	Interface
1.11	Os dados inseridos pelo usuário no requisito funcional devem estar criptografados.	Segurança
1.12	O sistema não poderá constar na sua consulta dados duplicados.	Confiabilidade
1.13	O campo de busca terá um ícone de lupa o mesmo com um atalho e um placeholder escrito “Pesquisar: ”.	Interface
1.14	Os resultados da consulta serão mostrados em GRID: nome fantasia, CNPJ, telefone, status e ação.	Usabilidade
1.15	Os botões presentes na GRID são: edição e excluir, ambos representados por um ícone de um lápis e outro por uma lixeira, respectivamente.	Interface
1.16	A GRID apresentará uma barra de rolagem vertical e horizontal para permitir a visualização de todas as linhas e colunas da grade.	Interface
1.17	O sistema apresentará uma tela de confirmação antes de excluir o registro.	segurança
1.18	O botão (ícone) de editar abrirá um fluxo alternativo onde poderá ser modificado os dados cadastrados.	Usabilidade
1.19	O sistema deverá ter uma atualização rápida das solicitações de cupons excluídos na GRID.	Desempenho
1.20	As consultas serão efetuadas pelos campos: nome fantasia ou CNPJ.	Usabilidade

Fonte: Produzido pelo autor (2024).

Quadro 41 -Autorizar Solicitação de Parceiros.

Requisito Funcional		
Nome:	Autorizar solicitação de parceiros	Código: F38
Descrição:	Autorizar solicitações de parceiros armazenados no sistema.	
Estimativa de Esforço: 6h	Prioridade: Média	
Requisitos Não funcionais		
ID NF	Descrição	Categoria
1.1	Capacidade do requisito funcional de recuperar dados após uma falha do sistema ou queda de energia.	Confiabilidade
1.2	O requisito funcional deve estar disponível 24h por dia para o usuário.	Desempenho
1.3	A função deverá possuir as cores seguindo o mesmo padrão do sistema.	Interface
1.4	O site contará com acessibilidade para pessoas com deficiência visual.	Usabilidade
1.5	O site contará com acessibilidade para pessoas com deficiência auditiva.	Usabilidade
1.6	O tamanho da tela deve se adaptar aos diferentes tamanhos de telas dos dispositivos eletrônicos, sejam eles com uso mobile ou desktop.	Portabilidade

1.7	É necessário o uso do sistema operacional Android ou IOS para executar o requisito funcional em sistemas mobile;	Portabilidade
1.8	O botão deve mudar de cor quando o usuário passar o mouse em cima.	Interface
1.9	Os dados inseridos pelo usuário no requisito funcional devem estar criptografados.	Segurança
1.10	O sistema não poderá constar na sua consulta dados duplicados.	Confiabilidade
1.11	Os resultados da consulta serão mostrados em formatos de cartões: nome fantasia, razão social, CNPJ, email, telefone, endereço e ação.	Usabilidade
1.12	Os botões presentes na cartões são: aceitar e cancelar, ambos representados por um ícone de um “ok” e outro por de “cancelar”, respectivamente.	Interface
1.13	O sistema apresentará uma tela de confirmação antes de excluir o registro.	segurança
1.14	O botão (ícone) de “ok” abrirá um fluxo alternativo onde poderá ser modificado os dados de autorização da solicitação do parceiro.	Usabilidade
1.15	O sistema deverá ter uma atualização rápida das autorizações de solicitação de parceiros excluídos nos cartões.	Desempenho

Fonte: Produzido pelo autor (2024).

Quadro 42 - Autorizar Solicitação de Cupons.

Requisito Funcional		
Nome:	Autorizar solicitação de cupons	Código: F39
Descrição:	Autorizar solicitações de cupons armazenados no sistema.	
Estimativa de Esforço: 6h	Prioridade: Média	
Requisitos Não funcionais		
ID NF	Descrição	Categoria
1.1	Capacidade do requisito funcional de recuperar dados após uma falha do sistema ou queda de energia.	Confiabilidade
1.2	O requisito funcional deve estar disponível 24h por dia para o usuário.	Desempenho
1.3	A função deverá possuir as cores seguindo o mesmo padrão do sistema.	Interface
1.4	O site contará com acessibilidade para pessoas com deficiência visual.	Usabilidade
1.5	O site contará com acessibilidade para pessoas com deficiência auditiva.	Usabilidade
1.6	O tamanho da tela deve se adaptar aos diferentes tamanhos de telas dos dispositivos eletrônicos, sejam eles com uso mobile ou desktop.	Portabilidade
1.7	É necessário o uso do sistema operacional Android ou IOS para executar o requisito funcional em sistemas mobile;	Portabilidade
1.8	O botão deve mudar de cor quando o usuário passar o mouse em cima.	Interface
1.9	Os dados inseridos pelo usuário no requisito funcional devem estar criptografados.	Segurança
1.10	O sistema não poderá constar na sua consulta dados duplicados.	Confiabilidade
1.11	Os resultados da consulta serão mostrados em formatos de cartões: título do cupom, descrição, preço, porcentagem de desconto e valor do produto com desconto, nome do parceiro e ação.	Usabilidade
1.12	Os botões presentes na cartões são: aceitar e cancelar, ambos representados por um ícone de um “ok” e outro por de “cancelar”, respectivamente.	Interface

1.13	O sistema apresentará uma tela de confirmação antes de excluir o registro.	segurança
1.14	O botão (ícone) de “ok” abrirá um fluxo alternativo onde poderá ser modificado os dados de autorização da solicitação do parceiro.	Usabilidade
1.15	O sistema deverá ter uma atualização rápida das autorizações de solicitação de cupons excluídos nos cartões.	Desempenho

Fonte: Produzido pelo autor (2024).

Quadro 43 - Emitir Relatório de Vendas.

Requisito Funcional		
Nome:	Emitir relatório de vendas	Código: F40
Descrição:	Gerar um relatório das vendas registradas no sistema.	
Estimativa de Esforço: 6h	Prioridade: Média	
Requisitos Não funcionais		
ID NF	Descrição	Categoria
1.1	Capacidade do requisito funcional de recuperar dados após uma falha do sistema ou queda de energia.	Confiabilidade
1.2	O requisito funcional deve estar disponível 24h por dia para o usuário.	Desempenho
1.3	O botão presente na funcionalidade é o emitir.	Usabilidade
1.4	O “Enter” deverá ser o atalho para o botão emitir.	Usabilidade
1.5	O campo onde o usuário vai digitar contará com um placeholder para facilitar o seu entendimento do dado necessário a ser informado.	Usabilidade
1.6	A função deverá possuir as cores seguindo o mesmo padrão do sistema.	Interface
1.7	O site contará com acessibilidade para pessoas com deficiência visual.	Usabilidade
1.8	O site contará com acessibilidade para pessoas com deficiência auditiva.	Usabilidade.3
1.9	O tamanho da tela deve se adaptar aos diferentes tamanhos de telas dos dispositivos eletrônicos, sejam eles com uso mobile ou desktop.	Portabilidade
1.10	É necessário o uso do sistema operacional Android ou IOS para executar o requisito funcional em sistemas mobile;	Portabilidade
1.11	O botão deve mudar de cor quando o usuário passar o mouse em cima.	Interface
1.12	Os dados inseridos pelo usuário no requisito funcional devem estar criptografados.	Segurança
1.13	O sistema não poderá constar na sua consulta dados duplicados.	Confiabilidade
1.14	Os campos de categoria, parceiro, cliente, cupom, status e forma de pagamento contaram com um ícone de um combobox, tendo a funcionalidade de um atalho para dados presentes no sistema.	Usabilidade
1.15	O campo de inserção de período contará com um ícone de calendário ao lado que funcionará como um atalho para preencher o período desejado.	Interface
1.16	O relatório de venda abrirá em uma nova guia. Tendo a opção de baixar ou imprimir.	Usabilidade
1.17	A função imprimir contará com o atalho Ctrl + P.	Usabilidade
1.18	Caso o usuário não tenha preenchido todos os campos obrigatórios, não poderá realizar a consulta.	Confiabilidade

Fonte: Produzido pelo autor (2024).

Quadro 44 - Emitir Relatório de Vouchers.

Requisito Funcional		
Nome:	Emitir relatório de vouchers	Código: F41
Descrição:	Gerar um relatório das vouchers registradas no sistema.	
Estimativa de Esforço: 6h	Prioridade: Média	
Requisitos Não funcionais		
ID NF	Descrição	Categoria
1.1	Capacidade do requisito funcional de recuperar dados após uma falha do sistema ou queda de energia.	Confiabilidade
1.2	O requisito funcional deve estar disponível 24h por dia para o usuário.	Desempenho
1.3	O botão presente na funcionalidade é o emitir.	Usabilidade
1.4	O “Enter” deverá ser o atalho para o botão emitir.	Usabilidade
1.5	O campo onde o usuário vai digitar contará com um placeholder para facilitar o seu entendimento do dado necessário a ser informado.	Usabilidade
1.6	A função deverá possuir as cores seguindo o mesmo padrão do sistema.	Interface
1.7	O site contará com acessibilidade para pessoas com deficiência visual.	Usabilidade
1.8	O site contará com acessibilidade para pessoas com deficiência auditiva.	Usabilidade.3
1.9	O tamanho da tela deve se adaptar aos diferentes tamanhos de telas dos dispositivos eletrônicos, sejam eles com uso mobile ou desktop.	Portabilidade
1.10	É necessário o uso do sistema operacional Android ou IOS para executar o requisito funcional em sistemas mobile;	Portabilidade
1.11	O botão deve mudar de cor quando o usuário passar o mouse em cima.	Interface
1.12	Os dados inseridos pelo usuário no requisito funcional devem estar criptografados.	Segurança
1.13	O sistema não poderá constar na sua consulta dados duplicados.	Confiabilidade
1.14	Os campos de categoria, parceiro, cliente, cupom, status e forma de pagamento contarão com um ícone de um combobox, tendo a funcionalidade de um atalho para dados presentes no sistema.	Usabilidade
1.15	O campo de inserção de período contará com um ícone de calendário ao lado que funcionará como um atalho para preencher o período desejado.	Interface
1.16	O relatório de venda abrirá em uma nova guia. Tendo a opção de baixar ou imprimir.	Usabilidade
1.17	A função imprimir contará com o atalho Ctrl + P.	Usabilidade
1.18	Caso o usuário não tenha preenchido todos os campos obrigatórios, não poderá realizar a consulta.	Confiabilidade

Fonte: Produzido pelo autor (2024).

Quadro 45 - Emitir Relatório de Categorias.

Requisito Funcional		
Nome:	Emitir relatório de categorias	Código: F42
Descrição:	Gerar um relatório das categorias registradas no sistema.	
Estimativa de Esforço: 6h	Prioridade: Média	

Requisitos Não funcionais		
ID NF	Descrição	Categoria
1.1	Capacidade do requisito funcional de recuperar dados após uma falha do sistema ou queda de energia.	Confiabilidade
1.2	O requisito funcional deve estar disponível 24h por dia para o usuário.	Desempenho
1.3	O botão presente na funcionalidade é o emitir.	Usabilidade
1.4	O “Enter” deverá ser o atalho para o botão emitir.	Usabilidade
1.5	O campo onde o usuário vai digitar contará com um placeholder para facilitar o seu entendimento do dado necessário a ser informado.	Usabilidade
1.6	A função deverá possuir as cores seguindo o mesmo padrão do sistema.	Interface
1.7	O site contará com acessibilidade para pessoas com deficiência visual.	Usabilidade
1.8	O site contará com acessibilidade para pessoas com deficiência auditiva.	Usabilidade.3
1.9	O tamanho da tela deve se adaptar aos diferentes tamanhos de telas dos dispositivos eletrônicos, sejam eles com uso mobile ou desktop.	Portabilidade
1.10	É necessário o uso do sistema operacional Android ou IOS para executar o requisito funcional em sistemas mobile;	Portabilidade
1.11	O botão deve mudar de cor quando o usuário passar o mouse em cima.	Interface
1.12	Os dados inseridos pelo usuário no requisito funcional devem estar criptografados.	Segurança
1.13	O sistema não poderá constar na sua consulta dados duplicados.	Confiabilidade
1.14	Os campos de categoria e status, contaram com um ícone de um combobox, tendo a funcionalidade de um atalho para dados presentes no sistema.	Usabilidade
1.15	O campo de inserção de período contará com um ícone de calendário ao lado que funcionará como um atalho para preencher o período desejado.	Interface
1.16	O relatório de venda abrirá em uma nova guia. Tendo a opção de baixar ou imprimir.	Usabilidade
1.17	A função imprimir contará com o atalho Ctrl + P.	Usabilidade
1.18	Caso o usuário não tenha preenchido todos os campos obrigatórios, não poderá realizar a consulta.	Confiabilidade

Fonte: Produzido pelo autor (2024).

Quadro 46 - Emitir Relatório de Clientes.

Requisito Funcional		
Nome:	Emitir Relatório de Clientes	Código: F43
Descrição:	Por meio do sistema, o usuário terá a possibilidade de gerar um relatório dos clientes cadastrados.	
Estimativa de Esforço: 6h	Prioridade: Média	
Requisitos Não funcionais		
ID NF	Descrição	Categoria
1.1	Capacidade do requisito funcional de recuperar dados após uma falha do sistema ou queda de energia.	Confiabilidade
1.2	O requisito funcional deve estar disponível 24h por dia para o usuário.	Desempenho
1.3	O botão presente na funcionalidade é o emitir.	Usabilidade
1.4	O “Enter” deverá ser o atalho para o botão emitir.	Usabilidade

1.5	O campo onde o usuário vai digitar contará com um placeholder para facilitar o seu entendimento do dado necessário a ser informado.	Usabilidade
1.6	A função deverá possuir as cores seguindo o mesmo padrão do sistema.	Interface
1.7	O site contará com acessibilidade para pessoas com deficiência visual.	Usabilidade
1.8	O site contará com acessibilidade para pessoas com deficiência auditiva.	Usabilidade.3
1.9	O tamanho da tela deve se adaptar aos diferentes tamanhos de telas dos dispositivos eletrônicos, sejam eles com uso mobile ou desktop.	Portabilidade
1.10	É necessário o uso do sistema operacional Android ou IOS para executar o requisito funcional em sistemas mobile;	Portabilidade
1.11	O botão deve mudar de cor quando o usuário passar o mouse em cima.	Interface
1.12	Os dados inseridos pelo usuário no requisito funcional devem estar criptografados.	Segurança
1.13	O sistema não poderá constar na sua consulta dados duplicados.	Confiabilidade
1.14	Os campos de cnpj, razão social e status, contaram com um ícone de um combobox, tendo a funcionalidade de um atalho para dados presentes no sistema.	Usabilidade
1.15	O campo de inserção de período contará com um ícone de calendário ao lado que funcionará como um atalho para preencher o período desejado.	Interface
1.16	O relatório de venda abrirá em uma nova guia. Tendo a opção de baixar ou imprimir.	Usabilidade
1.17	A função imprimir contará com o atalho Ctrl + P.	Usabilidade
1.18	Caso o usuário não tenha preenchido todos os campos obrigatórios, não poderá realizar a consulta.	Confiabilidade

Fonte: Produzido pelo autor (2024).

Quadro 47 - Emitir Relatório de Parceiros.

Requisito Funcional		
Nome:	Emitir Relatório de Parceiros	Código: F44
Descrição:	Por meio do sistema, o usuário terá a possibilidade de gerar um relatório dos parceiros cadastrados.	
Estimativa de Esforço: 6h	Prioridade: Média	
Requisitos Não funcionais		
ID NF	Descrição	Categoria
1.1	Capacidade do requisito funcional de recuperar dados após uma falha do sistema ou queda de energia.	Confiabilidade
1.2	O requisito funcional deve estar disponível 24h por dia para o usuário.	Desempenho
1.3	O botão presente na funcionalidade é o emitir.	Usabilidade
1.4	O "Enter" deverá ser o atalho para o botão emitir.	Usabilidade
1.5	O campo onde o usuário vai digitar contará com um placeholder para facilitar o seu entendimento do dado necessário a ser informado.	Usabilidade
1.6	A função deverá possuir as cores seguindo o mesmo padrão do sistema.	Interface
1.7	O site contará com acessibilidade para pessoas com deficiência visual.	Usabilidade

1.8	O site contará com acessibilidade para pessoas com deficiência auditiva.	Usabilidade.3
1.9	O tamanho da tela deve se adaptar aos diferentes tamanhos de telas dos dispositivos eletrônicos, sejam eles com uso mobile ou desktop.	Portabilidade
1.10	É necessário o uso do sistema operacional Android ou IOS para executar o requisito funcional em sistemas mobile;	Portabilidade
1.11	O botão deve mudar de cor quando o usuário passar o mouse em cima.	Interface
1.12	Os dados inseridos pelo usuário no requisito funcional devem estar criptografados.	Segurança
1.13	O sistema não poderá constar na sua consulta dados duplicados.	Confiabilidade
1.14	Os campos de cnpj, razão social e status, contaram com um ícone de um combobox, tendo a funcionalidade de um atalho para dados presentes no sistema.	Usabilidade
1.15	O campo de inserção de período contará com um ícone de calendário ao lado que funcionará como um atalho para preencher o período desejado.	Interface
1.16	O relatório de venda abrirá em uma nova guia. Tendo a opção de baixar ou imprimir.	Usabilidade
1.17	A função imprimir contará com o atalho Ctrl + P.	Usabilidade
1.18	Caso o usuário não tenha preenchido todos os campos obrigatórios, não poderá realizar a consulta.	Confiabilidade

Fonte: Produzido pelo autor (2024).

Quadro 48 - Emitir Relatório de Cupons.

Requisito Funcional		
Nome:	Emitir Relatório de Cupons	Código: F45
Descrição:	Por meio do sistema, o usuário terá a possibilidade de gerar um relatório dos cupons cadastrados.	
Estimativa de Esforço: 6h	Prioridade: Média	
Requisitos Não funcionais		
ID NF	Descrição	Categoria
1.1	Capacidade do requisito funcional de recuperar dados após uma falha do sistema ou queda de energia.	Confiabilidade
1.2	O requisito funcional deve estar disponível 24h por dia para o usuário.	Desempenho
1.3	O botão presente na funcionalidade é o emitir.	Usabilidade
1.4	O “Enter” deverá ser o atalho para o botão emitir.	Usabilidade
1.5	O campo onde o usuário vai digitar contará com um placeholder para facilitar o seu entendimento do dado necessário a ser informado.	Usabilidade
1.6	A função deverá possuir as cores seguindo o mesmo padrão do sistema.	Interface
1.7	O site contará com acessibilidade para pessoas com deficiência visual.	Usabilidade
1.8	O site contará com acessibilidade para pessoas com deficiência auditiva.	Usabilidade.3
1.9	O tamanho da tela deve se adaptar aos diferentes tamanhos de telas dos dispositivos eletrônicos, sejam eles com uso mobile ou desktop.	Portabilidade
1.10	É necessário o uso do sistema operacional Android ou IOS para executar o requisito funcional em sistemas mobile;	Portabilidade

1.11	O botão deve mudar de cor quando o usuário passar o mouse em cima.	Interface
1.12	Os dados inseridos pelo usuário no requisito funcional devem estar criptografados.	Segurança
1.13	O sistema não poderá constar na sua consulta dados duplicados.	Confiabilidade
1.14	Os campos de categoria, status e parceiro, contaram com um ícone de um combobox, tendo a funcionalidade de um atalho para dados presentes no sistema.	Usabilidade
1.15	O campo de inserção de período contará com um ícone de calendário ao lado que funcionará como um atalho para preencher o período desejado.	Interface
1.16	O relatório de venda abrirá em uma nova guia. Tendo a opção de baixar ou imprimir.	Usabilidade
1.17	A função imprimir contará com o atalho Ctrl + P.	Usabilidade
1.18	Caso o usuário não tenha preenchido todos os campos obrigatórios, não poderá realizar a consulta.	Confiabilidade

Fonte: Produzido pelo autor (2024).

Quadro 49 - Emitir Relatório de Agendas.

Requisito Funcional		
Nome:	Emitir Relatório de Agendas	Código: F46
Descrição:	Por meio do sistema, o usuário terá a possibilidade de gerar um relatório dos cupons cadastrados.	
Estimativa de Esforço: 6h	Prioridade: Média	
Requisitos Não funcionais		
ID NF	Descrição	Categoria
1.1	Capacidade do requisito funcional de recuperar dados após uma falha do sistema ou queda de energia.	Confiabilidade
1.2	O requisito funcional deve estar disponível 24h por dia para o usuário.	Desempenho
1.3	O botão presente na funcionalidade é o emitir.	Usabilidade
1.4	O “Enter” deverá ser o atalho para o botão emitir.	Usabilidade
1.5	O campo onde o usuário vai digitar contará com um placeholder para facilitar o seu entendimento do dado necessário a ser informado.	Usabilidade
1.6	A função deverá possuir as cores seguindo o mesmo padrão do sistema.	Interface
1.7	O site contará com acessibilidade para pessoas com deficiência visual.	Usabilidade
1.8	O site contará com acessibilidade para pessoas com deficiência auditiva.	Usabilidade.3
1.9	O tamanho da tela deve se adaptar aos diferentes tamanhos de telas dos dispositivos eletrônicos, sejam eles com uso mobile ou desktop.	Portabilidade
1.10	É necessário o uso do sistema operacional Android ou IOS para executar o requisito funcional em sistemas mobile;	Portabilidade
1.11	O botão deve mudar de cor quando o usuário passar o mouse em cima.	Interface
1.12	Os dados inseridos pelo usuário no requisito funcional devem estar criptografados.	Segurança
1.13	O sistema não poderá constar na sua consulta dados duplicados.	Confiabilidade

1.14	Os campos de categoria, status e parceiro, contaram com um ícone de um combobox, tendo a funcionalidade de um atalho para dados presentes no sistema.	Usabilidade
1.15	O campo de inserção de período contará com um ícone de calendário ao lado que funcionará como um atalho para preencher o período desejado.	Interface
1.16	O relatório de venda abrirá em uma nova guia. Tendo a opção de baixar ou imprimir.	Usabilidade
1.17	A função imprimir contará com o atalho Ctrl + P.	Usabilidade
1.18	Caso o usuário não tenha preenchido todos os campos obrigatórios, não poderá realizar a consulta.	Confiabilidade

Fonte: Produzido pelo autor (2024).

Quadro 50 - Emitir Relatório de Favoritos.

Requisito Funcional		
Nome:	Emitir Relatório de Favoritos	Código: F47
Descrição:	Gerar um relatório dos favoritos registrados no sistema.	
Estimativa de Esforço: 6h	Prioridade: Média	
Requisitos Não funcionais		
ID NF	Descrição	Categoria
1.1	Capacidade do requisito funcional de recuperar dados após uma falha do sistema ou queda de energia.	Confiabilidade
1.2	O requisito funcional deve estar disponível 24h por dia para o usuário.	Desempenho
1.3	O botão presente na funcionalidade é o emitir.	Usabilidade
1.4	O “Enter” deverá ser o atalho para o botão emitir.	Usabilidade
1.5	O campo onde o usuário vai digitar contará com um placeholder para facilitar o seu entendimento do dado necessário a ser informado.	Usabilidade
1.6	A função deverá possuir as cores seguindo o mesmo padrão do sistema.	Interface
1.7	O site contará com acessibilidade para pessoas com deficiência visual.	Usabilidade
1.8	O site contará com acessibilidade para pessoas com deficiência auditiva.	Usabilidade.3
1.9	O tamanho da tela deve se adaptar aos diferentes tamanhos de telas dos dispositivos eletrônicos, sejam eles com uso mobile ou desktop.	Portabilidade
1.10	É necessário o uso do sistema operacional Android ou IOS para executar o requisito funcional em sistemas mobile;	Portabilidade
1.11	O botão deve mudar de cor quando o usuário passar o mouse em cima.	Interface
1.12	Os dados inseridos pelo usuário no requisito funcional devem estar criptografados.	Segurança
1.13	O sistema não poderá constar na sua consulta dados duplicados.	Confiabilidade
1.14	Os campos de clientes, cupom e parceiro, contaram com um ícone de um combobox, tendo a funcionalidade de um atalho para dados presentes no sistema.	Usabilidade
1.15	O campo de inserção de período contará com um ícone de calendário ao lado que funcionará como um atalho para preencher o período desejado.	Interface
1.16	O relatório de venda abrirá em uma nova guia. Tendo a opção de baixar ou imprimir.	Usabilidade

1.17	A função imprimir contará com o atalho Ctrl + P.	Usabilidade
1.18	Caso o usuário não tenha preenchido todos os campos obrigatórios, não poderá realizar a consulta.	Confiabilidade

Fonte: Produzido pelo autor (2024).

Quadro 51 - Emitir Relatório de Recebimentos.

Requisito Funcional		
Nome:	Emitir Relatório de Recebimentos	Código: F48
Descrição:	Gerar um relatório dos recebimentos registrados no sistema.	
Estimativa de Esforço: 6h	Prioridade: Média	
Requisitos Não funcionais		
ID NF	Descrição	Categoria
1.1	Capacidade do requisito funcional de recuperar dados após uma falha do sistema ou queda de energia.	Confiabilidade
1.2	O requisito funcional deve estar disponível 24h por dia para o usuário.	Desempenho
1.3	O botão presente na funcionalidade é o emitir.	Usabilidade
1.4	O “Enter” deverá ser o atalho para o botão emitir.	Usabilidade
1.5	O campo onde o usuário vai digitar contará com um placeholder para facilitar o seu entendimento do dado necessário a ser informado.	Usabilidade
1.6	A função deverá possuir as cores seguindo o mesmo padrão do sistema.	Interface
1.7	O site contará com acessibilidade para pessoas com deficiência visual.	Usabilidade
1.8	O site contará com acessibilidade para pessoas com deficiência auditiva.	Usabilidade.3
1.9	O tamanho da tela deve se adaptar aos diferentes tamanhos de telas dos dispositivos eletrônicos, sejam eles com uso mobile ou desktop.	Portabilidade
1.10	É necessário o uso do sistema operacional Android ou IOS para executar o requisito funcional em sistemas mobile;	Portabilidade
1.11	O botão deve mudar de cor quando o usuário passar o mouse em cima.	Interface
1.12	Os dados inseridos pelo usuário no requisito funcional devem estar criptografados.	Segurança
1.13	O sistema não poderá constar na sua consulta dados duplicados.	Confiabilidade
1.14	Os campos de clientes, formas de pagamento, parceiro e status, contaram com um ícone de um combobox, tendo a funcionalidade de um atalho para dados presentes no sistema.	Usabilidade
1.15	O campo de inserção de período contará com um ícone de calendário ao lado que funcionará como um atalho para preencher o período desejado.	Interface
1.16	O relatório de venda abrirá em uma nova guia. Tendo a opção de baixar ou imprimir.	Usabilidade
1.17	A função imprimir contará com o atalho Ctrl + P.	Usabilidade
1.18	Caso o usuário não tenha preenchido todos os campos obrigatórios, não poderá realizar a consulta.	Confiabilidade

Fonte: Produzido pelo autor (2024).

Quadro 52 - Gerar Dashboard de Parceiros.

Requisito Funcional		
Nome:	Gerar Dashboard de Parceiro	Código: F49
Descrição:	O dashboard informa para o administrador todas as informações presente no sistema de forma simples e dinâmica.	
Estimativa de Esforço: 12h	Prioridade: Alta	
Requisitos Não funcionais		
ID NF	Descrição	Categoria
1.1	Capacidade do requisito funcional de recuperar dados após uma falha do sistema ou queda de energia.	Confiabilidade
1.2	O requisito funcional deve estar disponível 24h por dia para o usuário.	Desempenho
1.3	A função deverá possuir as cores seguindo o mesmo padrão do sistema.	Interface
1.4	O site contará com acessibilidade para pessoas com deficiência visual.	Usabilidade
1.5	O site contará com acessibilidade para pessoas com deficiência auditiva.	Usabilidade.3
1.6	O tamanho da tela deve se adaptar aos diferentes tamanhos de telas dos dispositivos eletrônicos, sejam eles com uso mobile ou desktop.	Portabilidade
1.7	É necessário o uso do sistema operacional Android ou IOS para executar o requisito funcional em sistemas mobile;	Portabilidade
1.8	O botão deve mudar de cor quando o usuário passar o mouse em cima.	Interface
1.9	Os dados aparecem no dashboard são dados confidenciais do sistema.	Segurança
1.10	O dashboard não poderá constar na sua consulta dados duplicados.	Confiabilidade
1.11	As informações presentes na funcionalidade precisam de atualizações constantes e imediatas.	Confiabilidade
1.12	No lado esquerdo o dashboard contará com um menu contendo todos os requisitos presentes no sistema.	Interface
1.13	O gráfico contendo as informações do requisito escolhido pelo usuário apareceram no centro da tela.	Interface
1.14	O dashboard terá sua divisão de informações em meses.	Interface
1.15	O gráfico contará com uma síntese das informações presente nele ao seu lado direito. Além de cartões informando os dados principais presentes no dashboard .	Interface

Fonte: Produzido pelo autor (2024).

Quadro 53 - Gerar Dashboard de Administrador.

Requisito Funcional		
Nome:	Gerar Dashboard de Administrador	Código: F50
Descrição:	O dashboard informa para o administrador todas as informações presente no sistema de forma simples e dinâmica.	
Estimativa de Esforço: 12h	Prioridade: Alta	
Requisitos Não funcionais		
ID NF	Descrição	Categoria
1.1	Capacidade do requisito funcional de recuperar dados após uma falha do sistema ou queda de energia.	Confiabilidade

1.2	O requisito funcional deve estar disponível 24h por dia para o usuário.	Desempenho
1.3	A função deverá possuir as cores seguindo o mesmo padrão do sistema.	Interface
1.4	O site contará com acessibilidade para pessoas com deficiência visual.	Usabilidade
1.5	O site contará com acessibilidade para pessoas com deficiência auditiva.	Usabilidade.3
1.6	O tamanho da tela deve se adaptar aos diferentes tamanhos de telas dos dispositivos eletrônicos, sejam eles com uso mobile ou desktop.	Portabilidade
1.7	É necessário o uso do sistema operacional Android ou IOS para executar o requisito funcional em sistemas mobile;	Portabilidade
1.8	O botão deve mudar de cor quando o usuário passar o mouse em cima.	Interface
1.9	Os dados que aparecem no dashboard são dados confidenciais do sistema.	Segurança
1.10	O dashboard não poderá constar na sua consulta dados duplicados.	Confiabilidade
1.11	As informações presentes na funcionalidade precisam de atualizações constantes e imediatas.	Confiabilidade
1.12	No lado esquerdo o dashboard contará com um menu contendo todos os requisitos presentes no sistema.	Interface
1.13	O gráfico contendo as informações do requisito escolhido pelo usuário apareceram no centro da tela.	Interface
1.14	O dashboard terá sua divisão de informações em meses.	Interface
1.15	O gráfico contará com uma síntese das informações presente nele ao seu lado direito. Além de cartões informando os dados principais presentes no dashboard .	Interface

Fonte: Produzido pelo autor (2024).

Quadro 54 - Agendar.

Requisito Funcional		
Nome:	Agendar	Código: F51
Descrição:	Será responsável de agendar data e horário para determinado cupom.	
Estimativa de Esforço: 10h	Prioridade: Média	
Requisitos Não funcionais		
ID NF	Descrição	Categoria
1.1	Capacidade do requisito funcional de recuperar dados após uma falha do sistema ou queda de energia.	Confiabilidade
1.2	O requisito funcional deve estar disponível 24h por dia para o usuário.	Desempenho
1.3	A função deverá apresentar o botão: EXEMPLO.	Usabilidade
1.4	O usuário do sistema terá que completar o código do voucher e também colocar o CPF do responsável pela compra, para segurança do voucher.	Segurança
1.5	O campo onde o usuário vai digitar contará com um placeholder para facilitar o seu entendimento do dado necessário a ser informado.	Usabilidade
1.6	A função deverá possuir as cores seguindo o mesmo padrão do sistema.	Interface
1.7	O site contará com acessibilidade para pessoas com deficiência visual.	Usabilidade
1.8	O site contará com acessibilidade para pessoas com deficiência auditiva.	Usabilidade

1.9	O tamanho da tela deve se adaptar aos diferentes tamanhos de telas dos dispositivos eletrônicos, sejam eles com uso mobile ou desktop.	Portabilidade
1.10	É necessário o uso do sistema operacional Android ou IOS para executar o requisito funcional em sistemas mobile;	Portabilidade
1.11	O botão deve mudar de cor quando o usuário passar o mouse em cima.	Interface
1.12	Os dados inseridos pelo usuário no requisito funcional devem estar criptografados.	Segurança
1.13	Garantir que os dados registrados pelo sistema não serão duplicados.	Confiabilidade

Fonte: Produzido pelo autor (2024).

3 PROTÓTIPO DO SISTEMA

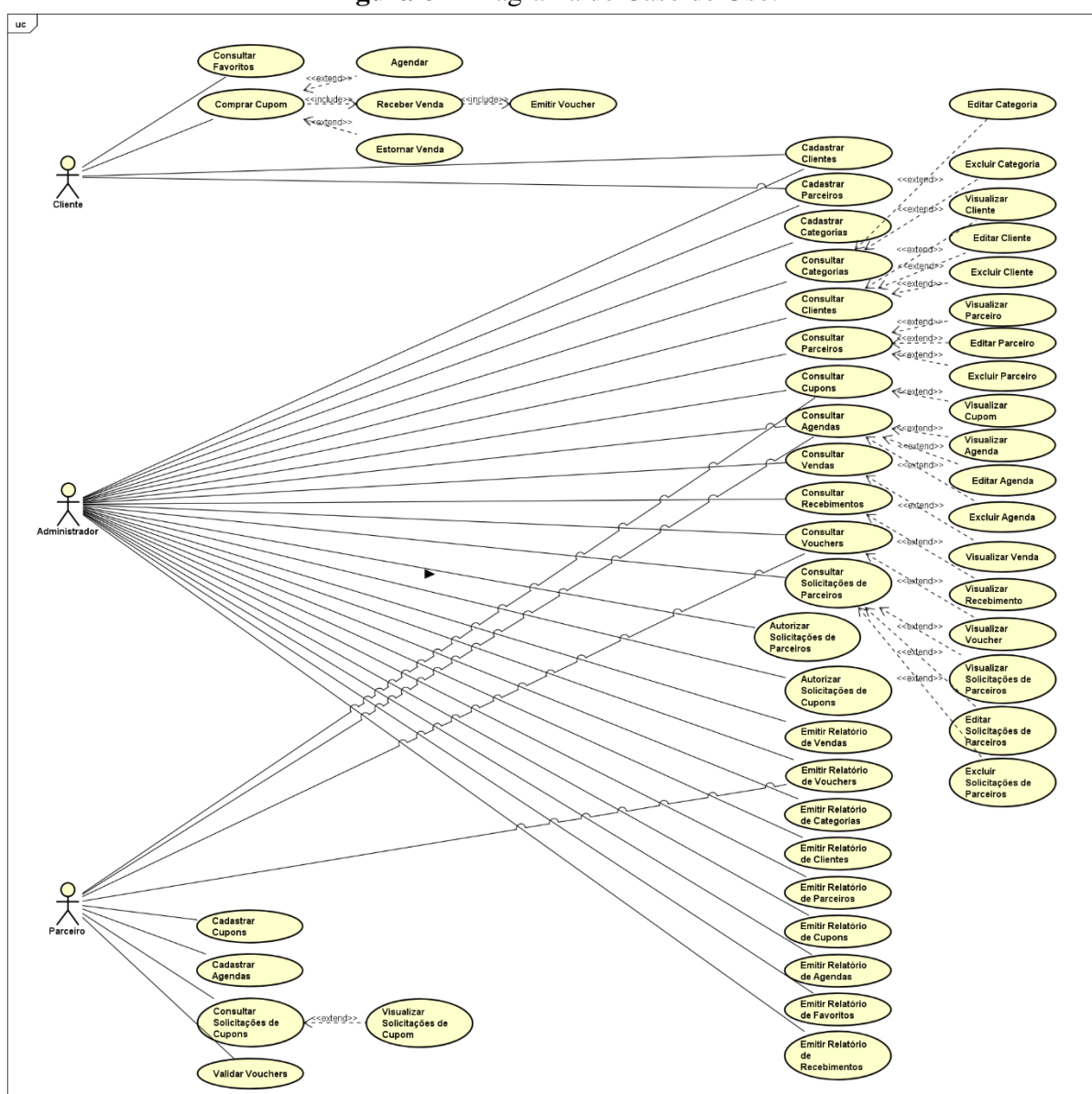
Durante a fase de projeto foram elaborados protótipos de cada requisito funcional para validação do requisito junto ao cliente. O protótipo foi elaborado com a ferramenta Figma e está disponível para acesso no link: [Link para o Protótipo do Sistema](#). Você pode visualizar a versão do sistema sob diferentes perspectivas: como cliente [Fluxo Cliente](#), como parceiro [Fluxo Parceiro](#) e como administrador [Fluxo Administrador](#).

4 CASO DE USO

O diagrama de caso de uso tem como objeto demonstrar os perfis de acesso ao sistema e quais funcionalidades os mesmos podem acessar.

4.1 Diagrama de Caso de Uso

Figura 6 – Diagrama de Caso de Uso.



Fonte: Produzido pelo autor (2024).

4.2 Caso de Uso Expandido

4.2.1 Caso de Uso – Acessar Login Cliente

Caso de Uso: Acessar Login Cliente (CSU01).

Descrição: O cliente deve efetuar o login no sistema.

Ator Primário: Cliente.

Fluxo Principal

1. O usuário ao acessar o site, clicar no ícone "usuário" referente a efetuar login.
2. O sistema abre a funcionalidade com todos os campos em branco e com o botão entrar, esqueci minha senha, cadastre-se.
3. O sistema solicita dados obrigatórios, sendo eles usuário e senha.
4. O usuário clica no botão Entrar.
5. Caso o usuário preencha alguma das informações incorretas, o sistema deverá alertar o usuário.
6. Caso as informações preenchidas estejam corretas o usuário será autenticado no sistema.

4.2.2 Caso de Uso – Acessar Login Parceiro

Caso de Uso: Acessar Login Parceiro (CSU02).

Descrição: O parceiro deve efetuar o login no sistema.

Ator Primário: Parceiro.

Fluxo Principal

1. O usuário ao acessar o site, clicar no ícone "usuário" referente a efetuar login.
2. O sistema abre a funcionalidade com todos os campos em branco e com o botão entrar, esqueci minha senha, cadastre-se.
3. O sistema solicita dados obrigatórios, sendo eles usuário e senha.
4. O usuário clica no botão Entrar.
5. Caso o usuário preencha alguma das informações incorretas, o sistema deverá alertar o usuário.
6. Caso as informações preenchidas estejam corretas o usuário será autenticado no sistema.

4.2.3 Caso de Uso – Acessar Login Administrador

Caso de Uso: Acessar Login Administrador (CSU03).

Descrição: O administrador deve efetuar o login no sistema.

Ator Primário: Administrador.

Fluxo Principal

1. O usuário ao acessar o site, clicar no ícone "usuário" referente a efetuar login.
2. O sistema abre a funcionalidade com todos os campos em branco e com o botão entrar, esqueci minha senha, cadastre-se.
3. O sistema solicita dados obrigatórios, sendo eles usuário e senha.
4. O usuário clica no botão Entrar.
5. Caso o usuário preencha alguma das informações incorretas, o sistema deverá alertar o usuário.
6. Caso as informações preenchidas estejam corretas o usuário será autenticado no sistema.

4.2.4 Caso de Uso – Redefinir Senha Cliente

Caso de Uso: Redefinir Senha Cliente (CSU04).

Descrição: O sistema deve enviar um e-mail para o cliente que deseja redefinir sua senha perdida ou esquecida.

Ator Primário: Cliente.

Fluxo Principal

1. O usuário ao acessar o site, clicar no ícone "usuário" referente a efetuar login.
2. O sistema abre a funcionalidade com todos os campos em branco e com o botão entrar, esqueci minha senha, cadastre-se.
3. O usuário clica em "esqueci minha senha".
4. O usuário preenche o campo e-mail.
5. O sistema envia um e-mail para o usuário com um link para alteração da senha.
6. O sistema informa que enviou o e-mail para o usuário válido.

4.2.5 Caso de Uso – Redefinir Senha Parceiro

Caso de Uso: Redefinir Senha Parceiro (CSU05).

Descrição: O sistema deve enviar um e-mail para o parceiro que deseja redefinir sua senha perdida ou esquecida.

Ator Primário: Parceiro.

Fluxo Principal

1. O usuário ao acessar o site, clicar no ícone "usuário" referente a efetuar login.
2. O sistema abre a funcionalidade com todos os campos em branco e com o botão entrar, esqueci minha senha, cadastre-se.
3. O usuário clica em "esqueci minha senha".
4. O usuário preenche o campo e-mail.
5. O sistema envia um e-mail para o usuário com um link para alteração da senha.
6. O sistema informa que enviou o e-mail para o usuário válido.

4.2.6 Caso de Uso – Redefinir Senha Administrador

Caso de Uso: Redefinir Senha Administrador (CSU06).

Descrição: O sistema deve enviar um e-mail para o administrador que deseja redefinir sua senha perdida ou esquecida.

Ator Primário: Administrador.

Fluxo Principal

1. O usuário ao acessar o site, clicar no ícone "usuário" referente a efetuar login.
2. O sistema abre a funcionalidade com todos os campos em branco e com o botão entrar, esqueci minha senha, cadastre-se.
3. O usuário clica em "esqueci minha senha".
4. O usuário preenche o campo e-mail.
5. O sistema envia um e-mail para o usuário com um link para alteração da senha.
6. O sistema informa que enviou o e-mail para o usuário válido.

4.2.7 Caso de Uso – Recuperar Senha Cliente

Caso de Uso: Recuperar Senha Cliente (CSU07).

Descrição: O sistema deve recuperar a senha do usuário.

Ator Primário: Cliente.

Pré-condições: O usuário deve acessar seu e-mail e receber um link do sistema.

Fluxo Principal

1. O usuário acessa o seu e-mail e clica no link.
2. O sistema abre a funcionalidade de recuperar senha, onde o usuário deve digitar a nova senha e a confirmação da nova senha.
3. O usuário digita os dois campos requisitados.
4. O sistema valida se os dois campos conferem.
5. O usuário clica em Salvar.
6. O sistema informa que a senha do usuário foi recuperada com sucesso.
7. O sistema direciona o usuário para a tela inicial.

4.2.8 Caso de Uso – Recuperar Senha Parceiro

Caso de Uso: Recuperar Senha Parceiro (CSU08).

Descrição: O sistema deve recuperar a senha do usuário.

Ator Primário: Parceiro.

Pré-condições: O usuário deve acessar seu e-mail e receber um link do sistema.

Fluxo Principal

1. O usuário acessa o seu e-mail e clica no link.
2. O sistema abre a funcionalidade de recuperar senha, onde o usuário deve digitar a nova senha e a confirmação da nova senha.
3. O usuário digita os dois campos requisitados.
4. O sistema valida se os dois campos conferem.
5. O usuário clica em Salvar.
6. O sistema informa que a senha do usuário foi recuperada com sucesso.
7. O sistema direciona o usuário para a tela inicial.

4.2.9 Caso de Uso – Recuperar Senha Administrador

Caso de Uso: Recuperar Senha Administrador (CSU09).

Descrição: O sistema deve recuperar a senha do usuário.

Ator Primário: Administrador.

Pré-condições: O usuário deve acessar seu e-mail e receber um link do sistema.

Fluxo Principal

1. O usuário acessa o seu e-mail e clica no link.
2. O sistema abre a funcionalidade de recuperar senha, onde o usuário deve digitar a nova senha e a confirmação da nova senha.
3. O usuário digita os dois campos requisitados.
4. O sistema valida se os dois campos conferem.
5. O usuário clica em Salvar.
6. O sistema informa que a senha do usuário foi recuperada com sucesso.
7. O sistema direciona o usuário para a tela inicial.

4.2.10 Caso de Uso – Alterar Senha Cliente

Caso de Uso: Alterar Senha Cliente (CSU10).

Descrição: O sistema deve realizar a alteração da senha do usuário.

Ator Primário: Cliente.

Pré-condições: O usuário deve estar logado no sistema, acessar o perfil e ir em alterar senha.

Fluxo Principal

1. O usuário acessa o seu perfil e vai em alterar senha.
2. O sistema abre a funcionalidade de alterar senha, onde o usuário deve digitar a senha atual, a nova senha e a confirmação da nova senha.
3. O usuário digita os três campos requisitados.
4. O sistema valida se os campos conferem.
5. O usuário clica em Salvar.
6. O sistema informa que a senha do usuário foi alterada com sucesso.
7. O sistema direciona o usuário para a tela inicial.

4.2.11 Caso de Uso – Alterar Senha Parceiro

Caso de Uso: Alterar Senha Parceiro (CSU11).

Descrição: O sistema deve realizar a alteração da senha do usuário.

Ator Primário: Parceiro.

Pré-condições: O usuário deve estar logado no sistema, acessar o perfil e ir em alterar senha.

Fluxo Principal

1. O usuário acessa o seu perfil e vai em alterar senha.
2. O sistema abre a funcionalidade de alterar senha, onde o usuário deve digitar a senha atual, a nova senha e a confirmação da nova senha.
3. O usuário digita os três campos requisitados.
4. O sistema valida se os campos conferem.
5. O usuário clica em Salvar.
6. O sistema informa que a senha do usuário foi alterada com sucesso.
7. O sistema direciona o usuário para a tela inicial.

4.2.12 Caso de Uso – Alterar Senha Administrador

Caso de Uso: Alterar Senha Administrador (CSU12).

Descrição: O sistema deve realizar a alteração da senha do usuário.

Ator Primário: Administrador.

Pré-condições: O usuário deve estar logado no sistema, acessar o perfil e ir em alterar senha.

Fluxo Principal

1. O usuário acessa o seu perfil e vai em alterar senha.
2. O sistema abre a funcionalidade de alterar senha, onde o usuário deve digitar a senha atual, a nova senha e a confirmação da nova senha.
3. O usuário digita os três campos requisitados.
4. O sistema valida se os campos conferem.
5. O usuário clica em Salvar.
6. O sistema informa que a senha do usuário foi alterada com sucesso.
7. O sistema direciona o usuário para a tela inicial.

4.2.13 Caso de Uso – Acessar Tela Inicial Cliente

Caso de Uso: Acessar Tela Inicial Cliente (CSU13).

Descrição: O usuário deve ter acesso a tela inicial com cupons e ofertas.

Ator Primário: Cliente.

Pré-condições: O usuário deve estar logado no sistema.

Fluxo Principal

1. O usuário acessa a tela inicial.

2. O sistema deve contar com carrossels, cards, com produtos disponíveis para compra no site.

4.2.14 Caso de Uso – Acessar Menu Inicial Parceiro

Caso de Uso: Acessar Menu Inicial Parceiro (CSU14).

Descrição: O usuário deve ter acesso ao menu inicial com opções para cadastros, consultas, dashboard.

Ator Primário: Parceiro.

Pré-condições: O usuário deve estar logado no sistema.

Fluxo Principal

1. O usuário acessa o menu inicial.
2. O sistema deve contar com um navbar com as opções para cadastros, consultas, dashboard.

4.2.15 Caso de Uso – Acessar Menu Inicial Administrador

Caso de Uso: Acessar Menu Inicial Parceiro (CSU15).

Descrição: O usuário deve ter acesso ao menu inicial com opções para cadastros, consultas, dashboard.

Ator Primário: Administrador.

Pré-condições: O usuário deve estar logado no sistema.

Fluxo Principal

1. O usuário acessa o menu inicial.
2. O sistema deve contar com um navbar com as opções para cadastros, consultas, dashboard.

4.2.16 Caso de Uso – Cadastrar Categorias

Caso de Uso: Cadastrar Categorias (CSU16).

Descrição: O usuário deve cadastrar uma categoria que não consta na base de dados.

Ator Primário: Administrador.

Pré-condições: O usuário deve estar logado no sistema.

Fluxo Principal

1. O usuário clica no botão “Categorias -> Cadastrar” disponível no Menu Principal.
2. O sistema abre a funcionalidade na aba dados principais com todos os campos em branco.
3. O sistema solicita dados obrigatórios e não obrigatórios do cliente, sendo eles categoria e descrição.
4. O usuário preenche os dados solicitados.
5. O usuário clica no botão cadastrar.
6. O sistema salva os dados no banco de dados e envia uma mensagem de confirmação com o texto “Cadastro realizado com sucesso!” na interface do usuário.

4.2.17 Caso de Uso – Cadastrar Clientes

Caso de Uso: Cadastrar Clientes (CSU17).

Descrição: O usuário deve cadastrar um cliente que não consta na base de dados.

Ator Primário: Cliente.

Fluxo Principal

1. O usuário acessa o sistema e clica no ícone referente a usuário.
2. O sistema abre a tela de login na aba dados principais com todos os campos em branco.
3. O usuário clica em "Cadastre-se".
4. O sistema abre a funcionalidade e solicita todos os campos obrigatórios e não obrigatórios.
5. O usuário preenche os dados solicitados.
6. O usuário clica no botão salvar.
7. O sistema salva os dados no banco de dados e envia uma mensagem de confirmação com o texto “Cadastro realizado com sucesso!” na interface do usuário.

4.2.18 Caso de Uso – Cadastrar Parceiros

Caso de Uso: Cadastrar Parceiros (CSU18).

Descrição: O usuário deve cadastrar um parceiro que não consta na base de dados.

Ator Primário: Parceiro.

Fluxo Principal

1. O usuário acessa o sistema e clica no "Seja um parceiro", disponível no rodapé.
2. O sistema abre a tela de login na aba dados principais com todos os campos em branco.
3. O sistema abre a funcionalidade e solicita todos os campos obrigatórios e não obrigatórios.
4. O usuário preenche os dados solicitados.
5. O usuário clica no botão salvar.
6. O sistema salva os dados no banco de dados e envia uma mensagem de confirmação com o texto “Solicitação de cadastro realizada com sucesso!” na interface do usuário.
7. O usuário acompanha por e-mail o cadastro.

4.2.19 Caso de Uso – Cadastrar Cupons

Caso de Uso: Cadastrar Cupons (CSU19).

Descrição: O usuário deve cadastrar um cupom que não consta na base de dados.

Ator Primário: Parceiro.

Pré-condições: O usuário deve estar logado no sistema.

Fluxo Principal

1. O usuário clica no botão “Cupons -> Cadastrar” disponível no Menu Principal.
2. O sistema abre a funcionalidade na aba dados principais com todos os campos em branco.
3. O sistema solicita dados obrigatórios e não obrigatórios.
4. O usuário preenche os dados solicitados.
5. O usuário clica no botão cadastrar.

6. O sistema salva os dados no banco de dados e envia uma mensagem de confirmação com o texto “Solicitação de cadastro realizada com sucesso!” na interface do usuário.

4.2.20 Caso de Uso – Cadastrar Agendas

Caso de Uso: Cadastrar Agendas (CSU20).

Descrição: O usuário deve cadastrar uma agenda que não consta na base de dados.

Ator Primário: Parceiro.

Pré-condições: O usuário deve estar logado no sistema.

Fluxo Principal

1. O usuário clica no botão “Agendas -> Cadastrar” disponível no Menu Principal.
2. O sistema abre a funcionalidade na aba dados principais com todos os campos em branco.
3. O sistema solicita dados obrigatórios e não obrigatórios.
4. O usuário preenche os dados solicitados.
5. O usuário clica no botão cadastrar.
6. O sistema salva os dados no banco de dados e envia uma mensagem de confirmação com o texto “Cadastro realizado com sucesso!” na interface do usuário.

4.2.21 Caso de Uso – Favoritar Cupons

Caso de Uso: Favoritar Cupons (CSU21).

Descrição: O usuário deve conseguir favoritar cupons para comprar depois.

Ator Primário: Cliente.

Pré-condições: O usuário deve estar logado no sistema.

Fluxo Principal

1. O usuário acessa a tela inicial onde terá acesso aos cupons.
2. O usuário deve clicar no coração disponível no card do produto.
3. Após o usuário clicar no coração, o coração deverá ficar vermelho, identificando que aquele cupom foi favoritado.

4.2.22 Caso de Uso – Comprar Cupons

Caso de Uso: Comprar Cupons (CSU22).

Descrição: O usuário deve comprar cupons.

Ator Primário: Cliente.

Pré-condições: O usuário deve estar logado no sistema.

Fluxo Principal

1. O usuário acessa a tela inicial onde terá acesso aos cupons.
2. O usuário deve clicar em cima da imagem do cupom ou no nome do cupom.
3. O sistema irá direcionar o usuário para a página do cupom, onde terá as informações do cupom, informações sobre o local de consumo, informações da oferta, entre outros.
4. O usuário deve clicar no botão "Comprar".
5. O sistema deve direcionar o cliente para o resumo da compra, onde poderá selecionar a quantidade.

4.2.23 Caso de Uso – Receber Vendas

Caso de Uso: Receber Vendas (CSU23).

Descrição: O usuário deve comprar cupons.

Ator Primário: Cliente.

Pré-condições: O usuário deve estar logado no sistema e finalizando o processo de compra de cupons.

Fluxo Principal

1. O usuário no processo final de compra de cupons deve selecionar a forma de pagamento dos cupons que está comprando "Pix ou Cartão de Crédito".

Fluxo Alternativo 1

1. Caso o usuário selecione Cartão de crédito.
2. O usuário deve clicar em "Continuar".

3. O sistema deve requerer todas as informações obrigatórias do cartão do usuário.
4. Após o usuário completar as informações, o mesmo deve clicar em continuar.
5. O sistema deve verificar o pagamento.
6. O sistema deve informar "Pagamento realizado com sucesso!".

Fluxo Alternativo 2

1. Caso o usuário selecione PIX.
2. O usuário deve clicar em "Continuar".
3. O sistema deve gerar um código QR e também o Pix copy cola, sendo assim o usuário irá selecionar a melhor forma de pagar.
4. Após o usuário realizar o pagamento.
5. O sistema deve verificar o pagamento
6. O sistema deve informar "Pagamento realizado com sucesso!".

4.2.24 Caso de Uso – Emitir Voucher

Caso de Uso: Comprar Cupons (CSU24).

Descrição: O sistema deve gerar o voucher do cliente.

Ator Primário: Cliente.

Pré-condições: O usuário deve finalizar o processo de compra de cupom.

Fluxo Principal

Após o processo de compra, o sistema deverá emitir e direcionar o usuário para o voucher.

O sistema deverá abrir uma nova aba, com o voucher emitido, com todas as informações do voucher.

Fluxo Alternativo 1

1. O usuário pode selecionar para efetuar o download do voucher.
2. O sistema deve gerar um arquivo .pdf com as informações do voucher.

Fluxo Alternativo 2

1. O usuário pode selecionar para imprimir o voucher.
2. O sistema deve imprimir o voucher.

4.2.25 Caso de Uso – Validar Vouchers

Caso de Uso: Validar Vouchers (CSU25).

Descrição: O parceiro deve informar ao sistema o código do voucher e cpf do comprador para a validação do voucher.

Ator Primário: Parceiro.

Pré-condições: O usuário deve realizar o *login* no sistema de parceiro.

Fluxo Principal

1. Após efetuar o login, o parceiro deve ir a tela inicial e selecionar no menu validar voucher.
2. O sistema abre a funcionalidade na aba dados com todos os campos em branco.
3. O sistema solicita dados obrigatórios da validação do voucher, sendo eles cpf do comprador e código do voucher.
4. O parceiro preenche os dados solicitados.
5. O parceiro clica no botão validar.
6. O sistema salva os dados no banco de dados e envia uma mensagem de confirmação com o texto “Voucher validado com sucesso!” na interface do parceiro.

4.2.26 Caso de Uso – Estornar Vendas

Caso de Uso: Estornar Vendas (CSU26).

Descrição: O cliente deve pedir estorno das suas compras.

Ator Primário: Cliente.

Pré-condições: O usuário deve realizar o *login* no sistema de cliente.

Fluxo Principal

1. Ao efetuar o *login* o cliente deve ir até o menu e clicar no botão de estornar
2. Após isso, o sistema irá direcionar o cliente a tela de estornar venda.
3. O cliente deve selecionar a quantidade de produtos que quer estornar
4. Após isso, o cliente deve clicar no botão “pedir estorno”.
5. O sistema irá direcionar o cliente a tela de estorno, que será solicitado que o cliente forneça o motivo do estorno

Fluxo Alternativo 1: Confirmar estorno

1. Após o passo 01 o cliente pode desejar aceitar fazer o estorno
2. O sistema solicita confirmação do pedido de estorno.
3. O administrador clica no botão sim e faz a confirmação retornando a mensagem "Pedido de estorno enviado com sucesso!".

4.2.27 Caso de Uso – Consultar Favoritos

Caso de Uso: Consultar Favoritos (CSU27).

Descrição: O cliente deve consultar os favoritos que consta na base de dados.

Ator Primário: Cliente.

Pré-condições: O usuário deve realizar o *login* no sistema de cliente.

Fluxo Principal

1. Ao efetuar o login o cliente deve ir até o menu e clicar no botão referente aos favoritos.
2. Após isso, o sistema irá direcionar o cliente à tela de favoritos.

Fluxo Alternativo 1: Excluir favorito

1. Sendo possível remover o produto dos favoritos ao clicar no ícone de coração.

4.2.28 Caso de Uso – Consultar Categorias

Caso de Uso: Consultar Categorias (CSU28).

Descrição: O administrador deve consultar um categorias que consta na base de dados.

Ator Primário: Administrador.

Pré-condições: O usuário deve realizar o *login* no sistema de administrador.

Fluxo Principal

1. Ao efetuar o login o administrador deve ir até o menu e clicar no botão consultar cliente.
2. Após isso, o sistema irá direcionar o administrador a tela de consulta de categorias, com o campo de busca em branco.

3. O administrador deve preencher o campo de busca com o nome da categoria desejado para buscar.
4. Após isso, o administrador deve clicar na lupa, que se refere a buscar.
5. O sistema fornece uma lista de clientes de acordo com texto digitado.

Fluxo Alternativo 1: Excluir Categoria

1. Após o passo 01 o administrador pode desejar excluir o cadastro de um cliente.
2. O administrador clica no botão excluir.
3. O sistema solicita confirmação da exclusão.
4. O administrador clica no botão sim e faz a confirmação retornando a mensagem "Registro excluído com sucesso!".

Fluxo Alternativo 2: Editar Categoria

1. Após o passo 01 o administrador pode desejar editar o cadastro de um cliente.
2. O administrador clica no botão editar.
3. O sistema preenche os campos da funcionalidade com os dados da cliente selecionada.
4. O administrador edita as informações que quer alterar.
5. O sistema solicita confirmação da edição.
6. O administrador clica no botão "sim" e faz a confirmação retornando a mensagem "Registro atualizado com sucesso!".

4.2.29 Caso de Uso – Consultar Clientes

Caso de Uso: Consultar Clientes (CSU29).

Descrição: O administrador deve consultar um cliente que consta na base de dados.

Ator Primário: Administrador.

Pré-condições: O usuário deve realizar o *login* no sistema de administrador.

Fluxo Principal

6. Ao efetuar o login o administrador deve ir até o menu e clicar no botão consultar cliente.

7. Após isso, o sistema irá direcionar o administrador a tela de consulta de clientes, com o campo de busca em branco.
8. O administrador deve preencher o campo de busca com o nome ou CPF do cliente desejado para buscar.
9. Após isso, o administrador deve clicar na lupa, que se refere a buscar.
10. O sistema fornece uma lista de clientes de acordo com texto digitado.

Fluxo Alternativo 1: Excluir Cliente

5. Após o passo 01 o administrador pode desejar excluir o cadastro de um cliente.
6. O administrador clica no botão excluir.
7. O sistema solicita confirmação da exclusão.
8. O administrador clica no botão sim e faz a confirmação retornando a mensagem "Registro excluído com sucesso!".

Fluxo Alternativo 2: Editar Cliente

7. Após o passo 01 o administrador pode desejar editar o cadastro de um cliente.
8. O administrador clica no botão editar.
9. O sistema preenche os campos da funcionalidade com os dados da cliente selecionada.
10. O administrador edita as informações que quer alterar.
11. O sistema solicita confirmação da edição.
12. O administrador clica no botão "sim" e faz a confirmação retornando a mensagem "Registro atualizado com sucesso!".

Fluxo Alternativo 3: Visualizar Cliente

1. Após o passo 01 o administrador pode desejar visualizar o cadastro de um cliente.
2. O administrador clica no botão visualizar.
3. Abre um *pop-up* para visualização das informações.
4. O administrador clicar em "Fechar"

4.2.30 Caso de Uso – Consultar Parceiros

Caso de Uso: Consultar Parceiros(CSU30).

Descrição: O administrador deve consultar um parceiro que consta na base de dados.

Ator Primário: Administrador.

Pré-condições: O usuário deve realizar o *login* no sistema de administrador.

Fluxo Principal

1. Ao efetuar o login o administrador deve ir até o menu e clicar no botão consultar parceiro.
2. Após isso, o sistema irá direcionar o administrador à tela de consulta de parceiro, com o campo de busca em branco.
3. O administrador deve preencher o campo de busca com o nome ou CNPJ do parceiro desejado para buscar.
4. Após isso, o administrador deve clicar na lupa, que se refere a buscar.
5. O sistema fornece uma lista de parceiros de acordo com texto digitado.

Fluxo Alternativo 1: Excluir Parceiro

1. Após o passo 01 o administrador pode desejar excluir o cadastro de um parceiro.
2. O administrador clica no botão excluir.
3. O sistema solicita confirmação da exclusão.
4. O administrador clica no botão sim e faz a confirmação retornando a mensagem "Registro excluído com sucesso!".

Fluxo Alternativo 2: Editar Parceiro

1. Após o passo 01 o administrador pode desejar editar o cadastro de um parceiro.
2. O administrador clica no botão editar.
3. O sistema preenche os campos da funcionalidade com os dados do parceiro selecionado.
4. O administrador edita as informações que quer alterar.
5. O sistema solicita confirmação da edição.
6. O administrador clica no botão "sim" e faz a confirmação retornando a mensagem "Registro atualizado com sucesso!".

Fluxo Alternativo 3: Visualizar Parceiro

1. Após o passo 01 o administrador pode desejar visualizar o cadastro de um parceiro.
2. O administrador clica no botão visualizar.
3. Abre um *pop-up* para visualização das informações.

4. O administrador clicar em “Fechar”

4.2.31 Caso de Uso – Consultar Cupons

Caso de Uso: Consultar Cupons(CSU31).

Descrição: O administrador deve consultar um cupom que consta na base de dados.

Ator Primário: Administrador.

Pré-condições: O usuário deve realizar o *login* no sistema de administrador.

Fluxo Principal

1. Ao efetuar o login o administrador deve ir até o menu e clicar no botão consultar cupons.
2. Após isso, o sistema irá direcionar o administrador à tela de consulta de cupons, com o campo de busca em branco.
3. O administrador deve preencher o campo de busca com o nome do cupom desejado para buscar.
4. Após isso, o administrador deve clicar na lupa, que se refere a buscar.
5. O sistema fornece uma lista de parceiros de acordo com texto digitado.

Fluxo Alternativo 1: Excluir Cupons

5. Após o passo 01 o administrador pode desejar excluir o cadastro de um cupom.
6. O administrador clica no botão excluir.
7. O sistema solicita confirmação da exclusão.
8. O administrador clica no botão sim e faz a confirmação retornando a mensagem "Registro excluído com sucesso!".

Fluxo Alternativo 2: Editar Cupons

7. Após o passo 01 o administrador pode desejar editar o cadastro de um cupom.
8. O administrador clica no botão editar.
9. O sistema preenche os campos da funcionalidade com os dados do cupom selecionado.
10. O administrador edita as informações que quer alterar.
11. O sistema solicita confirmação da edição.

12. O administrador clica no botão "sim" e faz a confirmação retornando a mensagem "Registro atualizado com sucesso!".

Fluxo Alternativo 3: Visualizar Cupons

5. Após o passo 01 o administrador pode desejar visualizar o cadastro de um cupom.
6. O administrador clica no botão visualizar.
7. Abre um *pop-up* para visualização das informações.
8. O administrador clicar em "Fechar"

4.2.32 Caso de Uso – Consultar Agendas

Caso de Uso: Consultar Agendas(CSU32).

Descrição: O administrador deve consultar uma agenda que consta na base de dados.

Ator Primário: Administrador.

Pré-condições: O usuário deve realizar o *login* no sistema de administrador.

Fluxo Principal

1. Ao efetuar o login o administrador deve ir até o menu e clicar no botão consultar agenda.
2. Após isso, o sistema irá direcionar o administrador à tela de consulta de agenda, com o campo de busca em branco.
3. O administrador deve preencher o campo de busca com o nome do cupom desejado para buscar.
4. Após isso, o administrador deve clicar na lupa, que se refere a buscar.
5. O sistema fornece uma lista de parceiros de acordo com texto digitado.

Fluxo Alternativo 1: Excluir Agendas

1. Após o passo 01 o administrador pode desejar excluir o cadastro de uma agenda.
2. O administrador clica no botão excluir.
3. O sistema solicita confirmação da exclusão.
4. O administrador clica no botão sim e faz a confirmação retornando a mensagem "Registro excluído com sucesso!".

Fluxo Alternativo 2: Editar Agendas

1. Após o passo 01 o administrador pode desejar editar o cadastro de um agenda.
2. O administrador clica no botão editar.
3. O sistema preenche os campos da funcionalidade com os dados do cupom selecionado.
4. O administrador edita as informações que quer alterar.
5. O sistema solicita confirmação da edição.
6. O administrador clica no botão "sim" e faz a confirmação retornando a mensagem "Registro atualizado com sucesso!".

Fluxo Alternativo 3: Visualizar Agendas

1. Após o passo 01 o administrador pode desejar visualizar o cadastro de uma agenda.
2. O administrador clica no botão visualizar.
3. Abre um *pop-up* para visualização das informações.
4. O administrador clicar em “Fechar”

4.2.33 Caso de Uso – Consultar Vendas

Caso de Uso: Consultar Vendas (CSU33).

Descrição: O administrador deve consultar uma venda que consta na base de dados.

Ator Primário: Administrador.

Pré-condições: O usuário deve realizar o *login* no sistema de administrador.

Fluxo Principal

1. Ao efetuar o login o administrador deve ir até o menu e clicar no botão consultar venda.
2. Após isso, o sistema irá direcionar o administrador à tela de consulta de agenda, com o campo de busca em branco.
3. O administrador deve preencher o campo de busca com o número da venda desejado para buscar.
4. Após isso, o administrador deve clicar na lupa, que se refere a buscar.
5. O sistema fornece uma lista de parceiros de acordo com texto digitado.

Fluxo Alternativo 1: Visualizar Vendas

1. Após o passo 01 o administrador pode desejar visualizar o cadastro de uma venda.
2. O administrador clica no botão visualizar.
3. Abre um *pop-up* para visualização das informações.
4. O administrador clicar em “Fechar”.

4.2.34 Caso de Uso – Consultar Recebimentos

Caso de Uso: Consultar Recebimentos (CSU34).

Descrição: O administrador deve consultar um recebimento que consta na base de dados.

Ator Primário: Administrador.

Pré-condições: O usuário deve realizar o *login* no sistema de administrador.

Fluxo Principal

1. Ao efetuar o login o administrador deve ir até o menu e clicar no botão consultar recebimento.
2. Após isso, o sistema irá direcionar o administrador à tela de consulta de recebimento, com o campo de busca em branco.
3. O administrador deve preencher o campo de busca com o código de recebimento desejado para buscar.
4. Após isso, o administrador deve clicar na lupa, que se refere a buscar.
5. O sistema fornece uma lista de parceiros de acordo com texto digitado.

Fluxo Alternativo 1: Visualizar Recebimento

1. Após o passo 01 o administrador pode desejar visualizar o cadastro de um recebimento.
2. O administrador clica no botão visualizar.
3. Abre um *pop-up* para visualização das informações.
4. O administrador clicar em “Fechar”.

4.2.35 Caso de Uso – Consultar Vouchers

Caso de Uso: Consultar Vouchers (CSU35).

Descrição: O administrador deve consultar um voucher que consta na base de dados.

Ator Primário: Administrador.

Pré-condições: O usuário deve realizar o *login* no sistema de administrador.

Fluxo Principal

1. Ao efetuar o login o administrador deve ir até o menu e clicar no botão consultar voucher.
2. Após isso, o sistema irá direcionar o administrador à tela de consulta de voucher, com o campo de busca em branco.
3. O administrador deve preencher o campo de busca com o código de voucher desejado para buscar.
4. Após isso, o administrador deve clicar na lupa, que se refere a buscar.
5. O sistema fornece uma lista de parceiros de acordo com texto digitado.

Fluxo Alternativo 1: Visualizar Vouchers

1. Após o passo 01 o administrador pode desejar visualizar o cadastro de um voucher.
2. O administrador clica no botão visualizar.
3. Abre um *pop-up* para visualização das informações.
4. O administrador clicar em “Fechar”.

4.2.36 Caso de Uso – Consultar Solicitação de Cupons

Caso de Uso: Consultar Solicitação de Cupons(CSU36).

Descrição: O administrador deve consultar solicitação de cupons que consta na base de dados.

Ator Primário: Administrador.

Pré-condições: O usuário deve realizar o *login* no sistema de administrador.

Fluxo Principal

1. Ao efetuar o login o administrador deve ir até o menu e clicar no botão consultar solicitações de cupons.
2. Após isso, o sistema irá direcionar o administrador à tela de consultar solicitação de cupons, com o campo de busca em branco.
3. O administrador deve preencher o campo de busca com o nome da solicitação do cupom desejado para buscar.
4. Após isso, o administrador deve clicar na lupa, que se refere a buscar.

5. O sistema fornece uma lista de parceiros de acordo com texto digitado.

Fluxo Alternativo 1: Visualizar Solicitação de Cupons

1. Após o passo 01 o administrador pode desejar visualizar o cadastro de uma solicitação de cupom.
2. O administrador clica no botão visualizar.
3. Abre um *pop-up* para visualização das informações.
4. O administrador clicar em “Fechar”.

4.2.37 Caso de Uso – Consultar Solicitação de Parceiros

Caso de Uso: Consultar Solicitação de Parceiros (CSU37).

Descrição: O administrador deve consultar solicitação de parceiros que consta na base de dados.

Ator Primário: Administrador.

Pré-condições: O usuário deve realizar o *login* no sistema de administrador.

Fluxo Principal

1. Ao efetuar o login o administrador deve ir até o menu e clicar no botão consultar solicitações de parceiro.
2. Após isso, o sistema irá direcionar o administrador à tela de consulta de agenda, com o campo de busca em branco.
3. O administrador deve preencher o campo de busca com o nome do parceiros ou CNPJ desejado para buscar.
4. Após isso, o administrador deve clicar na lupa, que se refere a buscar.
5. O sistema fornece uma lista de parceiros de acordo com texto digitado.

Fluxo Alternativo 1: Excluir Solicitação de Parceiros

1. Após o passo 01 o administrador pode desejar excluir o cadastro de uma solicitação de parceiro.
2. O administrador clica no botão excluir.
3. O sistema solicita confirmação da exclusão.

4. O administrador clica no botão sim e faz a confirmação retornando a mensagem "Registro excluído com sucesso!".

Fluxo Alternativo 2: Editar Solicitação de Parceiros

1. Após o passo 01 o administrador pode desejar editar o cadastro de uma solicitação de parceiro.
2. O administrador clica no botão editar.
3. O sistema preenche os campos da funcionalidade com os dados da solicitação de parceiro selecionado.
4. O administrador edita as informações que quer alterar.
5. O sistema solicita confirmação da edição.
6. O administrador clica no botão "sim" e faz a confirmação retornando a mensagem "Registro atualizado com sucesso!".

4.2.38 Caso de Uso – Autorizar Solicitação de Parceiros

Caso de Uso: Autorizar Solicitação de Parceiros (CSU38).

Descrição: O administrador deve consultar as autorizações de solicitação de parceiros que constam na base de dados.

Ator Primário: Administrador.

Pré-condições: O usuário deve realizar o *login* no sistema de administrador.

Fluxo Principal

1. Ao efetuar o login o administrador deve ir até o menu e clicar no botão autorizar solicitações de parceiro.
2. Após isso, o sistema irá direcionar o administrador à tela de consulta de agenda, com o campo de busca em branco.
3. O sistema fornece uma lista de solicitações parceiros de acordo com os dados do sistema.

Fluxo Alternativo 1: Excluir Solicitação de Parceiros

1. Após o passo 01 o administrador pode desejar excluir o cadastro de uma autorização de solicitação de parceiro.

2. O administrador clica no botão excluir.
3. O sistema solicita confirmação da exclusão.
4. O administrador clica no botão sim e faz a confirmação retornando a mensagem "Registro excluído com sucesso!".

Fluxo Alternativo 2: Aceitar Solicitação de Parceiros

1. Após o passo 01 o administrador pode desejar autorizar uma solicitação de parceiro.
2. O administrador clica no botão aceitar.
3. O sistema solicita a confirmação da autorização.
4. O administrador clica no botão "sim" e faz a confirmação retornando a mensagem "Registro atualizado com sucesso!".

4.2.39 Caso de Uso – Autorizar Solicitação de Cupons

Caso de Uso: Autorizar Solicitação de Cupons (CSU39).

Descrição: O administrador deve consultar as autorizações de solicitação de cupons que constam na base de dados.

Ator Primário: Administrador.

Pré-condições: O usuário deve realizar o *login* no sistema de administrador.

Fluxo Principal

1. Ao efetuar o login o administrador deve ir até o menu e clicar no botão autorizar solicitações de parceiro.
2. Após isso, o sistema irá direcionar o administrador à tela de consulta de agenda, com o campo de busca em branco.
3. O sistema fornece uma lista de solicitações parceiros de acordo com os dados do sistema.

Fluxo Alternativo 1: Excluir Solicitação de Cupons

1. Após o passo 01 o administrador pode desejar excluir o cadastro de uma autorização de solicitação de cupons.

2. O administrador clica no botão excluir.
3. O sistema solicita confirmação da exclusão.
4. O administrador clica no botão sim e faz a confirmação retornando a mensagem "Registro excluído com sucesso!".

Fluxo Alternativo 2: Aceitar Solicitação de Cupons

1. Após o passo 01 o administrador pode desejar autorizar uma solicitação de cupons.
2. O administrador clica no botão aceitar.
3. O sistema solicita a confirmação da autorização.
4. O administrador clica no botão "sim" e faz a confirmação retornando a mensagem "Registro atualizado com sucesso!".

4.2.40 Caso de Uso – Emitir Relatório de Vendas

Caso de Uso: Emitir Relatório de Vendas (CSU40).

Descrição: O administrador deve emitir um relatório de vendas que consta na base de dados.

Ator Primário: Administrador.

Pré-condições: O usuário deve realizar o *login* no sistema de administrador.

Fluxo Principal

1. O administrador deve entrar no painel de controle administrativo disponível pela url fornecida ao administrativo.
2. O sistema abre a funcionalidade com todos os campos em branco e com o botão Emitir ativados.
3. O sistema solicita não obrigatórios os dados do relatório de vendas, sendo eles período inicial e final, parceiro, status, cliente, código de venda, cupom e código do voucher.
4. O usuário preenche os dados do relatório de vendas para emitir seu relatório.
5. O administrador clica no botão Emitir.
6. O sistema envia uma mensagem de confirmação com o texto "Relatório emitido com sucesso!" na interface do administrador.
7. O sistema gera um relatório de vouchers.

Fluxo Alternativo 1: Excluir Relatório de Vendas

1. Após o passo 01 o administrador pode desejar fechar o relatório de vendas.
2. O sistema fornece uma lista de vendas de acordo com as informações solicitadas.
3. O sistema mostra uma lista de vendas de acordo com as informações informadas pelo administrador.
4. Quando o administrador clicar no botão fechar volta para o painel central administrativo.

Fluxo Alternativo 2: Baixar relatório de Vendas

1. Após o passo 01 o administrador pode desejar baixar o relatório das vendas.
2. O administrador clica no ícone de download.
3. O download será baixado em sua máquina com o relatório das vendas especificado nas funcionalidades solicitadas.

Fluxo Alternativo 3: Imprimir relatório de Vendas

1. Após o passo 01 o administrador pode desejar imprimir o relatório dos Vendas.
2. O administrador ícone de imprimir.
3. O administrador terá uma página em sua tela que mostrará o conteúdo a ser impresso.
4. O administrador clica no botão imprimir para realizar a ação.
5. O botão cancelar no fluxo alternativo volta para onde estava anteriormente.

4.2.41 Caso de Uso – Emitir Relatório de Vouchers**Caso de Uso: Emitir Relatório de Vouchers(CSU41).**

Descrição: O administrador deve emitir um relatório de vouchers que consta na base de dados.

Ator Primário: Administrador.

Pré-condições: O usuário deve realizar o *login* no sistema de administrador.

Fluxo Principal

1. O administrador deve entrar no painel de controle administrativo disponível pela url fornecida ao administrativo.
2. O sistema abre a funcionalidade com todos os campos em branco e com o botão Emitir ativados.

3. O sistema solicita não obrigatórios dos Vouchers, sendo eles período inicial e final, parceiro, status, cliente, código de venda, cupom e código de voucher.
4. O usuário preenche os dados do voucher para emitir seu relatório.
5. O administrador clica no botão Emitir.
6. O sistema envia uma mensagem de confirmação com o texto "Relatório emitido com sucesso!" na interface do administrador.
7. O sistema gera um relatório de vouchers.

Fluxo Alternativo 1: Fechar relatório de Vouchers

1. Após o passo 01 o administrador pode desejar fechar o relatório de um voucher.
2. O sistema fornece uma lista de vouchers de acordo com as informações solicitadas.
3. O sistema mostra uma lista de vouchers de acordo com as informações informadas pelo administrador.
4. Quando o administrador clicar no botão fechar volta para o painel central administrativo.

Fluxo Alternativo 2: Baixar relatório de Vouchers

1. Após o passo 01 o administrador pode desejar baixar o relatório dos vouchers.
2. O administrador clica no ícone de download.
3. O download será baixado em sua máquina com o relatório dos vouchers especificado nas funcionalidades solicitadas.

Fluxo Alternativo 3: Imprimir relatório de Vouchers

1. Após o passo 01 o administrador pode desejar imprimir o relatório dos vouchers.
2. O administrador ícone de imprimir.
3. O administrador terá uma página em sua tela que mostrará o conteúdo a ser impresso.
4. O administrador clica no botão imprimir para realizar a ação.
5. O botão cancelar no fluxo alternativo volta para onde estava anteriormente.

4.2.42 Caso de Uso – Emitir Relatório de Categorias

Caso de Uso: Emitir Relatório de Categorias (CSU42).

Descrição: O administrador deve emitir um relatório de categorias que consta na base de dados.

Ator Primário: Administrador.

Pré-condições: O usuário deve realizar o *login* no sistema de administrador.

Fluxo Principal

1. O administrador deve entrar no painel de controle administrativo disponível pela url fornecida ao administrativo.
2. O sistema abre a funcionalidade com todos os campos em branco e com o botão Emitir ativados.
3. O sistema solicita não obrigatórios da categoria, sendo eles período inicial e final, categoria e status.
4. O usuário preenche os dados do cliente para emitir seu relatório.
5. O administrador clica no botão Emitir.
6. O sistema envia uma mensagem de confirmação com o texto "Relatório emitido com sucesso!" na interface do administrador.
7. O sistema gera um relatório de categorias.

Fluxo Alternativo 1: Fechar relatório de Categoria

1. Após o passo 01 o administrador pode desejar fechar o relatório de um categorias.
2. O sistema fornece uma lista de categorias de acordo com as informações solicitadas.
3. Caso o administrador clicar no botão fechar volta para o painel central administrativo.

Fluxo Alternativo 2: Baixar relatório de Categoria

1. Após o passo 01 o administrador pode desejar baixar o relatório de categorias.
2. O administrador clica no ícone de download.
3. O download será baixado em sua máquina com o relatório das categorias especificado nas funcionalidades solicitadas.

Fluxo Alternativo 3: Imprimir relatório de Categoria

1. Após o passo 01 o administrador pode desejar imprimir o relatório de categorias.
2. O administrador ícone de imprimir.
3. O administrador terá uma página em sua tela que mostrará o conteúdo a ser impresso.
4. O administrador clica no botão imprimir para realizar a ação.
5. O botão cancelar no fluxo alternativo volta para onde estava anteriormente.

4.2.43 Caso de Uso – Emitir Relatório de Clientes

Caso de Uso: Emitir Relatório de Clientes (CSU43).

Descrição: O administrador deve emitir um relatório de cliente que consta na base de dados.

Ator Primário: Administrador.

Pré-condições: O usuário deve realizar o *login* no sistema de administrador.

Fluxo Principal

1. O administrador deve entrar no painel de controle administrativo disponível pela url fornecida ao administrativo.
2. O sistema abre a funcionalidade com todos os campos em branco e com o botão Emitir ativados.
3. O sistema solicita não obrigatórios do Cliente, sendo eles período inicial e final, cpf, razão social e status.
4. O usuário preenche os dados do cliente para emitir seu relatório.
5. O administrador clica no botão Emitir.
6. O sistema envia uma mensagem de confirmação com o texto "Relatório emitido com sucesso!" na interface do administrador.
7. O sistema gera um relatório de clientes.

Fluxo Alternativo 1: Fechar relatório de Cliente

1. Após o passo 01 o administrador pode desejar fechar o relatório de um cliente.
2. O sistema fornece uma lista de clientes de acordo com as informações solicitadas.
3. O sistema mostra uma lista de clientes de acordo com as informações informadas pelo administrador.
4. Quando o administrador clicar no botão fechar volta para o painel central administrativo.

Fluxo Alternativo 2: Baixar relatório de Cliente

1. Após o passo 01 o administrador pode desejar baixar o relatório do cliente.
2. O administrador clica no ícone de download.
3. O download será baixado em sua máquina com o relatório dos clientes especificado nas funcionalidades solicitadas.

Fluxo Alternativo 3: Imprimir relatório de Cliente

1. Após o passo 01 o administrador pode desejar imprimir o relatório do cliente.
2. O administrador ícone de imprimir.
3. O administrador terá uma página em sua tela que mostrará o conteúdo a ser impresso.
4. O administrador clica no botão imprimir para realizar a ação.
5. O botão cancelar no fluxo alternativo volta para onde estava anteriormente.

4.2.44 Caso de Uso – Emitir Relatório de Parceiros

Caso de Uso: Emitir Relatório de Parceiros (CSU44).

Descrição: O administrador deve emitir um relatório de parceiros que consta na base de dados.

Ator Primário: Administrador.

Pré-condições: O usuário deve realizar o *login* no sistema de administrador.

Fluxo Principal:

1. O administrador deve entrar no painel de controle administrativo disponível pela url fornecida ao administrativo.
2. O sistema abre a funcionalidade com todos os campos em branco e com o botão Emitir ativados.
3. O sistema solicita não obrigatórios dos Parceiros, sendo eles período inicial e final, CNPJ, razão social e status.
4. O usuário preenche os dados do parceiro para emitir seu relatório.
5. O administrador clica no botão Emitir.
6. O sistema envia uma mensagem de confirmação com o texto "Relatório emitido com sucesso!" na interface do administrador.

7. O sistema gera um relatório de parceiros.

Fluxo Alternativo 1: Fechar relatório de Parceiros

1. Após o passo 01 o administrador pode desejar fechar o relatório de um parceiro.
2. O sistema fornece uma lista de parceiros de acordo com as informações solicitadas.
3. O sistema mostra uma lista de parceiros de acordo com as informações informadas pelo administrador.
4. Quando o administrador clicar no botão fechar volta para o painel central administrativo.

Fluxo Alternativo 2: Baixar relatório de Parceiros

1. Após o passo 01 o administrador pode desejar baixar o relatório dos parceiros.
2. O administrador clica no ícone de download.
3. O download será baixado em sua máquina com o relatório dos parceiros especificado nas funcionalidades solicitadas.

Fluxo Alternativo 3: Imprimir relatório de Parceiros

1. Após o passo 01 o administrador pode desejar imprimir o relatório dos parceiros.
2. O administrador ícone de imprimir.
3. O administrador terá uma página em sua tela que mostrará o conteúdo a ser impresso.
4. O administrador clica no botão imprimir para realizar a ação.
5. O botão cancelar no fluxo alternativo volta para onde estava anteriormente.

4.2.45 Caso de Uso – Emitir Relatório de Cupons

Caso de Uso: Emitir Relatório de Cupons (CSU45).

Descrição: O administrador deve emitir um relatório de cupons que consta na base de dados.

Ator Primário: Administrador.

Pré-condições: O usuário deve realizar o *login* no sistema de administrador.

Fluxo Principal:

1. O administrador deve entrar no painel de controle administrativo disponível pela url fornecida ao administrativo.
2. O sistema abre a funcionalidade com todos os campos em branco e com o botão Emitir ativados.
3. O sistema solicita não obrigatórios do produto, sendo eles período inicial e final, categoria, status e parceiro.
4. O usuário preenche os dados do produto para emitir seu relatório.
5. O administrador clica no botão Emitir.
6. O sistema envia uma mensagem de confirmação com o texto "Relatório emitido com sucesso!" na interface do administrador.
7. O sistema gera um relatório de produtos.

Fluxo Alternativo 1: Fechar relatório de Produto

1. Após o passo 01 o administrador pode desejar fechar o relatório de um cupom.
2. O sistema fornece uma lista de cupom de acordo com as informações solicitadas.
3. Caso o administrador clicar no botão fechar volta para o painel central administrativo.

Fluxo Alternativo 2: Baixar relatório de Produto

1. Após o passo 01 o administrador pode desejar baixar o relatório de cupom.
2. O administrador clica no ícone de download.
3. O download será baixado em sua máquina com o relatório dos cupons especificados nas funcionalidades solicitadas.

Fluxo Alternativo 3: Imprimir relatório de Produto

1. Após o passo 01 o administrador pode desejar imprimir o relatório de cupom.
2. O administrador ícone de imprimir.
3. O administrador terá uma página em sua tela que mostrará o conteúdo a ser impresso.
4. O administrador clica no botão imprimir para realizar a ação.
5. O botão cancelar no fluxo alternativo volta para onde estava anteriormente.

4.2.46 Caso de Uso – Emitir Relatório de Agendas

Caso de Uso: Emitir Relatório de Agendas (CSU46).

Descrição: O administrador deve emitir um relatório de agenda que consta na base de dados.

Ator Primário: Administrador.

Pré-condições: O usuário deve realizar o *login* no sistema de administrador.

Fluxo Principal:

1. O administrador deve entrar no painel de controle administrativo disponível pela url fornecida ao administrativo.
2. O sistema abre a funcionalidade com todos os campos em branco e com o botão Emitir ativados.
3. O sistema solicita não obrigatórios os dados do relatório de agenda, sendo eles período inicial e final, cupons, status e parceiro.
4. O usuário preenche os dados do relatório de agenda para emitir seu relatório.
5. O administrador clica no botão Emitir.
6. O sistema envia uma mensagem de confirmação com o texto "Relatório emitido com sucesso!" na interface do administrador.
7. O sistema gera um relatório de agenda.

Fluxo Alternativo 1: Fechar relatório de Agenda

1. Após o passo 01 o administrador pode desejar fechar o relatório de agenda.
2. O sistema fornece uma lista de agenda de acordo com as informações solicitadas.
3. O sistema mostra uma lista de agenda de acordo com as informações informadas pelo administrador.
4. Quando o administrador clicar no botão fechar volta para o painel central administrativo.

Fluxo Alternativo 2: Baixar relatório de Agenda

1. Após o passo 01 o administrador pode desejar baixar o relatório das agendas.
2. O administrador clica no ícone de download.
3. O download será baixado em sua máquina com o relatório das agendas especificadas nas funcionalidades solicitadas.

Fluxo Alternativo 3: Imprimir relatório de Agenda

1. Após o passo 01 o administrador pode desejar imprimir o relatório das agendas.
2. O administrador ícone de imprimir.
3. O administrador terá uma página em sua tela que mostrará o conteúdo a ser impresso.
4. O administrador clica no botão imprimir para realizar a ação.
5. O botão cancelar no fluxo alternativo volta para onde estava anteriormente

4.2.47 Caso de Uso – Emitir Relatório de Favoritos

Caso de Uso: Emitir Relatório de Favoritos (CSU47).

Descrição: O administrador deve emitir um relatório de agenda que consta na base de dados.

Ator Primário: Administrador.

Pré-condições: O usuário deve realizar o *login* no sistema de administrador.

Fluxo Principal:

1. O administrador deve entrar no painel de controle administrativo disponível pela url fornecida ao administrativo.
2. O sistema abre a funcionalidade com todos os campos em branco e com o botão Emitir ativados.
3. O sistema solicita não obrigatórios dos Favoritos, sendo eles período inicial e final, cliente, cupom e parceiro.
4. O usuário preenche os dados do parceiro para emitir seu relatório.
5. O administrador clica no botão Emitir.
6. O sistema envia uma mensagem de confirmação com o texto "Relatório emitido com sucesso!" na interface do administrador.
7. O sistema gera um relatório de favoritos.

Fluxo Alternativo 1: Fechar relatório de Favoritos

1. Após o passo 01 o administrador pode desejar fechar o relatório de um favorito.
2. O sistema fornece uma lista de favoritos de acordo com as informações solicitadas.
3. O sistema mostra uma lista de favoritos de acordo com as informações informadas pelo administrador.

4. Quando o administrador clicar no botão fechar volta para o painel central administrativo.

Fluxo Alternativo 2: Baixar relatório de Favoritos

1. Após o passo 01 o administrador pode desejar baixar o relatório dos favoritos.
2. O administrador clica no ícone de download.
3. O download será baixado em sua máquina com o relatório dos favoritos especificado nas funcionalidades solicitadas.

Fluxo Alternativo 3: Imprimir relatório de Favoritos

1. Após o passo 01 o administrador pode desejar imprimir o relatório dos favoritos.
2. O administrador ícone de imprimir.
3. O administrador terá uma página em sua tela que mostrará o conteúdo a ser impresso.
4. O administrador clica no botão imprimir para realizar a ação.
5. O botão cancelar no fluxo alternativo volta para onde estava anteriormente.

4.2.48 Caso de Uso – Emitir Relatório de Recebimentos

Caso de Uso: Emitir Relatório de Recebimentos (CSU48).

Descrição: O administrador deve emitir um relatório de recebimento que consta na base de dados.

Ator Primário: Administrador.

Pré-condições: O usuário deve realizar o *login* no sistema de administrador.

Fluxo Principal:

1. O administrador deve entrar no painel de controle administrativo disponível pela url fornecida ao administrativo.
2. O sistema abre a funcionalidade com todos os campos em branco e com o botão Emitir ativados.
3. O sistema solicita não obrigatórios os dados do relatório de recebimentos, sendo eles período inicial e final, cliente, forma de pagamento, parceiro, código de venda e status.

4. O administrador pode selecionar quais conta ele deseja como todas, em aberto, fechadas e em processo.
5. O administrador pode escolher a forma de pagamento, como crédito, débito, dinheiro e pix.
6. O usuário preenche os dados do relatório de recebimentos para emitir seu relatório.
7. O administrador clica no botão Emitir.
8. O sistema envia uma mensagem de confirmação com o texto "Relatório emitido com sucesso!" na interface do administrador.
9. O sistema gera um relatório de recebimentos.

Fluxo Alternativo 1: Fechar relatório de Recebimentos

1. Após o passo 01 o administrador pode desejar fechar o relatório de recebimentos.
2. O sistema mostra uma lista de recebimentos de acordo com as informações informadas pelo administrador.
3. Quando o administrador clicar no botão fechar volta para o painel central administrativo.

Fluxo Alternativo 2: Baixar relatório de Recebimentos

1. Após o passo 01 o administrador pode desejar baixar o relatório de recebimento.
2. O administrador clica no ícone de download.
3. O download será baixado em sua máquina com o relatório de recebimento especificado nas funcionalidades solicitadas.

Fluxo Alternativo 3: Imprimir relatório de Recebimentos

1. Após o passo 01 o administrador pode desejar imprimir o relatório de recebimento.
2. O administrador ícone de imprimir.
3. O administrador terá uma página em sua tela que mostrará o conteúdo a ser impresso.
4. O administrador clica no botão imprimir para realizar a ação.
5. O botão cancelar no fluxo alternativo volta para onde estava anteriormente.

4.2.49 Caso de Uso – Gerar Dashboard de Parceiro

Caso de Uso: Gerar Dashboard de Parceiro (CSU49).

Descrição: O parceiro deve gerar um dashboard de parceiro que consta na base de dados.

Ator Primário: Parceiro.

Pré-condições: O usuário deve realizar o *login* no sistema de parceiro.

Fluxo Principal:

1. O parceiro deve entrar no painel de controle do parceiro.
2. O sistema abre a funcionalidade com todos os gráficos preenchidos.

4.2.50 Caso de Uso – Gerar Dashboard de Administrador

Caso de Uso: Gerar Dashboard de Administrador (CSU50).

Descrição: O parceiro deve gerar um dashboard de administrador que consta na base de dados.

Ator Primário: Administrador.

Pré-condições: O usuário deve realizar o *login* no sistema de administrador.

Fluxo Principal:

1. O parceiro deve entrar no painel de controle do administrador.
2. O sistema abre a funcionalidade com todos os gráficos preenchidos.

4.2.51 Caso de Uso – Agendar

Caso de Uso: Agendar (CSU51).

Descrição: O cliente deve selecionar a data para o consumo do produto.

Ator Primário: Cliente.

Pré-condições: O usuário deve realizar o *login* no sistema de administrador.

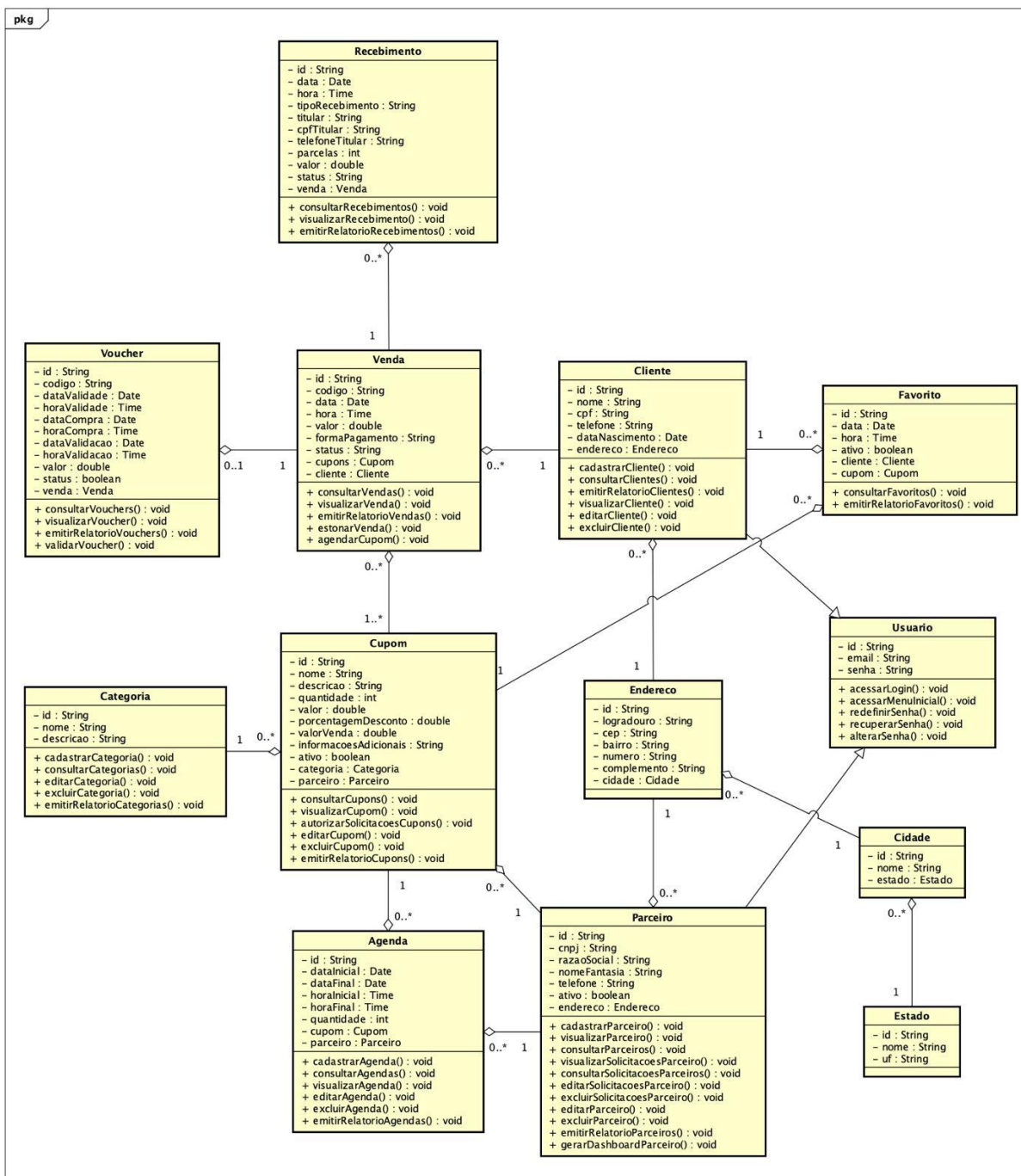
Fluxo Principal:

1. Após o cliente selecionar um produto para comprar, o sistema vai pedir a data para o determinado consumo do produto.
2. O cliente deve clicar no botão agendar o produto.
3. O cliente deve selecionar a data do produto e o horário desejado.

4. O cliente deve clicar em salvar.
5. O sistema deve sincronizar o produto com a agenda selecionada.
6. O produto será agendado com a data selecionada.

5 DIAGRAMA DE CLASSES

Figura 7 - Diagrama de classes.



Fonte: Produzido pelo autor (2024).

6 DIAGRAMAS DE SEQUÊNCIA

Serão apresentados a seguir apenas os diagramas correspondentes aos seguintes requisitos: Acessar Login Cliente, Recuperar Senha Cliente, Redefinir Senha Cliente, Alterar Senha Cliente, Cadastrar Categoria, Consultar Categoria, Acessar Menu Inicial Cliente, Comprar Cupons e Emitir Relatório de Categorias.

Figura 7 - Diagrama de Sequência - Acessar Login Cliente

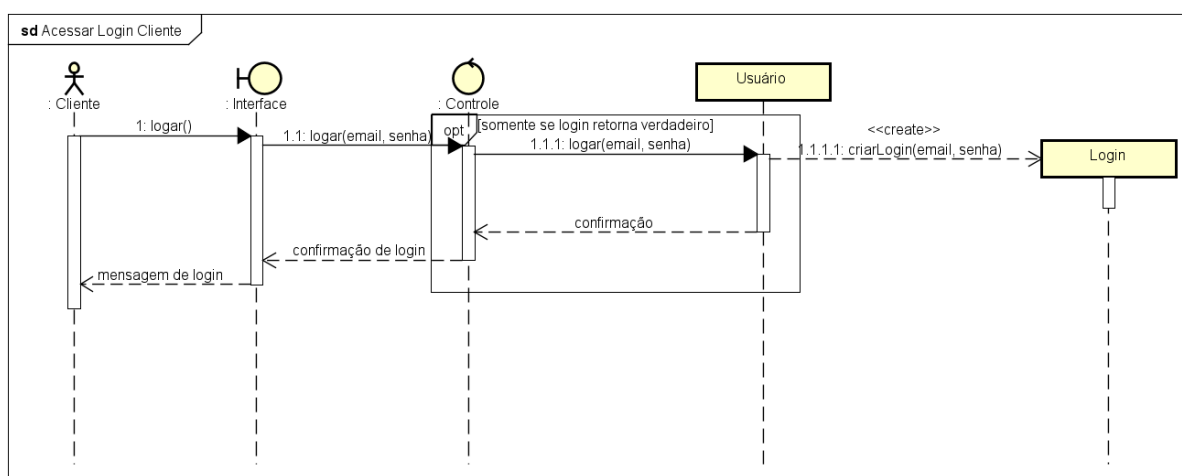


Figura 8 - Diagrama de Sequência - Recuperar Senha Cliente

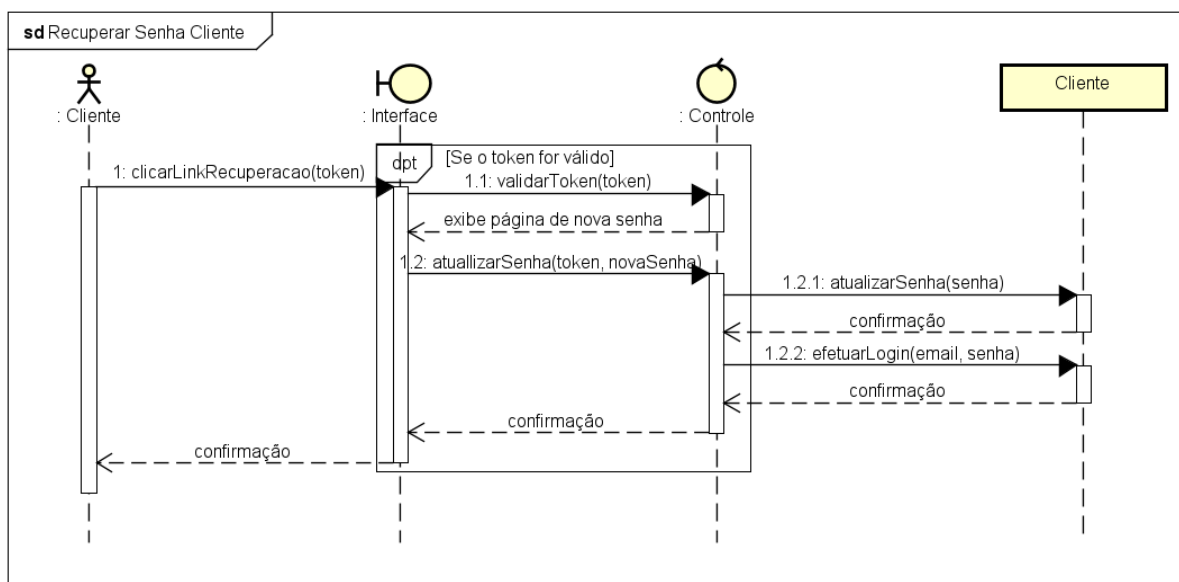


Figura 9 - Diagrama de Sequência - Redefinir Senha Cliente

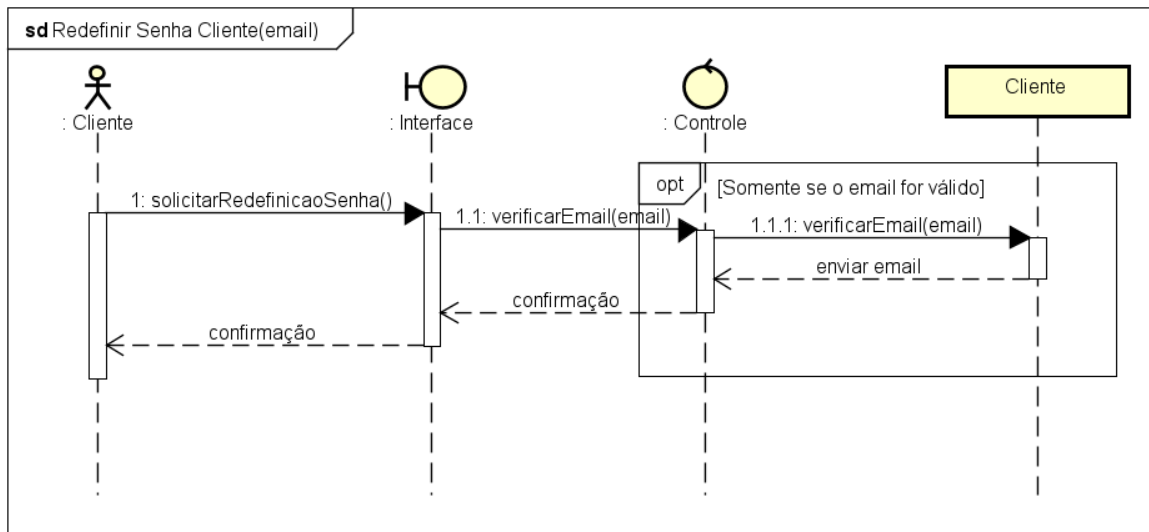


Figura 10 - Diagrama de Sequência - Alterar Senha Cliente

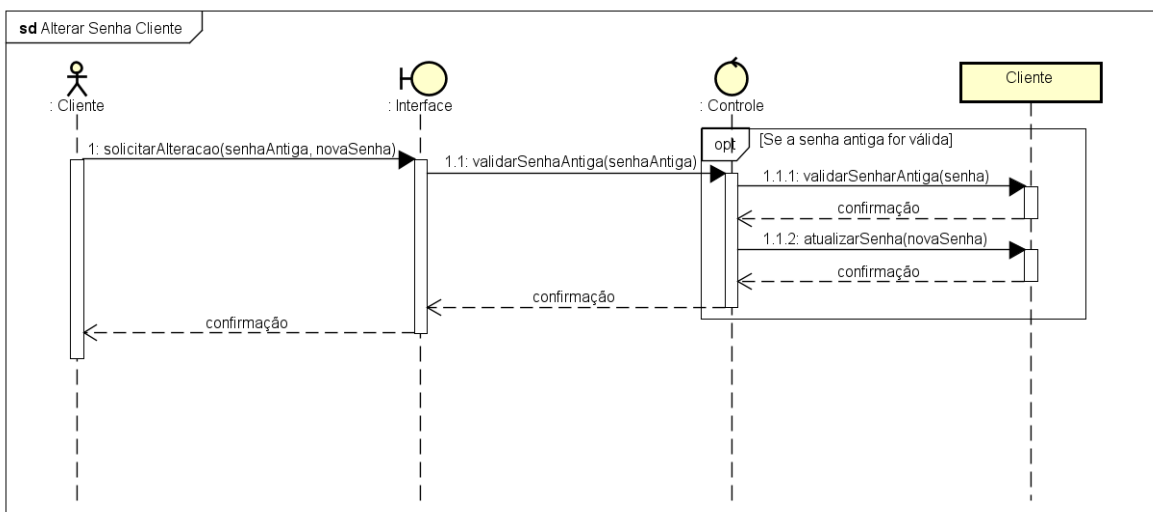


Figura 11 - Diagrama de Sequência - Cadastrar Categorias

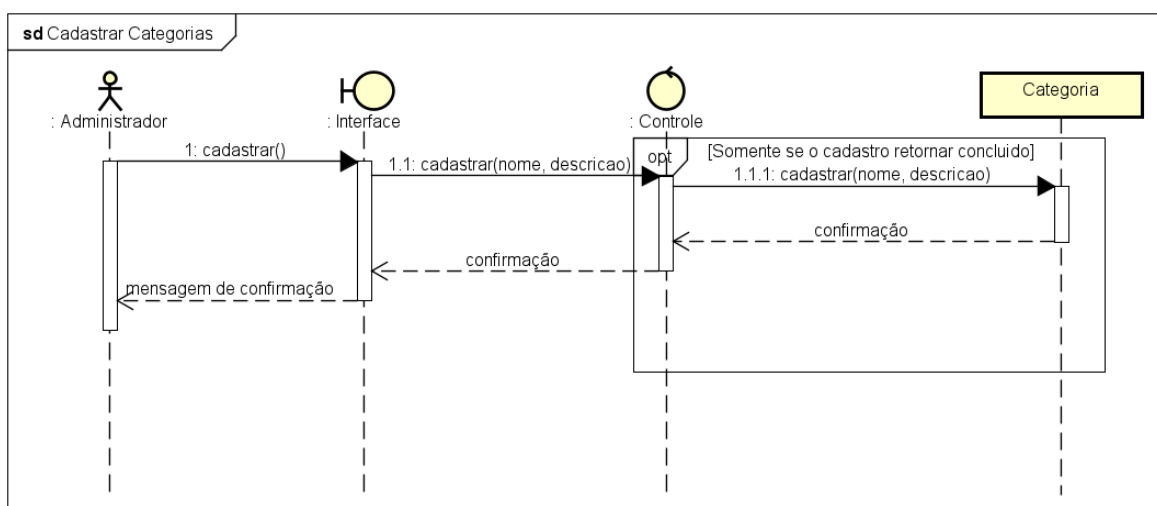


Figura 12 - Diagrama de Sequência - Consultar Categorias

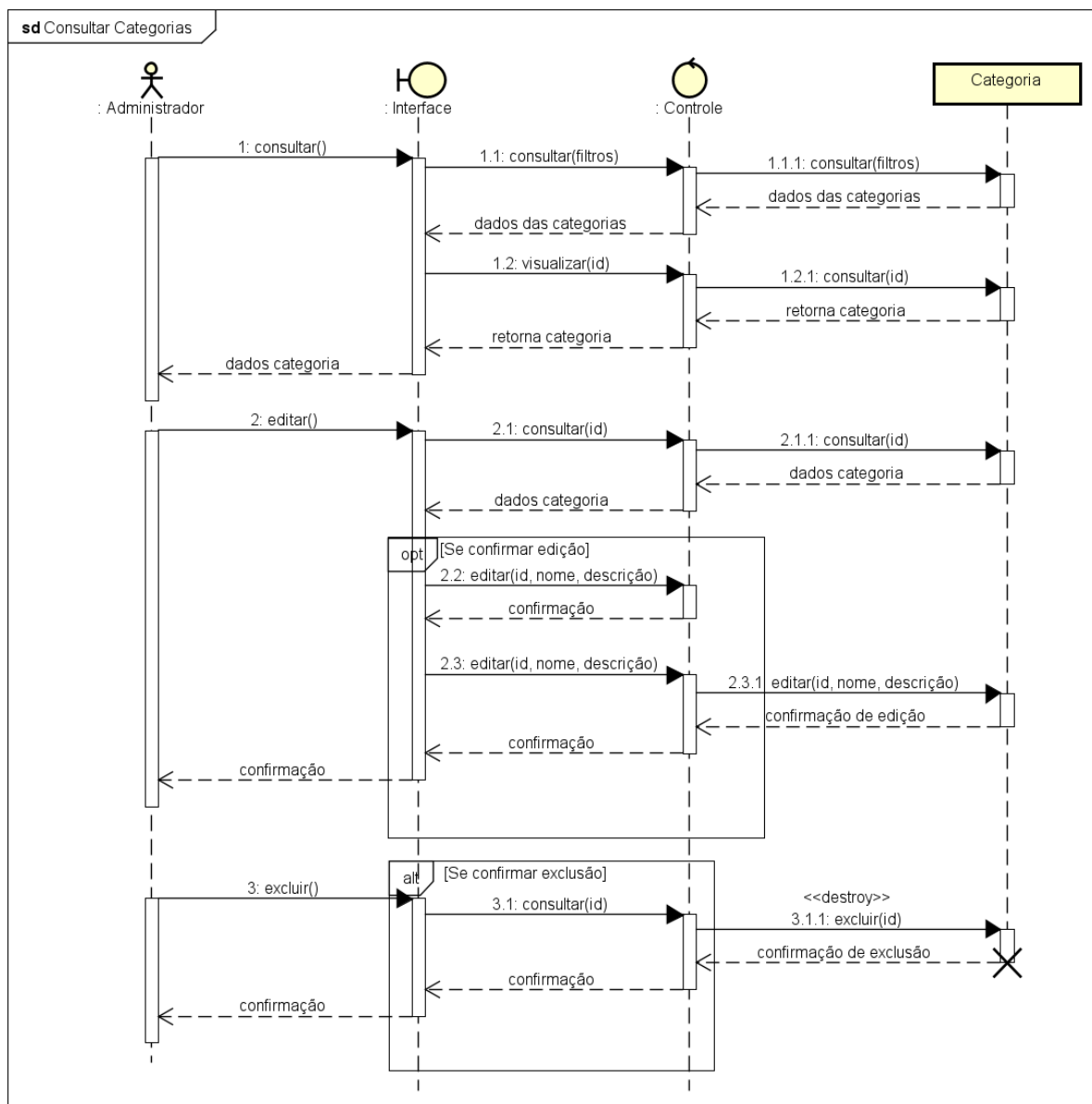


Figura 13 - Diagrama de Sequência - Acessar Menu Inicial Cliente

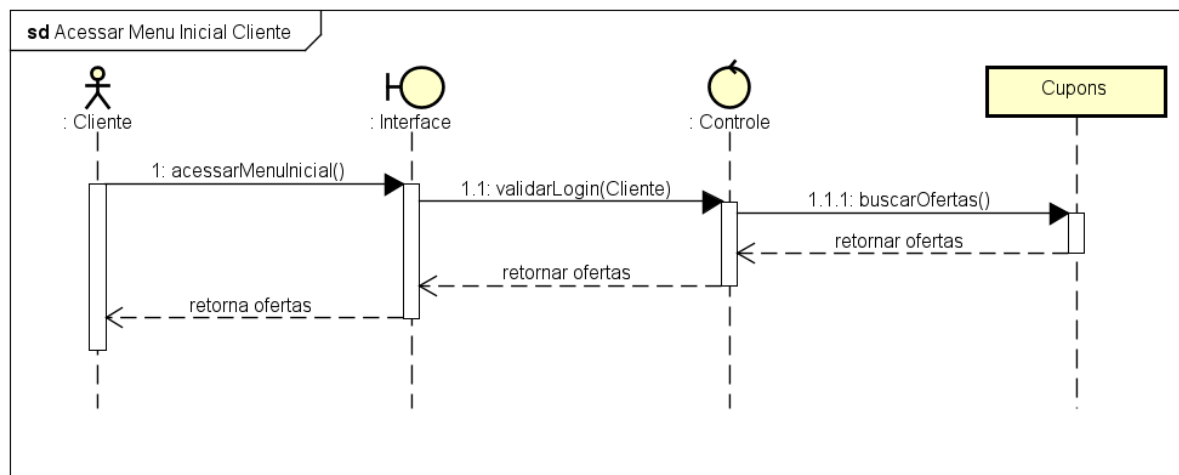


Figura 14 - Diagrama de Sequência - Comprar Cupons

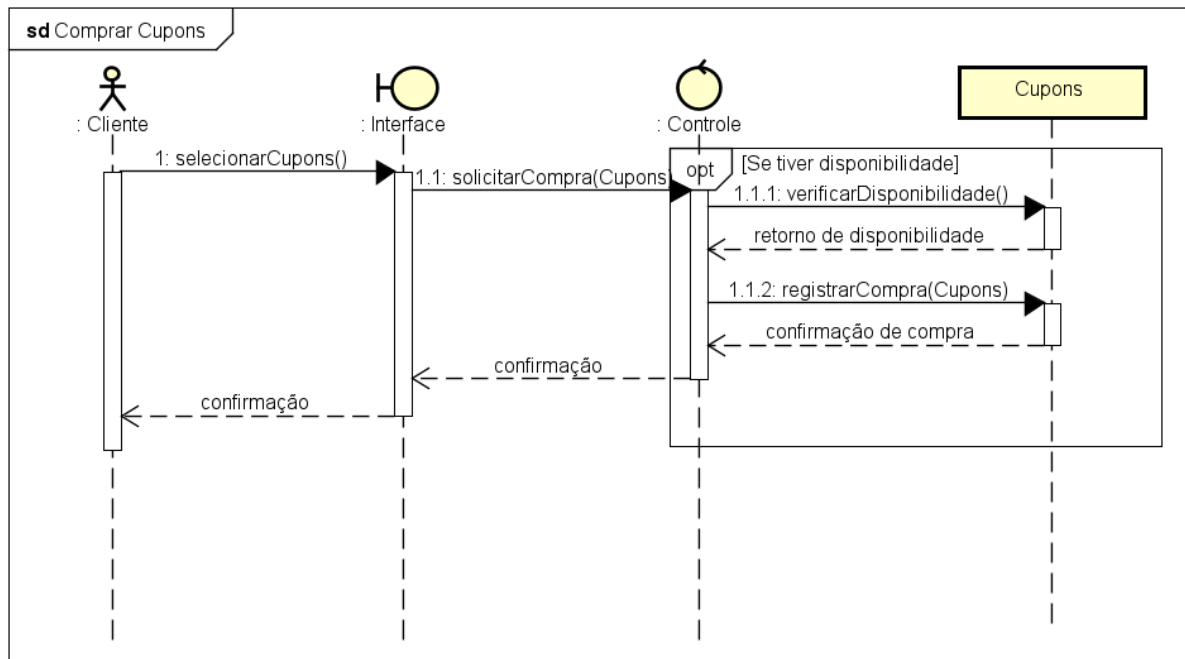
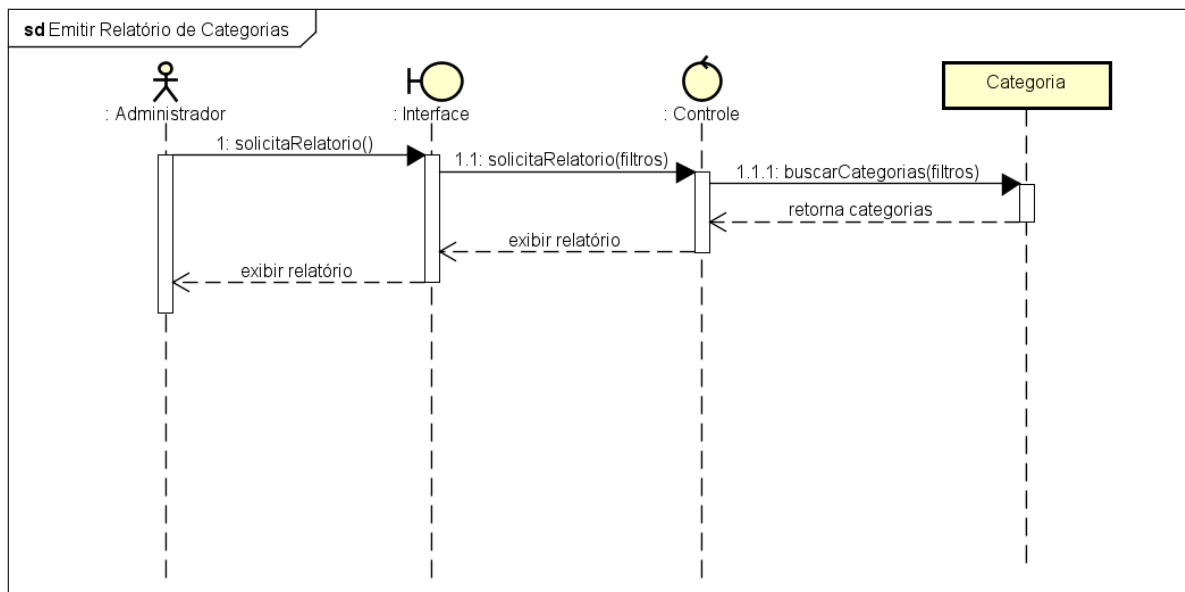


Figura 15 - Diagrama de Sequência - Emitir Relatório de Categorias



REFERÊNCIAS

AGILE ALLIANCE. Scrum. Disponível em: <https://www.agilealliance.org/glossary/scrum>. Acesso em: 30 ago. 2024.

COUTINHO, Flávio Motta. Vantagens de usar a metodologia Scrum para desenvolvimento web. **TecMundo, 2024.** Disponível em: <https://www.tecmundo.com.br/software/145932-vantagens-usar-metodologia-scrum-desenvolvimento-web.htm>. Acesso em: 27 jul. 2024.

JENSEN, Steen Lerche. Fundamentos internacionais do Scrum Master. Scrum.as, [s.d.]. Disponível em: <https://www.scrum.as/academy.php?show=5&chapter=8>. Acesso em: 20 jul. 2024.

MOUNTAIN GOAT SOFTWARE. Introduction to Scrum. Disponível em: <https://www.mountaingoatsoftware.com/agile/scrum>. Acesso em: 30 ago. 2024.

Scrum: a metodologia ágil explicada de forma definitiva. MindMaster, [s.d.]. Disponível em: <https://mindmaster.com.br/scrum>. Acesso em: 20 jul. 2024.

SCRUM.ORG. What is Scrum? Disponível em: <https://www.scrum.org/resources/what-is-scrum>. Acesso em: 30 ago. 2024.

SCRUM ALLIANCE. Scrum Framework. Disponível em: <https://www.scrumalliance.org/about-scrum/scrum-framework>. Acesso em: 30 ago. 2024.