



UNIVERSIDADE ESTADUAL DA PARAÍBA
CAMPUS I - CAMPINA GRANDE
CENTRO DE CIÊNCIAS E TECNOLOGIA
DEPARTAMENTO DE COMPUTAÇÃO
CURSO DE GRADUAÇÃO EM BACHARELADO EM CIÊNCIA DA
COMPUTAÇÃO

LUCAS MARTINS NUNES

DESENVOLVIMENTO DE UM SOFTWARE DE ENTREGAS PARA UM
CENTRO DE DISTRIBUIÇÃO

CAMPINA GRANDE - PB

2023

LUCAS MARTINS NUNES

Desenvolvimento de um Software de Entregas para um Centro de
Distribuição

Trabalho de Conclusão de Curso apresentado ao Departamento de Computação da Universidade Estadual da Paraíba, como requisito para obtenção do título de Graduação em Ciência da Computação.

Área de concentração: Desenvolvimento de Sistemas de Logística e Rastreamento de Entregas

Orientador: Prof. Dr. Paulo Eduardo e Silva Barbosa

CAMPINA GRANDE

2023

É expressamente proibido a comercialização deste documento, tanto na forma impressa como eletrônica. Sua reprodução total ou parcial é permitida exclusivamente para fins acadêmicos e científicos, desde que na reprodução figure a identificação do autor, título, instituição e ano do trabalho.

N972d Nunes, Lucas Martins.

Desenvolvimento de um *software* de entregas para um centro de distribuição [manuscrito] / Lucas Martins Nunes. - 2023.

64 p. : il. colorido.

Digitado.

Trabalho de Conclusão de Curso (Graduação em Ciência da Computação) - Universidade Estadual da Paraíba, Centro de Ciências e Tecnologia, 2023.

"Orientação : Prof. Dr. Paulo Eduardo e Silva Barbosa, Coordenação do Curso de Computação - CCT. "

1. Aplicações web. 2. Rastreamento. 3. Serviço de delivery. I. Título

21. ed. CDD 005.3

LUCAS MARTINS NUNES

**DESENVOLVIMENTO DE UM SOFTWARE DE ENTREGAS
PARA UM
CENTRO DE DISTRIBUIÇÃO**

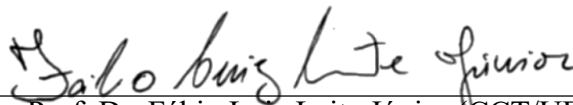
Trabalho de Conclusão de Curso de Graduação em Ciência da Computação da Universidade Estadual da Paraíba, como requisito à obtenção do título de Bacharel em Ciência da Computação.

Área de concentração: Desenvolvimento de Sistemas de Logística e Rastreamento de Entregas

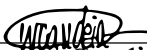
Aprovada em 21 de Agosto de 2023.



Prof. Dr. Paulo Eduardo e Silva Barbosa (CCT/UEPB)
Orientador(a)



Prof. Dr. Fábio Luiz Leite Júnior (CCT/UEPB)
Examinador(a)



Prof. Dr. Wellington Candêia de Araujo (CCT/UEPB)
Examinador(a)

Este trabalho é
dedicado às minhas
amadas avós, Adeline
e Valdíque.

AGRADECIMENTOS

Gostaria de expressar minha sincera gratidão a todos que contribuíram na minha jornada até o presente trabalho de conclusão de curso. Agradeço de coração aos meus amigos, familiares e professores por seu apoio, orientação e incentivo ao longo desta jornada acadêmica.

Agradeço especialmente à Tatiane, pelo apoio constante e incentivo durante todo o processo. Sua presença significou muito para mim e sou grato por tê-la ao meu lado.

Agradeço a todos vocês por acreditarem em mim, por me inspirarem e por serem uma fonte de força e encorajamento. Suas contribuições foram inestimáveis e tornaram essa jornada acadêmica ainda mais significativa. Estou verdadeiramente honrado e grato por ter pessoas tão especiais em minha vida.

Mais uma vez, meu sincero agradecimento a todos.

“Science is more than a body of knowledge;
it is away of thinking.” - Carl Sagan (1934
- 1996), in: The Demon-Haunted World:
Science as a Candle in the Dark.’

RESUMO

O objetivo deste trabalho é apresentar a implementação do software de entregas para um centro de distribuição. Este sistema possui um painel administrativo e uma página de rastreamento de entregas, oferecendo funcionalidades como cadastro de usuários, gerenciamento de notas fiscais, geração de lotes de entregas, rastreamento em um tempo de resposta de baixa latência e finalização de entregas com comprovantes. O software foi desenvolvido utilizando *Laravel* como *framework backend*, *Redis* e *Socket.IO* para o serviço de mensageria, *HTML*, *CSS*, *JavaScript* e *Vue.js* para o *frontend*, e *MySQL* como banco de dados relacional.

Palavras-chave: aplicações Web; Laravel; Redis; rastreamento; tempo de resposta em baixa latência.

ABSTRACT

The objective of this work is to present the implementation of the delivery software for a distribution center. This system features an administrative panel and a delivery tracking page, offering functionalities such as user registration, management of invoices, generation of delivery batches, tracking with low latency response time and, and completion of deliveries with receipts. The software was developed using *Laravel* as the *backend framework*, *Redis* and *Socket.IO* for the messaging service, *HTML*, *CSS*, *JavaScript*, and *Vue.js* for the *frontend*, and *MySQL* as the relational database.

Keywords: Web applications; Laravel; Redis; tracking; low latency response time.

LISTA DE ILUSTRAÇÕES

Figura 1 - Diagrama de Casos de Uso Geral	24
Figura 2 - Fluxo de autenticação e requisições usando JWT	25
Figura 3 - Cadastro de Notas Fiscais	25
Figura 4 - Geração de Lote de Entrega	26
Figura 5 - Representação das camadas do sistema	29
Figura 6 - Notificações Pub/Sub com <i>Redis</i> e <i>Socket.IO</i>	31
Figura 7 - Tela de Login	33
Figura 8 - Mapa de rastreamento com localização	34
Figura 9 - Informações da entrega	34
Figura 10 - Eventos da entrega	35
Figura 11 - Painel administrativo - Dashboard	36
Figura 12 - Listar Notas Fiscais	36
Figura 13 - Gerar Lote de entrega - Notas Fiscais	37
Figura 14 - Listar Entregas do usuário - Minhas entregas	37
Figura 15 - Ver detalhes da Entrega do usuário - Minhas entregas	38
Figura 16 - Iniciar entrega de nota presente no lote - Minhas entregas	38
Figura 17 - Confirmar inicio de entrega - Minhas entregas	39
Figura 18 - Finalizar entrega - Minhas entregas	39
Figura 19 - Manutenção de dados pessoais - Meus dados	40
Figura 20 - Listar Lotes de entrega	40
Figura 21 - Visualizar Lote de entrega	41
Figura 22 - Listar Centros de distribuição	41
Figura 23 - Criar Centro de distribuição	42
Figura 24 - Editar Centro de distribuição	42
Figura 25 - Excluir Centro de distribuição	43
Figura 26 - Entrada de notas fiscais - importação de XML	43
Figura 27 - Listar Notas Fiscais	44
Figura 28 - Vizualizar detalhe de Nota Fiscal	44
Figura 29 - Listar Usuários	45
Figura 30 - Criar Usuários	45

Figura 31 - Editar Usuários	46
Figura 32 - Dados do veículo	46
Figura 33 - Excluir Usuários	47
Figura 34 - Excluir Nota Fiscal	47
Figura 35 - Modelo Entidade Relacionamento	57
Figura 36 - Diagrama de tabelas do banco de dados	58
Figura 37 - Tela de rastreamento por código	60
Figura 38 - Tela de rastreamento por documentos	60
Figura 39 - Painel administrativo (Admin) - Dashboard	61
Figura 40 - Gerar Lote de entrega - Notas Fiscais (Admin/Entregador)	61
Figura 41 - Gerar Lote de entrega - Notas Fiscais (Admin/Entregador)	62
Figura 42 - Exemplo de feedback de sucesso (Admin/Entregador)	62
Figura 43 - Exemplo de Feedback de erro (Admin/Entregador)	63
Figura 44 - Logout do sistema (Admin/Entregador)	63

LISTA DE ABREVIATURAS E SIGLAS

API Application Programming Interface

CDs Centros de Distribuição

CSS Cascading Style Sheets

HTML HyperText Markup Language

JWT JSON Web Token

LGPD Lei Geral de Proteção de Dados Pessoais

PHP Hypertext Preprocessor

SQL Structured Query Language

SUMÁRIO

	Página
1	INTRODUÇÃO 13
1.1	OBJETIVOS 14
1.1.1	Objetivo Geral 14
1.1.2	Objetivos Específicos 14
2	REVISÃO BIBLIOGRÁFICA 16
2.1	Sistema de Entregas sob demanda 16
2.2	Centros de Distribuição 17
2.2.1	Mercado Livre e Amazon 17
2.3	Tecnologias Utilizadas 17
2.3.1	PHP e Laravel 18
2.3.2	Redis 18
2.3.3	Socket.IO 18
2.3.4	HTML e CSS 19
2.3.5	JavaScript 19
2.3.6	Vue.js 20
2.3.7	MySQL 20
3	METODOLOGIA 21
3.1	Levantamento de Requisitos 21
3.1.1	Requisitos Funcionais do Sistema 21
3.1.2	Requisitos Não Funcionais do Sistema 22
3.2	Projeto da Solução 23
3.3	Implementação 26
3.4	Testes e Validação 27
4	DESENVOLVIMENTO DO SOFTWARE 28
4.1	Arquitetura do Sistema 28
4.2	Backend com Laravel 29
4.3	Serviço de <i>Broadcasting</i> com <i>Redis</i> e <i>Socket.IO</i> 29

		12
4.4	Frontend com HTML, CSS, JavaScript e Vue.js	31
4.5	Banco de Dados Relacional MySQL	32
5	FUNCIONALIDADES DO SISTEMA	33
5.1	Tela de Login e Página de Rastreamento	33
5.2	Painel Administrativo	35
5.2.1	Funcionalidades para Entregadores	35
5.2.2	Funcionalidade para Administradores	40
6	RESULTADOS E DISCUSSÃO	48
6.1	Testes e Validação do Software	48
6.2	Avaliação das Funcionalidades Apresentadas	49
7	CONSIDERAÇÕES FINAIS	51
7.1	Contribuições do Trabalho	51
7.2	Limitações e possíveis melhorias	51
7.3	Conclusão	51
	REFERÊNCIAS	53
	APÊNDICE A – MODELO ENTIDADE RELACIONAMENTO	56
	APÊNDICE B – DEMAIS TELAS DO SISTEMA	59
	APÊNDICE C – REPOSITÓRIO GIT	64

1 INTRODUÇÃO

Com o advento da “Era Digital” e o crescente uso da internet, diversas áreas têm sido impactadas pela revolução tecnológica, incluindo a logística e o setor de entregas. Atualmente, é cada vez mais comum a terceirização dos serviços de entrega por meio de parcerias com entregadores independentes, proporcionando maior agilidade e flexibilidade no processo logístico. Tendo em vista estas considerações, surge a necessidade de desenvolver um software que facilite e otimize as operações de um centro de distribuição, conectando-o de forma eficiente aos entregadores parceiros.

Neste sentido, os avanços tecnológicos trouxeram consigo mudanças significativas na forma como nos comunicamos, consumimos e realizamos transações comerciais. De acordo com Rodrigues, Bechara e Grubba (2020, p. 3) “O enfrentamento da questão da era digital, da digitalização, tecnologização ou informatização da sociedade e da vida humana revela-se bastante atual”, visto que a popularização da internet e o uso de dispositivos móveis impulsionaram o surgimento de novas formas de negócios e serviços, transformando a dinâmica do mercado.

Vistas tais questões, a relação da tecnologia com o cotidiano na “[...] era digital, ocasiona profundos impactos, transformando a sociedade como um todo, principalmente na forma como as pessoas interagem, estabelecem seus negócios” (RODRIGUES, BECHARA e GRUBBA, 2020, p. 6). A partir disso, constatamos que empresas como a *Amazon* e o *Mercado Livre* se destacaram ao adotar estratégias de terceirização de entregas, permitindo que entregadores parceiros realizem o transporte e a entrega das encomendas, enquanto a empresa mantém o controle e a gestão dos processos.

No entanto, percebe-se que muitas dessas soluções ainda não oferecem uma visibilidade completa ao cliente em relação ao status da entrega. A falta de rastreamento em um tempo de resposta de baixa latência e feedbacks precisos pode gerar desconforto e incertezas no cliente, prejudicando a experiência de compra. Portanto, é essencial desenvolver uma solução que não apenas facilite a conexão entre o centro de distribuição e os entregadores parceiros, mas também forneça aos clientes a possibilidade de acompanhar o progresso da entrega.

A partir dessa problemática, desenvolvemos o Delivery Center, uma aplicação *WEB*, em que o foco é além de concentrar a relação entre o centro de distribuição e o entregador

parceiro, também dará a visibilidade quase que em tempo real do status da entrega ao cliente, podendo ser acompanhada por meio do mapa. Considerando o exposto, justificamos que o nosso Software realiza a partir de um entregador parceiro, a conexão entre a empresa e o cliente de forma mais eficiente, proporcionando uma melhor experiência de usuário para esse tipo de serviço em específico.

Além disso, o Software contribui para o aprimoramento das pequenas empresas no seu processo de logística, podendo utilizar a conexão com os entregadores parceiros e por consequência tornando mais ágil e confiável aos clientes. Tendo como princípio ser um sistema de uso livre e visando ajudar a comunidade, assim, vale ressaltar, que a relação entre entregador parceiro e centro de distribuição é proposto no sistema apenas como vínculo de serviço, não contendo parte relacionada a vínculo empregatício ou pagamento de taxas de serviço, portanto, caberá a empresa a remuneração aos seus parceiros.

O sistema Delivery Center é uma aplicação *web*, que usa conceitos de computação de alto desempenho, contendo uma arquitetura baseada em camadas, *Front-end* e *API REST*, serviço de mensageria e programação orientada a objetos e eventos. Nesse sentido, utilizamos para o desenvolvimento a aplicação *VUE.JS* para o *Front-end*, *PHP* com *Laravel* para a *API*, assim como *MySQL* para persistência dos dados, *REDIS* e *Socket.io* para o serviço de *Broadcasting* e mensageria.

1.1 OBJETIVOS

1.1.1 Objetivo Geral

Disponibilizar a aplicação *web* Delivery Center, para viabilizar o uso de entrega entre o centro de distribuição e o entregador parceiro em um tempo de resposta de baixa latência pelo usuário.

1.1.2 Objetivos Específicos

Para alcançar o objetivo geral deste trabalho de conclusão de curso, os seguintes objetivos específicos serão considerados:

- Disponibilizar um painel administrativo que possibilite o cadastro de usuários, a gestão de centros de distribuição, entrada de notas fiscais e relatórios no *dashboard*.
- Permitir que os entregadores parceiros gerem lotes de entrega a partir das notas fiscais cadastradas.

- Disponibilizar o sistema de rastreamento em um tempo de resposta de baixa latência, no qual os entregadores possam atualizar sua localização e os clientes possam acompanhar o progresso da entrega por meio de um mapa interativo.
- Disponibilizar uma interface amigável e intuitiva, proporcionando uma melhor experiência de uso tanto para os administradores do sistema quanto para os clientes.
- Contribuir para a melhoria dos processos de entrega, agilizando as operações e proporcionando maior transparência e confiabilidade aos clientes.

2 REVISÃO BIBLIOGRÁFICA

Nesta seção, descrevemos os sistemas de entregas presentes atualmente na sociedade, focando nos mais comuns. Além disso, destacamos os principais centros de distribuição e quais tecnologias utilizamos no desenvolvimento do *software* presente neste projeto.

2.1 Sistema de Entregas sob demanda

Os aplicativos para celulares, *tablets* e computadores facilitaram a vida moderna do ser humano. Segundo o jornalista France Junior, da Rádio Jornal da USP (2021), a entrega se tornou uma tendência na contemporaneidade devido à produção e o consumo da sociedade, o que significa que “[...] na esteira desse avanço, surge o *delivery* que, com a pandemia e o isolamento social, ganhou protagonismo no abastecimento de alimentos” (JUNIOR, 2021, [n.p.]). Com o advento, as pessoas têm mais tempo para pedir *delivery* sem a dificuldade de ter que se deslocar de onde estão para buscar uma entrega, seja de comida ou produtos diversos, como itens farmacêuticos, produtos de pet shop e itens de supermercado. Além disso, alguns sistemas de entrega se destacam atualmente no Brasil. De acordo com o site *TechTudo* (2019), algumas empresas se destacam no ramo de entregas e são elas *Ifood* e *Rappi*, que fazem *delivery* não apenas de comida, mas também de outros objetos como citados anteriormente.

3.1.1 Ifood e Rappi

- ***iFood***: Empresa brasileira especializada em *delivery*, o *iFood* é também uma empresa de tecnologia que reúne a praticidade da entrega de mercadorias entre os restaurantes, mercados, farmácias e consumidores. Hoje, conta com cerca de 300 mil estabelecimentos cadastrados, segundo o site da empresa (IFOOD, 2023).
- ***Rappi***: O *Rappi*, assim como o *iFood*, é uma empresa especializada em *delivery*, tendo sua diferenciação em entregar “qualquer produto”. De acordo com informações presentes no site da empresa, o *Rappi* nasceu em 2017 na Colômbia, após o seu fundador Simon Borrero ao observar que o mercado precisaria da capacidade de se reinventar a todo momento (RAPPI, 2018).

2.2 Centros de Distribuição

Os Centros de Distribuição são espaços físicos onde abrigam as mercadorias que saem das fábricas até serem retiradas para a entrega em lojas ou diretamente aos clientes. Sendo assim, é um ponto estratégico que garante agilidade nas entregas.

Segundo o site *HiveCloud*, esses locais são grandes galpões com terreno extenso para armazenar o maior número possível de mercadorias. Esses “CDs” podem pertencer tanto a transportadoras, quanto à empresas como Mercado Livre e *Amazon* (BRASIL, 2023).

2.2.1 Mercado Livre e Amazon

- **Mercado Livre:** Nasceu em 1999, na Argentina, como um *marketplace*. Segundo o site da empresa, atualmente o Mercado Livre é a maior empresa de tecnologia da América Latina, reunindo vários vendedores e ofertas de produtos, impactando o Produto Interno Bruto – o PIB – brasileiro em 0,12 (MERCADOLIVRE, [s.d.]).
- **Amazon:** A *Amazon* foi criada em Seattle, nos Estados Unidos em 5 de julho de 1994, pelo empresário estadunidense Jeff Besos. A empresa, inicialmente, era uma pequena loja digital de livros que se expandiu e, atualmente, é considerada uma das maiores empresas de tecnologia segundo o *Canal Tech*. A Amazon comercializa produtos de diversos, como brinquedos, alimento, papelaria, livros, bolsas, dentre outros. Além disso, oferece também serviços de *streaming* através das plataformas *Twitch*, *Prime Video* e *Amazon Music* (CANALTECH, 2023).

2.3 Tecnologias Utilizadas

O conceito aplicado para o funcionamento do projeto foi o de *cloud computing*, sendo este “um conjunto de recursos como capacidade de processamento, armazenamento, conectividade, plataformas, aplicações e serviços disponibilizados na Internet” (TAURION, 2009, p.2).

As tecnologias utilizadas foram as mais compatíveis para o desenvolvimento do projeto de Software de Entregas. Entre elas estão Laravel, Redis, Socket.IO, HTML, CSS, JavaScript, Vue, js e MySQL. Portanto, essas tecnologias estarão presentes ao longo do presente documento, contextualizadas de acordo com sua utilização.

Nesta seção, apresentamos como cada tecnologia funciona.

2.3.1 PHP e Laravel

O PHP é uma abreviatura em inglês de *Hypertext Preprocessor*, segundo a documentação oficial *PHP* (2001, [n.p.]), é “uma linguagem de script open source de uso geral, muito utilizada, e especialmente adequada para o desenvolvimento web e que pode ser embutida dentro do *HTML*”.

Sobre o Laravel, de acordo com a sua documentação, podemos afirmar que “é um *framework* de aplicações web com sintaxe expressiva e elegante. Uma estrutura da Web fornece uma estrutura e um ponto de partida para a criação de seu aplicativo” (LARAVEL, 2011).

2.3.2 Redis

O Redis (*Remote Dictionary Server*) é um armazenamento de estrutura de dados na memória de código aberto, podendo ser utilizado como banco de dados, cache, agente de mensagens e mecanismo de *streaming*. Para Lazoti, o *Redis* é importante para escrever e ler dados com rapidez, pois “[...] é um servidor TCP que faz uso do modelo cliente-servidor” (LAZOTI, 2014, p. 8).

Visando o melhor desempenho, o *Redis* trabalha com um conjunto de dados na memória. Dependendo do seu caso de uso, o *Redis* pode manter seus dados despejando periodicamente o conjunto de dados no disco ou anexando cada comando a um log baseado em disco. É possível desabilitar a persistência se precisar apenas de um cache na memória rico em recursos e em rede.

Ademais, o *Redis* oferece suporte à replicação assíncrona, com sincronização rápida sem bloqueio e reconexão automática com ressincronização parcial na divisão da rede.

Além das funções mencionadas, também inclui as funcionalidades de Transações, Pub/Sub, Script Lua, Chaves com tempo de vida limitado, LRU e Failover automático.

2.3.3 Socket.IO

O *Socket.IO* é uma biblioteca *JavaScript* que permite a comunicação bidirecional em tempo real entre o servidor e os clientes. Ele fornece uma abstração sobre os diferentes mecanismos de comunicação, como o *WebSocket*, e oferece suporte a *fallbacks*, como *long polling*, para garantir a compatibilidade com navegadores mais antigos.

Socket.IO é uma biblioteca JavaScript para aplicativos web em com tempo de resposta de baixa latência. Ele tem duas partes principais: uma biblioteca do lado do cliente que é executada no navegador e uma biblioteca do lado do servidor para Node.js. Ambas têm uma API quase idêntica. Como o próprio nome sugere, o *Socket.IO* se baseia no modelo de evento de socket, permitindo que os desenvolvedores emitam eventos e respondam a eles (SOCKET.IO, 2021, [n.p.]).

2.3.4 HTML e CSS

O *HTML* (*HyperText Markup Language*) é uma linguagem de marcação utilizada para estruturar o conteúdo de uma página da web. Ela define a forma como os elementos e componentes de uma página são organizados, como cabeçalhos, parágrafos, imagens, links e outros elementos. Assim como, é uma base fundamental para a criação e estruturação de páginas web.

De acordo com Silva, (2014, p. 15): “*HTML* é a linguagem que descreve a estrutura e o significado do conteúdo de uma página web. *CSS* é a linguagem que descreve a apresentação, o layout e a aparência visual do conteúdo”.

Ademais, *CSS* (*Cascading Style Sheets*) é uma linguagem de estilo utilizada para controlar a apresentação e o design de uma página web. Com o *CSS*, é possível definir cores, fontes, tamanhos, margens, espaçamentos e outros aspectos visuais de uma página. Ele permite separar a estrutura do conteúdo (definido pelo *HTML*) da sua apresentação visual.

2.3.5 JavaScript

JavaScript é uma linguagem de programação versátil e amplamente utilizada para desenvolvimento web, que permite aos desenvolvedores criarem interatividade e dinamismo em sites e aplicativos. Com uma sintaxe relativamente simples, é possível manipular elementos da página, realizar cálculos, validar formulários e muito mais. Além disso, também é possível utilizá-lo em ambientes de servidor por meio do Node.js, expandindo suas capacidades para além do navegador.

[...] JavaScript é uma linguagem de alto nível, dinâmica, interpretada e não tipada, conveniente para estilos de programação orientados a objetos e funcionais. A sintaxe de JavaScript é derivada da linguagem Java, das funções de primeira classe

de Scheme e da herança baseada em protótipos de Self. Mas não é preciso conhecer essas linguagens nem estar familiarizado com esses termos para utilizar este livro e aprender JavaScript (FLANAGAN, 2012, p. 19).

2.3.6 Vue.js

Vue.js é um framework JavaScript de código aberto que permite o desenvolvimento de interfaces de usuário interativas e responsivas. Com sua abordagem baseada em componentes, o Vue.js facilita a criação de aplicativos web modernos e escaláveis, oferecendo uma experiência de desenvolvimento flexível e eficiente.

Um framework progressivo para a construção de interfaces de usuário. Ao contrário de outros frameworks monolíticos, o Vue foi projetado desde sua concepção para ser adotável incrementalmente. A biblioteca principal é focada exclusivamente na camada visual (view layer), sendo fácil adotar e integrar com outras bibliotecas ou projetos existentes. Por outro lado, Vue também é perfeitamente capaz de dar poder a sofisticadas *Single-Page Applications* quando usado em conjunto com ferramentas modernas e bibliotecas de apoio.(VUEJS, 2023, [s.p.]).

2.3.7 MySQL

De acordo com a documentação oficial do MySQL (MYSQL, 2011) trata-se de um sistema de gerenciamento de banco de dados relacional amplamente utilizado, reconhecido por sua confiabilidade e desempenho. O MySQL permite o armazenamento e recuperação eficiente de grandes volumes de dados, além de oferecer recursos avançados, tais como consultas complexas, replicação e segurança robusta. Sua utilização é difundida em uma variedade de aplicações, desde pequenos websites até sistemas empresariais de alta escala.

3 METODOLOGIA

Nesta seção, descrevemos a metodologia utilizada no desenvolvimento do software de entregas para o centro de distribuição, focando no levantamento de requisitos e na concepção da solução baseada em modelos de negócio como os do Mercado Livre e Amazon, que envolvem a terceirização da entrega por meio de entregadores parceiros. A partir dessa ideia, destacamos o diferencial da nossa solução, que permite que o entregador parceiro realize o lote de entrega na plataforma, enquanto o destinatário acompanha o status da entrega por meio de eventos de atualização em um mapa com tempo de resposta de baixa latência.

3.1 Levantamento de Requisitos

Durante o levantamento de requisitos, foram analisados modelos de negócio adotados por empresas líderes no setor de entregas, como o Mercado Livre e a Amazon. Observamos a estratégia de terceirização da entrega por meio de entregadores parceiros, o que se mostrou uma prática eficiente e utilizável. Com base nessa abordagem, estabelecemos a regra de negócio para o nosso software de entregas.

A principal ideia é que o centro de distribuição seja responsável por gerenciar as notas fiscais e a geração dos lotes de entrega. Os entregadores parceiros se cadastram na plataforma e têm a possibilidade de visualizar os lotes disponíveis para entrega. Eles podem selecionar os lotes desejados e se comprometer a realizá-los dentro de um determinado prazo.

3.1.1 Requisitos Funcionais do Sistema

- **Autenticação de Usuários:** O sistema permite que administradores e entregadores façam login no painel administrativo por meio de autenticação com e-mail e senha.
- **Página de Rastreamento:** O sistema fornece uma página de rastreamento acessível ao público, onde os destinatários possam inserir o código de rastreamento ou os números da nota fiscal e do documento identificador do destinatário para acompanhar o status da entrega.

- **Busca de Notas Fiscais:** Os entregadores podem realizar consultas de notas fiscais disponíveis para entrega, filtrados pelo centro de distribuição com opções de busca por número da nota fiscal, emissor ou outras informações relevantes.
- **Geração de Lotes de Entrega:** Os entregadores podem gerar lotes de entrega a partir das notas fiscais disponíveis, selecionando as notas fiscais desejadas para compor cada lote.
- **Rastreamento em um tempo de resposta de baixa latência:** O sistema permite que os entregadores enviem suas coordenadas para a API, que será responsável por atualizar a localização no mapa em na página de rastreamento.
- **Finalização da Entrega:** Os entregadores podem finalizar a entrega, anexando um comprovante de que a mesma foi efetuada com sucesso, além de fornecer uma observação opcional, se necessário.
- **Cadastro de Usuários:** O administrador pode cadastrar novos usuários no sistema, com diferentes perfis (administrador ou entregador), e vincular um veículo ao usuário entregador, se aplicável.
- **Cadastro de Centros de Distribuição:** O administrador pode cadastrar, alterar e excluir centros de distribuição, fornecendo informações como nome, endereço e outros detalhes relevantes.
- **Entrada de Notas Fiscais:** O administrador pode realizar a entrada de notas fiscais no sistema, importando arquivos XML ou inserindo manualmente as informações necessárias. Nesse processo também é gerado o código de rastreamento de cada nota fiscal, além de registrar o evento de entrada.

3.1.2 Requisitos Não Funcionais do Sistema

- **Segurança:** O sistema garante a segurança dos dados, estando de acordo com a LGPD, protegendo as informações sensíveis dos usuários, como nome, email, senha e endereço. O acesso à API deve ser protegido por um token *JWT* para garantir a autenticação e autorização adequadas.

- **Performance:** O sistema é capaz de lidar com um alto volume de requisições, garantindo tempos de resposta rápidos e desempenho eficiente durante o rastreamento com um tempo de resposta de baixa latência.
- **Usabilidade:** O sistema é intuitivo e de fácil utilização, com interfaces claras e organizadas, permitindo que os usuários realizem suas tarefas de forma rápida e sem complicações.
- **Feedbacks ao Usuário:** O sistema fornece feedbacks claros e informativos ao usuário em ações de sucesso ou erro, garantindo uma experiência de uso intuitiva.
- **Confiabilidade:** O sistema é confiável, garantindo o registro correto das informações de rastreamento e a atualização adequada do status da entrega.
- **Integração:** O sistema é capaz de integrar-se com serviços de mensageria, como REDIS e socket.io, para fornecer atualizações de localização com um tempo de resposta de baixa latência.
- **Escalabilidade:** O sistema é projetado para ser escalável, permitindo a adição de mais usuários, centros de distribuição e notas fiscais à medida que a empresa cresce.
- **Compatibilidade:** O sistema é compatível com diferentes navegadores da web e dispositivos, garantindo uma experiência consistente para os usuários.
- **Manutenibilidade:** O sistema é facilmente mantido, permitindo atualizações, correções de bugs e melhorias no futuro, sem interromper o fluxo de trabalho dos usuários.

3.2 Projeto da Solução

Com base nas informações coletadas durante o levantamento de requisitos, projetamos uma solução que permita a interação entre o centro de distribuição, os entregadores parceiros e os destinatários. A plataforma tem duas visões principais: o painel administrativo, acessado pelo centro de distribuição, e a página de rastreamento de entregas, acessada pelos destinatários.

No painel administrativo, o centro de distribuição pode cadastrar novos usuários, incluindo administradores e entregadores parceiros. Além disso, é possível cadastrar o

centro de distribuição em si e importar as notas fiscais por meio de arquivos XML. Essas notas fiscais serão utilizadas para a geração dos lotes de entrega. As figuras a seguir ilustram alguns dos casos de uso projetados.

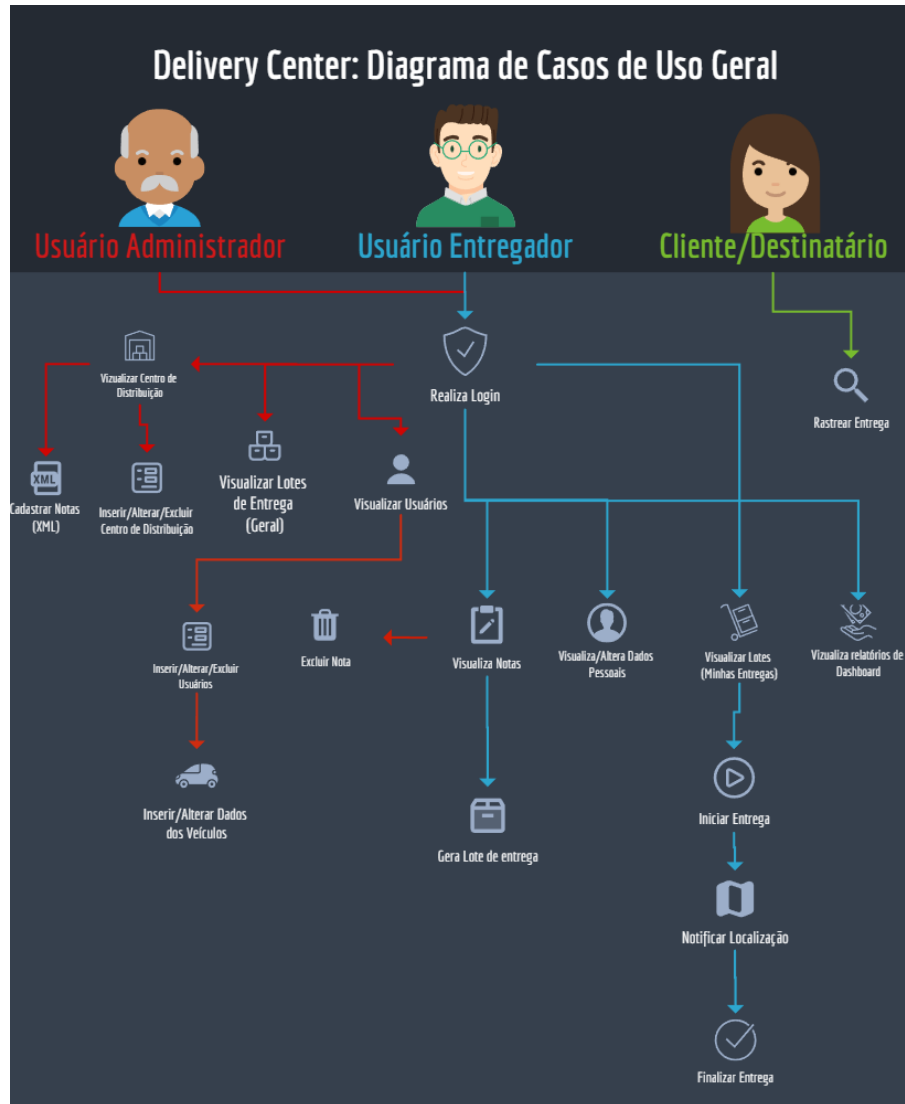


Figura 1 – Diagrama de Casos de Uso Geral

Fonte: Elaborado pelo autor, 2023.

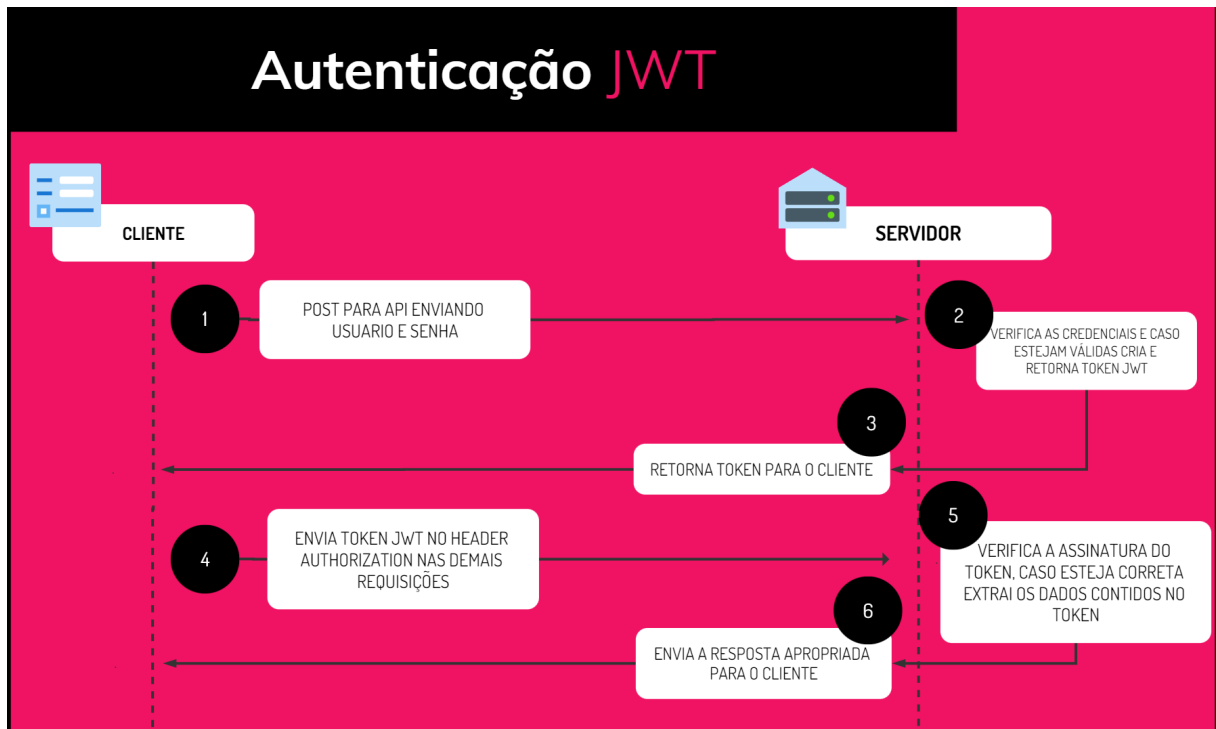


Figura 2 – Fluxo de autenticação e requisições usando JWT

Fonte: Elaborado pelo autor, 2023.

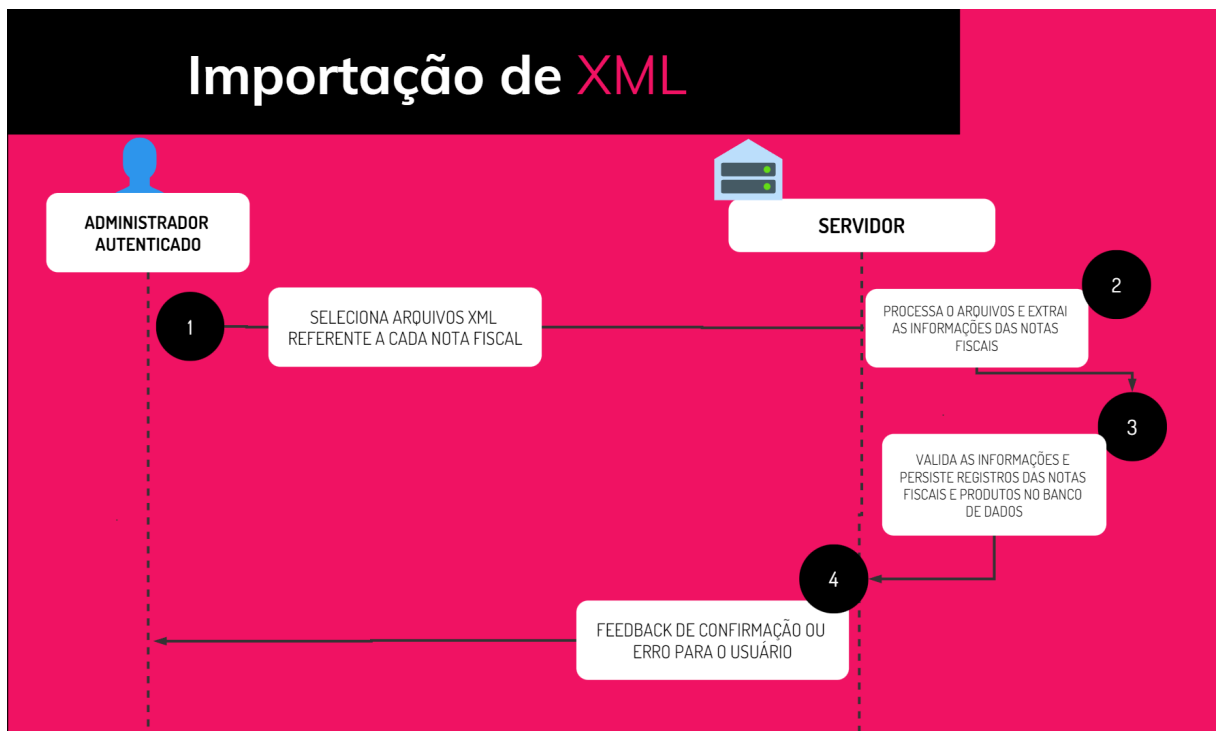


Figura 3 – Cadastro de Notas Fiscais

Fonte: Elaborado pelo autor, 2023.

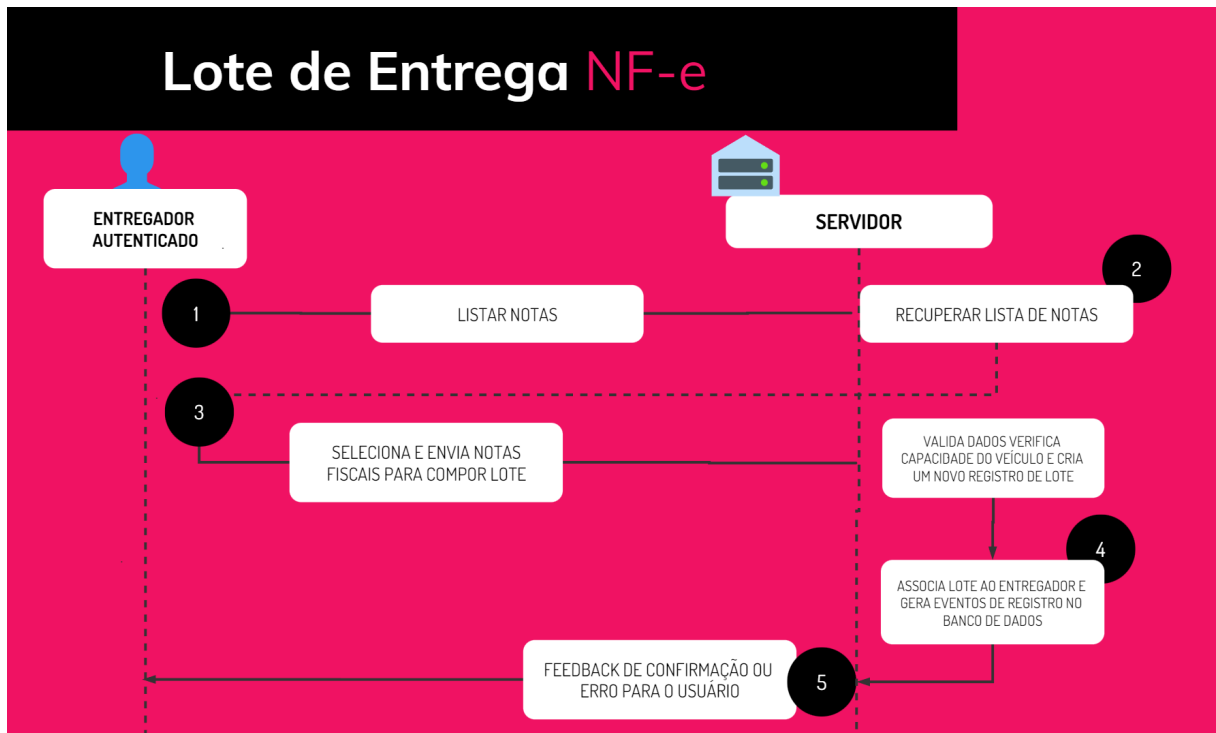


Figura 4 – Geração de Lote de Entrega

Fonte: Elaborado pelo autor, 2023.

3.3 Implementação

Na implementação do software de entregas, utilizamos o *framework* Laravel (OTWELL e BORGES, 2013; RESS, DALMOLIN e LAGUNA, 2014). para o desenvolvimento do *backend*. Essa escolha se baseia na robustez, segurança e flexibilidade oferecidas pelo Laravel, permitindo uma rápida implementação das funcionalidades requeridas.

Para o serviço de mensageria, utilizamos o Redis em conjunto com o Socket.IO. Essa combinação permite o envio de atualizações em um tempo de resposta de baixa latência para os destinatários, refletindo as mudanças no mapa de rastreamento conforme o entregador parceiro atualiza sua localização.

Portanto, o *frontend* foi desenvolvido utilizando HTML, CSS e JavaScript, com o *framework* Vue.js para agilizar o desenvolvimento da interface de usuário. Essa escolha nos permitiu criar uma experiência de usuário intuitiva e responsiva, fornecendo acesso fácil às informações de rastreamento e atualizações com um tempo de resposta de baixa latência.

3.4 Testes e Validação

Durante a etapa de testes e validação, realizamos testes unitários, testes de integração e testes de aceitação para garantir a qualidade e o correto funcionamento do software de entregas. Os testes serão executados em diferentes cenários, como a criação de usuários, o cadastro de notas fiscais, a geração de lotes de entrega e o rastreamento.

Além disso, realizamos validações junto aos usuários finais, incluindo destinatários e entregadores parceiros, para obter *feedback* e realizar eventuais ajustes na solução, garantindo sua usabilidade e eficácia.

Com essa metodologia, estamos confiantes de que desenvolvemos um software de entregas eficiente e inovador, permitindo que o entregador parceiro realize o lote de entrega em nossa plataforma e que o destinatário acompanhe o status da entrega através de eventos de atualização no mapa.

4 DESENVOLVIMENTO DO SOFTWARE

Nesta seção, detalhamos o desenvolvimento do software de entregas para o centro de distribuição, abordando a arquitetura do sistema, as tecnologias utilizadas e as principais etapas de implementação.

4.1 Arquitetura do Sistema

A arquitetura do sistema foi planejada para a construção de uma aplicação com um tempo de resposta de baixa latência, levando em consideração os requisitos e a escalabilidade da solução. Optamos por uma arquitetura em camadas, em que cada camada tem uma responsabilidade específica. A seguir, descrevemos as principais camadas do sistema:

Camada de apresentação: Essa camada é responsável pela interface de usuário e interação com o sistema. Utilizamos tecnologias web, como HTML, CSS e JavaScript, juntamente com o framework Vue.js, para criar uma interface intuitiva e responsiva.

Camada de aplicação: Aqui, estão concentradas as regras de negócio do sistema. Utilizamos o framework Laravel (OTWELL e BORGES, 2013; RESS, DALMOLIN e LAGUNA, 2014) no backend para implementar essa camada, permitindo uma fácil criação e gerenciamento das funcionalidades, como cadastro de usuários, notas fiscais e geração de lotes de entrega.

Camada de persistência: Utilizamos o banco de dados relacional MySQL para armazenar e recuperar os dados do sistema. O MySQL oferece uma estrutura robusta e confiável para garantir a integridade dos dados e o desempenho adequado.

Camada de serviços externos: Nesta camada, integramos serviços externos, como o Redis e o Socket.IO ao nosso backend, para fornecer o serviço de *broadcasting*. O Redis é utilizado para receber e publicar as atualizações de localização do entregador parceiro, enquanto o Socket.IO é responsável por transmitir essas atualizações para os destinatários com um tempo de resposta de baixa latência.

Camada de segurança: Essa camada é responsável por garantir a proteção dos dados sensíveis e controlar o acesso aos recursos do sistema. Para isso, utilizamos o padrão de autenticação e autorização com token *JWT*, que proporciona uma forma segura de autenticar os usuários e controlar suas permissões de acesso. Ao fazer login no sistema, os usuários receberão um token, que será enviado em todas as solicitações subsequentes para

autenticação e autorização. Além disso, serão aplicadas boas práticas de criptografia para proteger as informações armazenadas no banco de dados, bem como técnicas de prevenção contra ataques comuns, como injeção de código e cross-site scripting (*XSS*). A camada de segurança foi implementada complementando outras funcionalidades do sistema, como a de autenticação, controle de acesso e gerenciamento de sessões, para garantir a integridade e a confidencialidade dos dados, bem como a proteção contra acesso não autorizado.

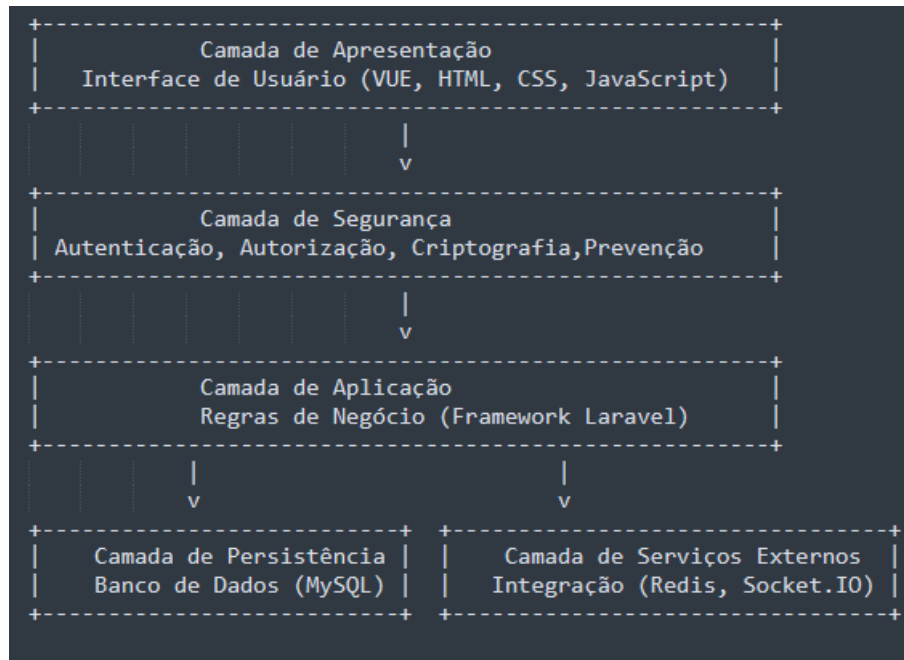


Figura 5 – Representação das camadas do sistema

Fonte: Elaborado pelo autor, 2023.

4.2 Backend com Laravel

O backend do software de entregas foi desenvolvido utilizando o framework Laravel (OTWELL e BORGES, 2013; RESS, DALMOLIN e LAGUNA, 2014). O Laravel oferece uma estrutura sólida e eficiente para a criação de APIs e o gerenciamento das regras de negócio. Nesse sentido, utilizamos os recursos do Laravel, como roteamento, controle de acesso, validação de dados e interação com o banco de dados MySQL, para implementar as funcionalidades do sistema.

4.3 Serviço de *Broadcasting* com *Redis* e *Socket.IO*

Um serviço de *broadcasting* em aplicações com um tempo de resposta de baixa latência refere-se à capacidade de enviar mensagens ou dados para múltiplos destinatários simulta-

neamente. Em vez de enviar mensagens individualmente para cada destinatário, o serviço de *broadcasting* permite que uma única mensagem seja enviada e recebida por vários clientes ou dispositivos conectados.

O *WebSocket* é um protocolo de comunicação bidirecional que permite a troca de dados em tempo real entre um servidor e um cliente. Ele estabelece uma conexão persistente entre o servidor e o cliente, permitindo a comunicação em tempo real e a troca de informações em ambas as direções.

Quando combinados, o serviço de *broadcasting* e o *WebSocket* permitem que aplicações de tempo real transmitam informações para vários clientes de forma eficiente. Em vez de os clientes solicitarem repetidamente atualizações do servidor, o servidor pode enviar automaticamente as atualizações para todos os clientes conectados por meio da conexão *WebSocket*, garantindo uma comunicação rápida e eficiente. Sendo assim, esses recursos são frequentemente utilizados em aplicações que exigem atualizações em tempo real, como aplicativos de bate-papo, feeds de notícias, painéis de monitoramento, jogos online e colaboração simultânea.

Para permitir o rastreamento das entregas, integramos o *Redis* e o *Socket.IO* ao nosso sistema. O *Redis* é utilizado como nosso driver de transmissão, onde o nosso sistema transmitirá o evento no serviço de broadcast *Pub/Sub* para publicar e receber as atualizações de localização dos entregadores parceiros.

O *Socket.IO* é responsável por estabelecer uma conexão bidirecional entre o servidor e os clientes (destinatários). Ele envia as atualizações de localização do *Redis* para os clientes conectados, permitindo que o mapa de rastreamento seja atualizado automaticamente à medida que o entregador se move.

Como servidor *Socket.IO*, utilizamos o *laravel-echo-server*. É um servidor baseado em *Node.js* que pode se conectar ao *Redis* e pode estabelecer conexões com o *WebSocket* e com os cliente. Ele pegará o evento de transmissão e o enviará para clientes conectados. Além disso, é capaz de receber eventos diretamente de qualquer cliente conectado e transmiti-los para outros clientes conectados.

No lado cliente, usamos *socket.io-client* e *laravel-echo*. O primeiro dos dois é o importante, que se conectará ao nosso servidor *Socket.IO*. O segundo é uma biblioteca auxiliar que facilita a audição dos eventos.

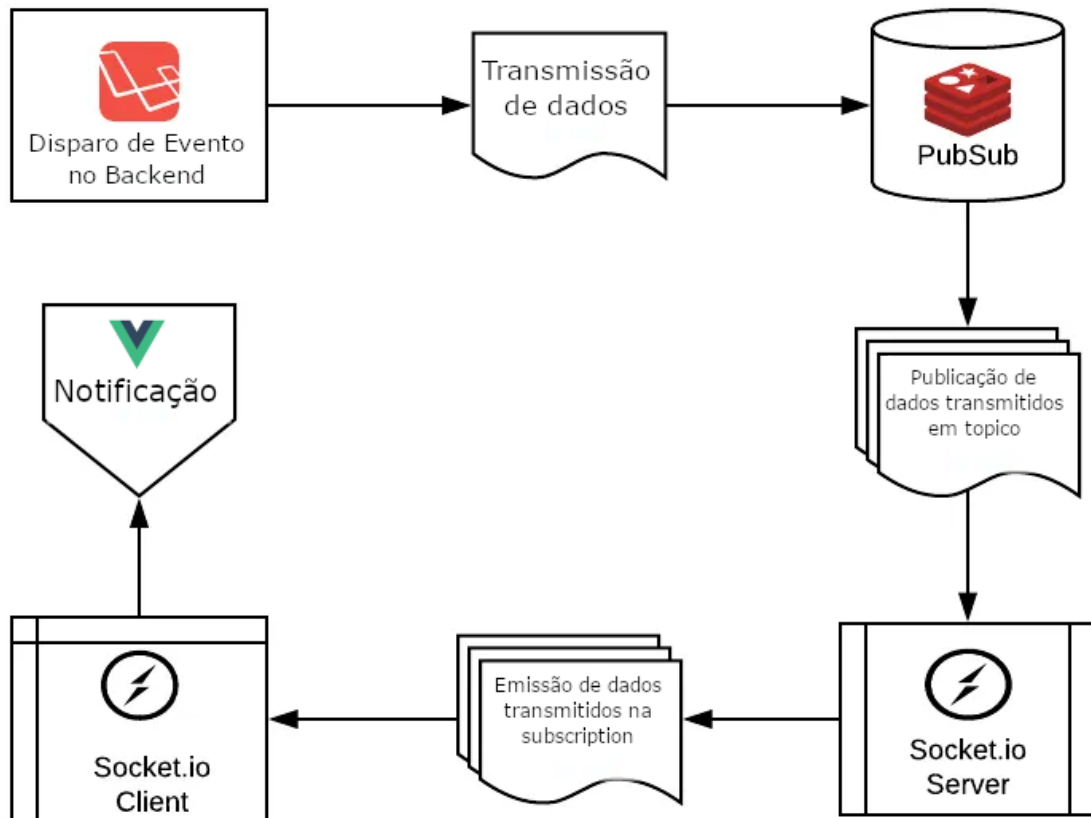


Figura 6 – Notificações Pub/Sub com *Redis* e *Socket.IO*

Fonte: Elaborado pelo autor, 2023.

4.4 Frontend com HTML, CSS, JavaScript e Vue.js

No *frontend*, utilizamos as tecnologias *WEB* padrão, como *HTML*, *CSS* e *JavaScript*, juntamente com o framework *Vue.js*. O *Vue.js* facilita o desenvolvimento de interfaces de usuário interativas e reativas, permitindo que os destinatários acompanhem o status da entrega com um tempo de resposta de baixa latência.

Criamos uma página de rastreamento de entregas intuitiva, em que os destinatários podem visualizar o mapa com a localização atual do entregador parceiro. As atualizações de localização são exibidas instantaneamente, fornecendo uma experiência de rastreamento em um tempo de resposta de baixa latência e aumentando a transparência e confiabilidade do processo de entrega.

4.5 Banco de Dados Relacional MySQL

Utilizamos o banco de dados relacional *MySQL* para armazenar os dados do sistema, como informações de usuários, centro de distribuição, notas fiscais e lotes de entrega. O *MySQL* oferece uma estrutura flexível e confiável, permitindo a organização eficiente dos dados e a recuperação rápida das informações necessárias para o funcionamento adequado do software de entregas.

Ao projetar o banco de dados, levamos em consideração a integridade dos dados e a eficiência das consultas, garantindo uma base sólida para o funcionamento do sistema.

Portanto, com essa abordagem de desenvolvimento, garantimos a implementação de um software de entregas robusto, eficiente e com recursos de rastreamento em um tempo de resposta de baixa latência, oferecendo uma experiência avançada tanto para o centro de distribuição quanto para os destinatários.

5 FUNCIONALIDADES DO SISTEMA

O sistema de entregas do centro de distribuição foi desenvolvido para oferecer uma experiência completa tanto para os entregadores quanto para os administradores. Possui um conjunto abrangente de funcionalidades que abordamos nesta seção, destacando a tela de login, a página de rastreamento e as opções disponíveis no painel administrativo.

5.1 Tela de Login e Página de Rastreamento

O sistema inicia com uma tela de *login*, em que os usuários, sejam eles administradores ou entregadores, devem inserir seu e-mail e senha para acessar o painel administrativo. Além disso, a tela de *login* também oferece a opção de ir diretamente para a página de rastreamento, caso o usuário queira apenas acompanhar o status de uma entrega sem acessar a área administrativa.

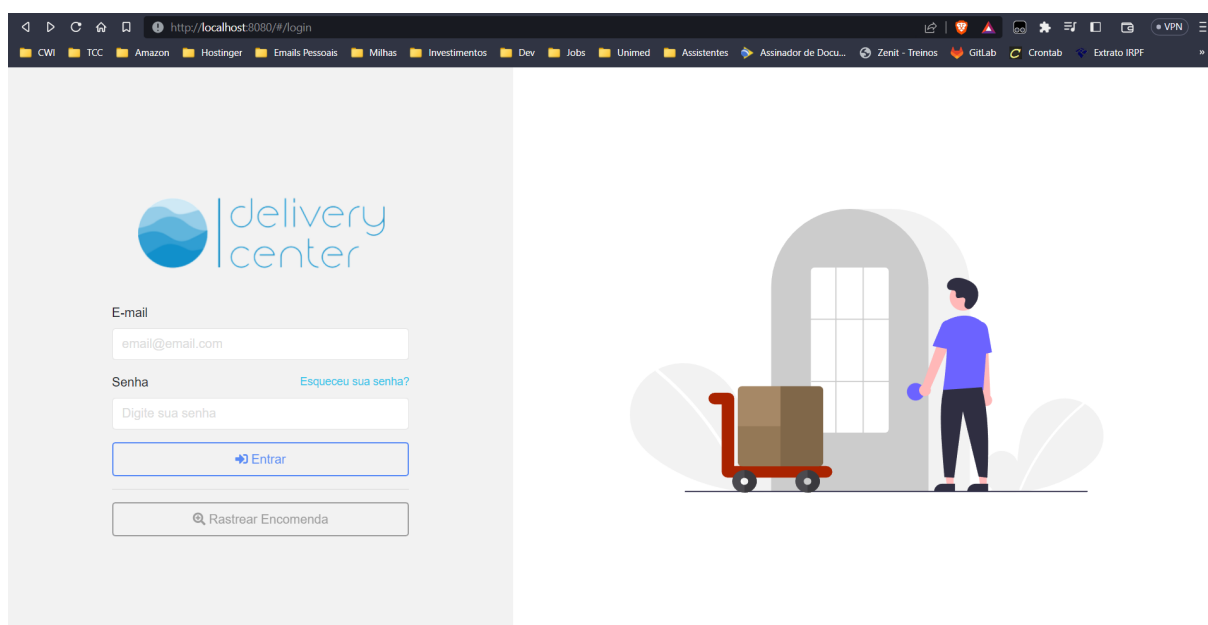


Figura 7 – Tela de Login

Fonte: Elaborado pelo autor, 2023.

A página de rastreamento permite que os destinatários verifiquem o status de suas entregas de forma simples e conveniente. Um formulário de busca oferece opções para pesquisar pelo código de rastreio ou pelos números da nota fiscal e do documento identificador do destinatário. Ao realizar a busca, o sistema exibe um mapa com a localização atual do entregador, juntamente com os eventos registrados da entrega. Bem como, são

apresentados os dados do entregador, nota fiscal e endereço de entrega, fornecendo informações detalhadas sobre o andamento da entrega.

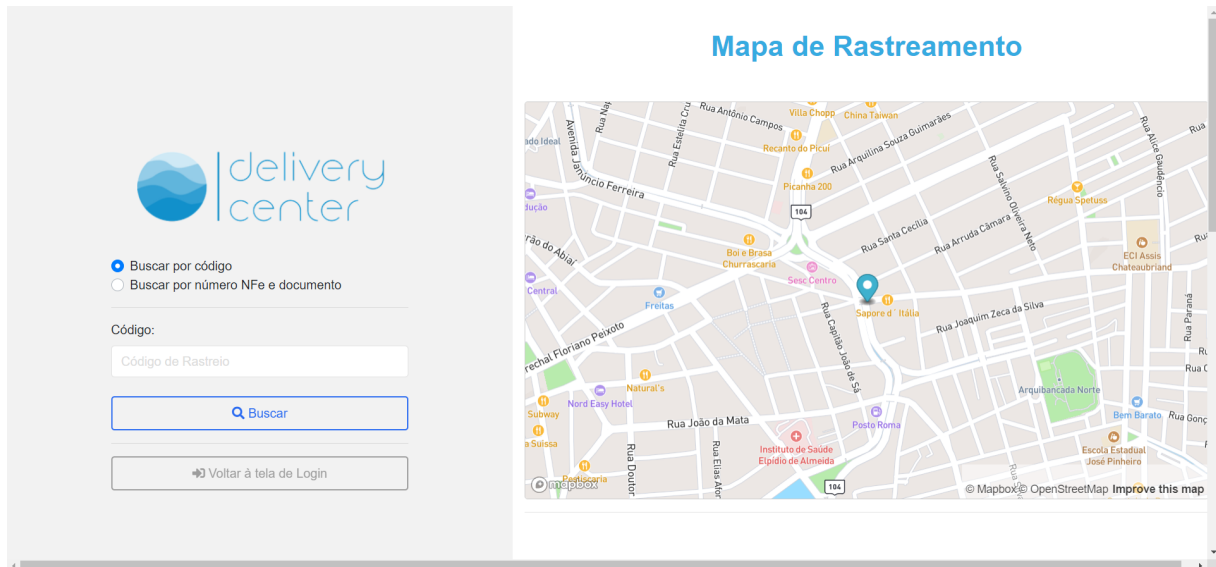


Figura 8 – Mapa de rastreamento com localização

Fonte: Elaborado pelo autor, 2023.

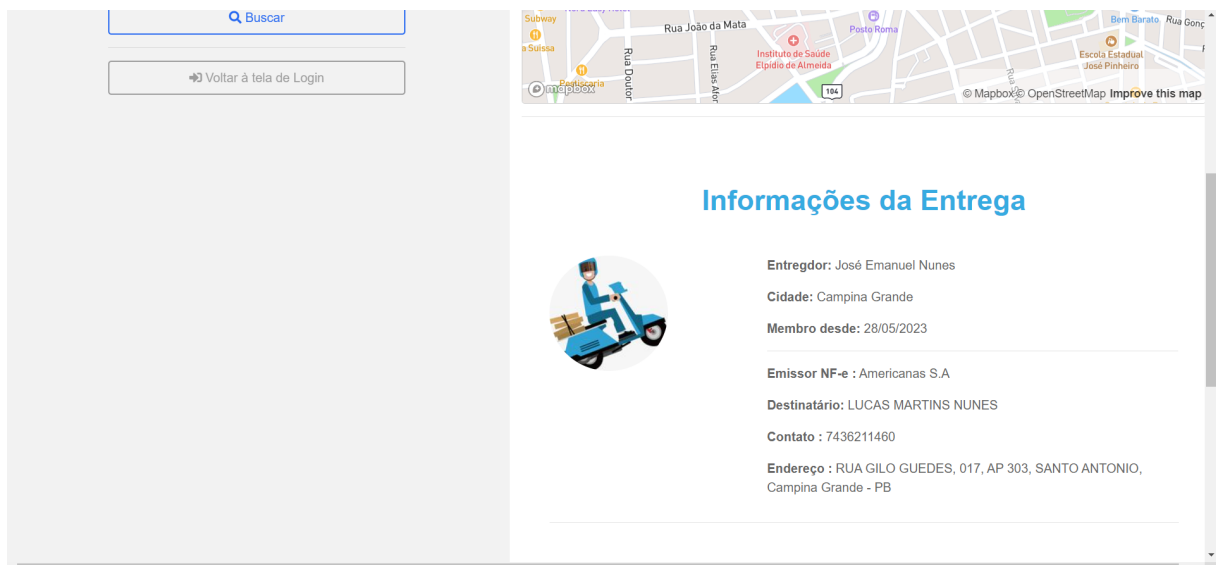
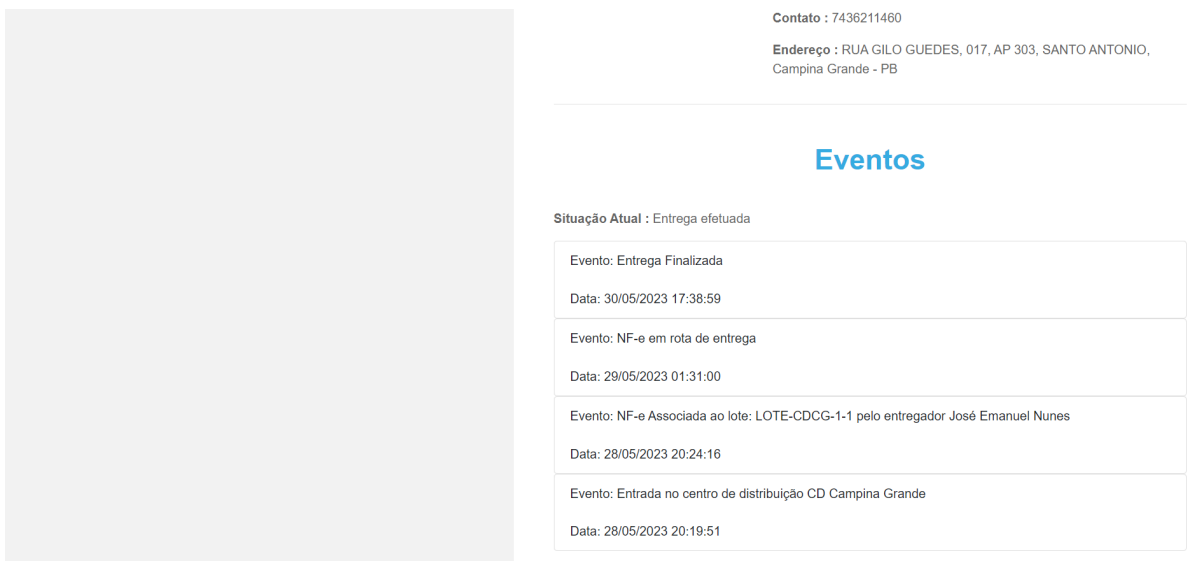


Figura 9 – Informações da entrega

Fonte: Elaborado pelo autor, 2023.



The screenshot displays a user interface for a delivery management system. On the right side, there is contact information: 'Contato : 7436211460' and 'Endereço : RUA GILO GUEDES, 017, AP 303, SANTO ANTONIO, Campina Grande - PB'. Below this, a section titled 'Eventos' shows the current status as 'Entrega efetuada'. A table lists four events with their respective dates and times.

Contato : 7436211460	
Endereço : RUA GILO GUEDES, 017, AP 303, SANTO ANTONIO, Campina Grande - PB	
Eventos	
Situação Atual : Entrega efetuada	
Evento: Entrega Finalizada	
Data: 30/05/2023 17:38:59	
Evento: NF-e em rota de entrega	
Data: 29/05/2023 01:31:00	
Evento: NF-e Associada ao lote: LOTE-CDCG-1-1 pelo entregador José Emanuel Nunes	
Data: 28/05/2023 20:24:16	
Evento: Entrada no centro de distribuição CD Campina Grande	
Data: 28/05/2023 20:19:51	

Figura 10 – Eventos da entrega

Fonte: Elaborado pelo autor, 2023.

5.2 Painel Administrativo

O painel administrativo oferece funcionalidades específicas para administradores e entregadores, permitindo um gerenciamento eficiente do sistema.

5.2.1 Funcionalidades para Entregadores

Dashboard: Os entregadores têm acesso a um painel de controle com gráficos e relatórios que fornecem informações relevantes sobre suas atividades de entrega. Isso permite que acompanhem o desempenho e tenham uma visão geral das entregas realizadas.

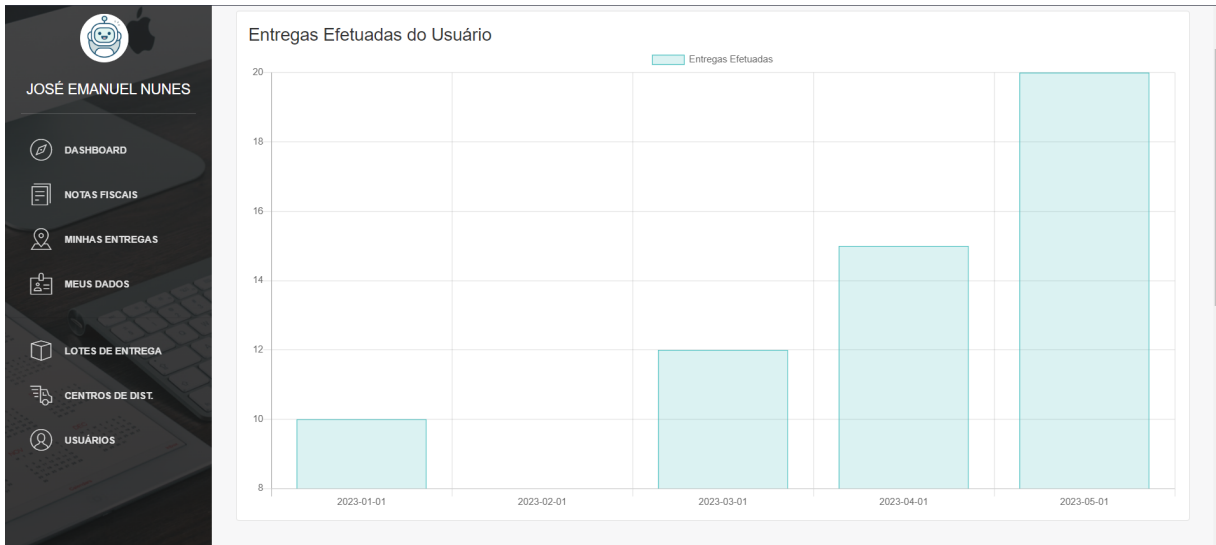


Figura 11 – Painel administrativo - Dashboard

Fonte: Elaborado pelo autor, 2023.

Consulta de Notas Fiscais: Essa funcionalidade permite aos entregadores consultar as notas fiscais disponíveis para entrega. Eles podem gerar lotes de entrega, selecionando as notas fiscais desejadas para compor cada lote, facilitando o planejamento e organização das entregas.

The screenshot shows the "Delivery Center Monitor" interface. The main section is titled "Notas Fiscais" and "Cadastros". It includes a dropdown menu for "CENTRO DE DISTRIBUIÇÃO" set to "CD Campina Grande". Below this is a search bar and a pagination control set to "10 por página". The main content is a table listing tax invoices.

#	NÚMERO	EMISSOR	DESTINATÁRIO	RASTREIO	EMISSÃO	ENTRADA	STATUS	AÇÕES
<input type="checkbox"/>	3628	MAGAZINE LUIZA S/A	LUCAS MARTINS NUNES	CDCG-1-3628	25/08/2022 23:42	27/06/2023 11:01	Aguardando Retirada	<input type="checkbox"/>
<input type="checkbox"/>	463540	MAGAZINE LUIZA S/A	LUCAS MARTINS NUNES	CDCG-2-463540	16/02/2023 13:21	27/06/2023 11:01	Aguardando Retirada	<input type="checkbox"/>
<input type="checkbox"/>	32178526	Americanas S.A	LUCAS MARTINS NUNES	CDCG-3-32178526	29/09/2022 11:43	27/06/2023 11:01	Aguardando Retirada	<input type="checkbox"/>

Figura 12 – Listar Notas Fiscais

Fonte: Elaborado pelo autor, 2023.

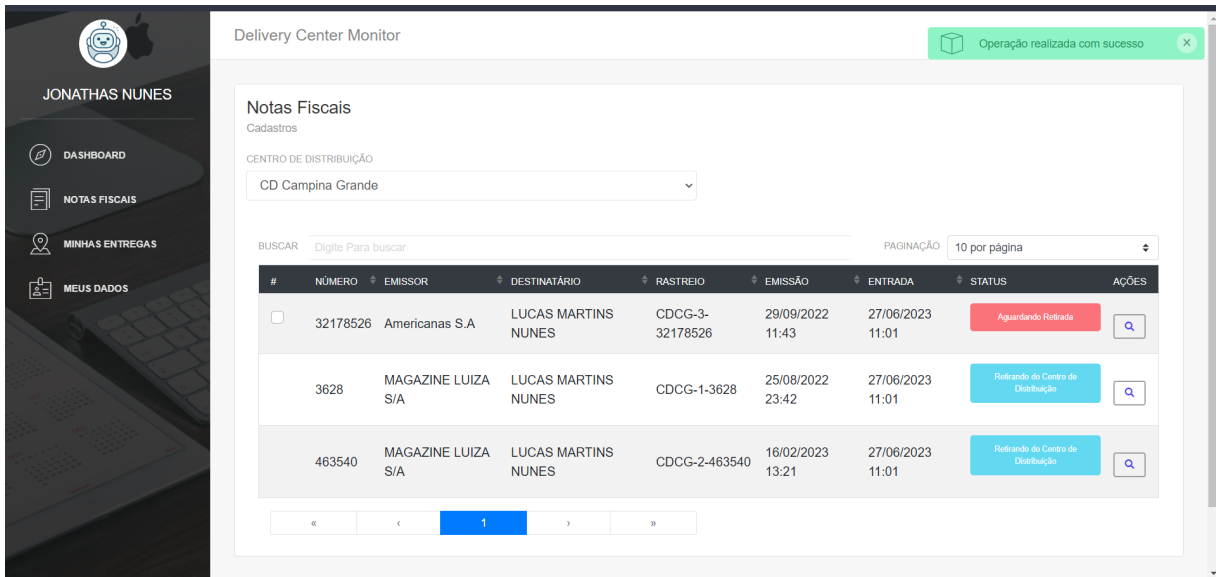


Figura 13 – Gerar Lote de entrega - Notas Fiscais

Fonte: Elaborado pelo autor, 2023.

Minhas Entregas: Nessa seção, os entregadores podem visualizar os lotes de entrega gerados. Eles têm a opção de dar início à entrega de cada nota fiscal, registrar a finalização da entrega anexando um comprovante e, se necessário, adicionar uma observação opcional para fornecer informações adicionais sobre a entrega.

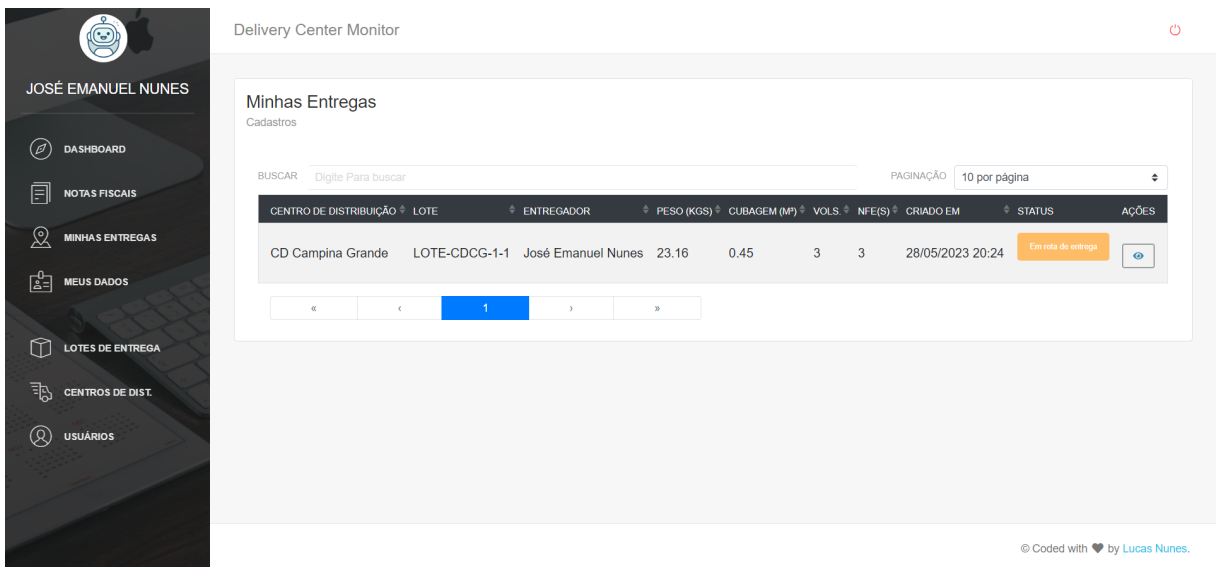


Figura 14 – Listar Entregas do usuário - Minhas entregas

Fonte: Elaborado pelo autor, 2023.

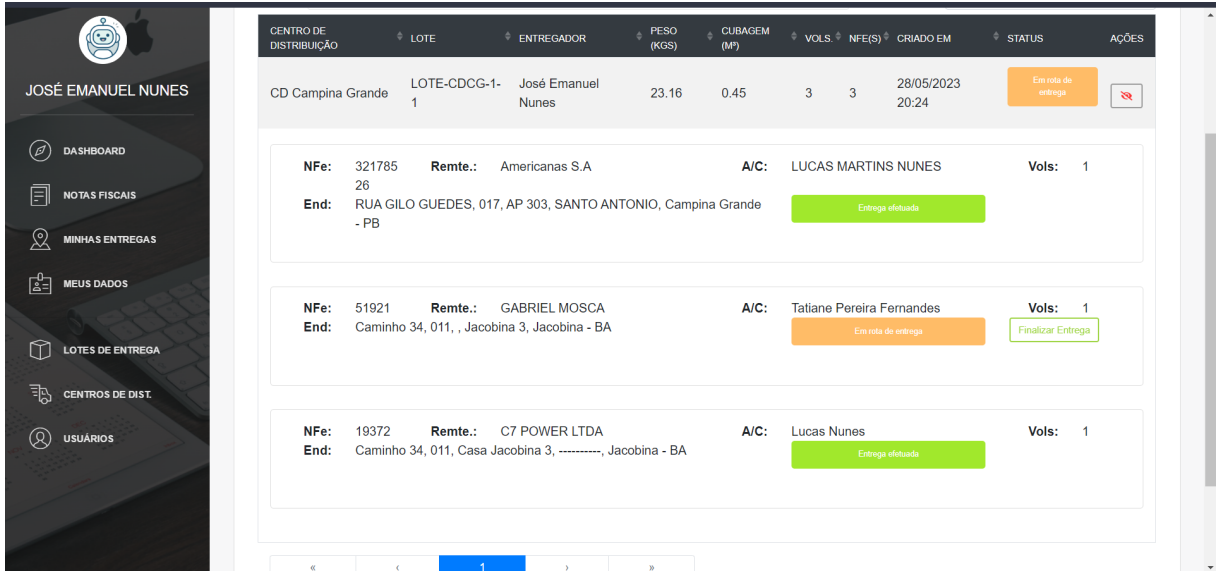


Figura 15 – Ver detalhes da Entrega do usuário - Minhas entregas

Fonte: Elaborado pelo autor, 2023.

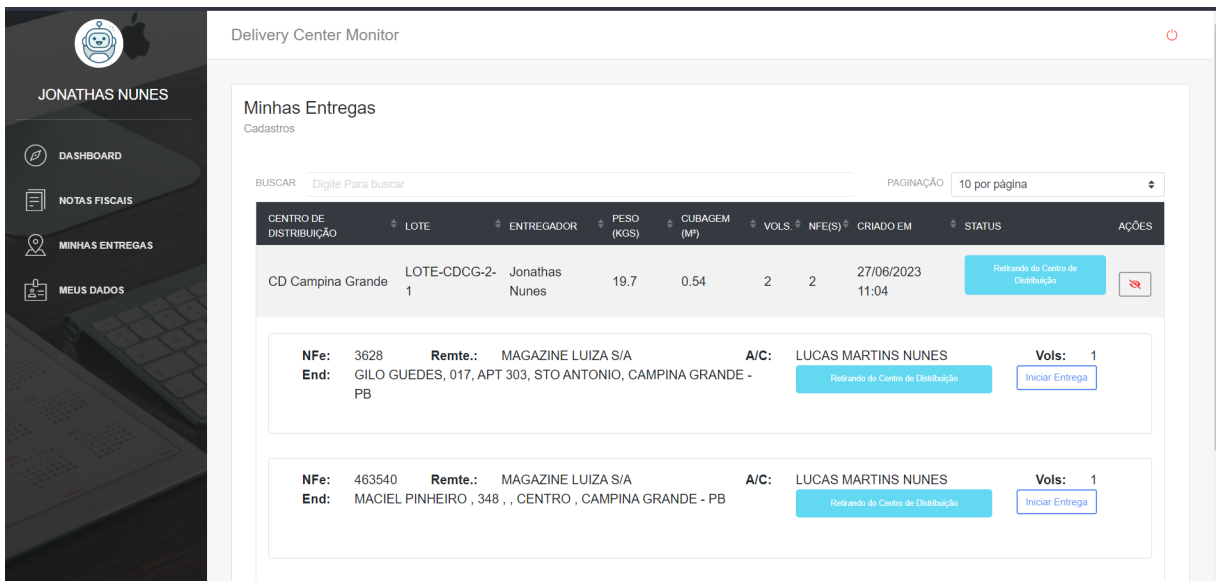


Figura 16 – Iniciar entrega de nota presente no lote - Minhas entregas

Fonte: Elaborado pelo autor, 2023.

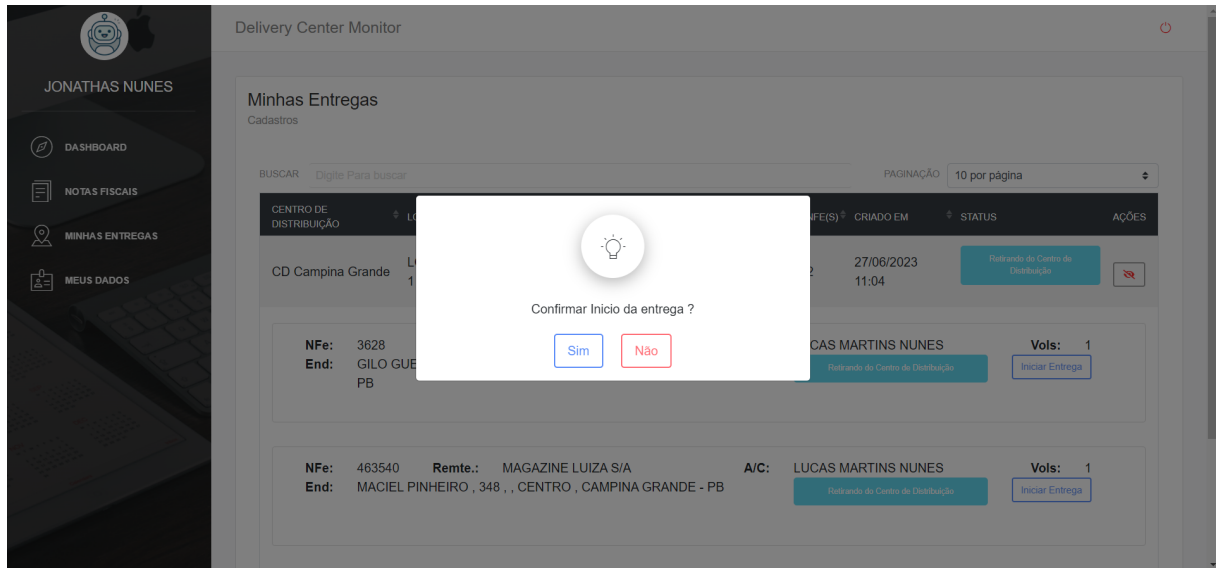


Figura 17 – Confirmar inicio de entrega - Minhas entregas

Fonte: Elaborado pelo autor, 2023.

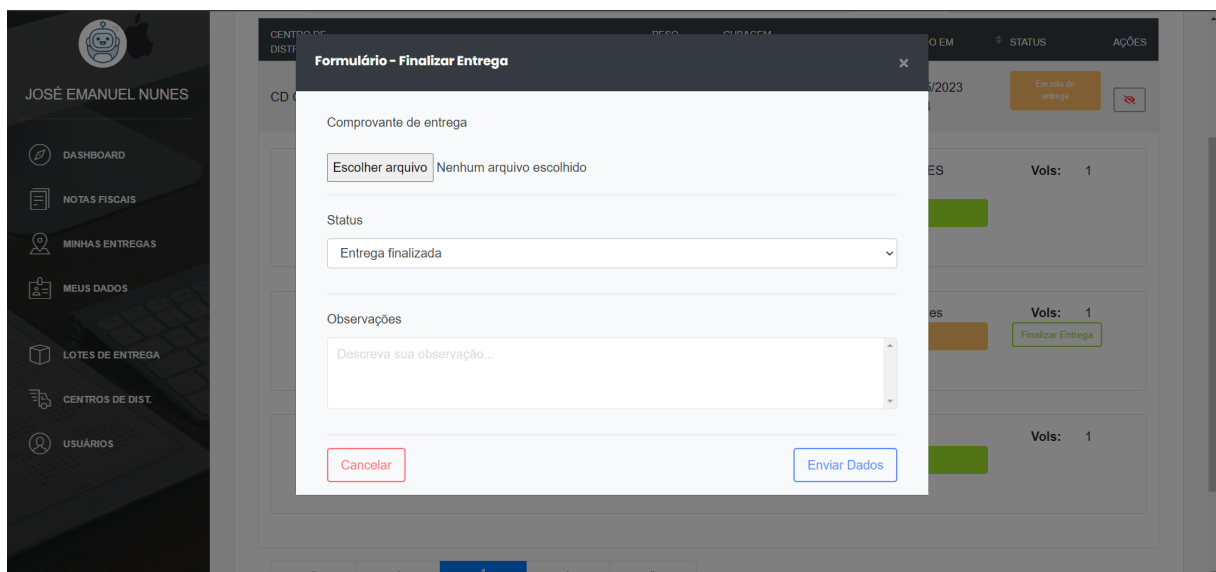


Figura 18 – Finalizar entrega - Minhas entregas

Fonte: Elaborado pelo autor, 2023.

Meus Dados: Os entregadores podem visualizar seus dados registrados no sistema, incluindo informações de contato e veículo associado. Além disso, têm a possibilidade de realizar alterações nesses dados, caso necessário.

Delivery Center Monitor

JOSÉ EMANUEL NUNES

Editar Perfil

NOME: José Emanuel Nunes | E-MAIL: admin@admin.com | CPF: 920.983.920-01

LOGRADOURO: Logradouro

NÚMERO: Número | REFERÊNCIA: Referência | CEP: CEP

CIDADE: Campina Grande | ESTADO: Paraíba | CONTATO: NÃO INFORMADO

SENHA: Senha | CONFIRMAR SENHA: Confirmar Senha

Atualizar Perfil

José Emanuel Nunes
Administrador
Membro desde: 28/05/2023

Veículo Cadastrado: **Moto**
Identificador | Placa: JOJ-1254
Cubagem máxima: 10.00
Peso máximo: 166.20

Para alterar demais informações referentes ao seu cadastro contate um Administrador

Figura 19 – Manutenção de dados pessoais - Meus dados

Fonte: Elaborado pelo autor, 2023.

5.2.2 Funcionalidade para Administradores

Lotes de Entrega: Os administradores têm acesso a uma listagem completa de todos os lotes de entrega registrados em cada centro de distribuição. Isso permite uma visão abrangente das entregas em andamento, facilitando o acompanhamento e o gerenciamento eficiente das atividades.

Delivery Center Monitor

JOSÉ EMANUEL NUNES

Lotes de Entrega
Cadastros

CENTRO DE DISTRIBUIÇÃO: CD Campina Grande

BUSCAR: Digite Para buscar | PAGINAÇÃO: 10 por página

LOTE	ENTREGADOR	PESO (KGS)	CUBAGEM (M³)	VOLS.	NFE(S)	CRIADO EM	STATUS	AÇÕES
LOTE-CDCG-1-1	José Emanuel Nunes	23.16	0.45	3	3	28/05/2023 20:24	Em rota de entrega	

© Coded with ❤️ by Lucas Nunes.

Figura 20 – Listar Lotes de entrega

Fonte: Elaborado pelo autor, 2023.

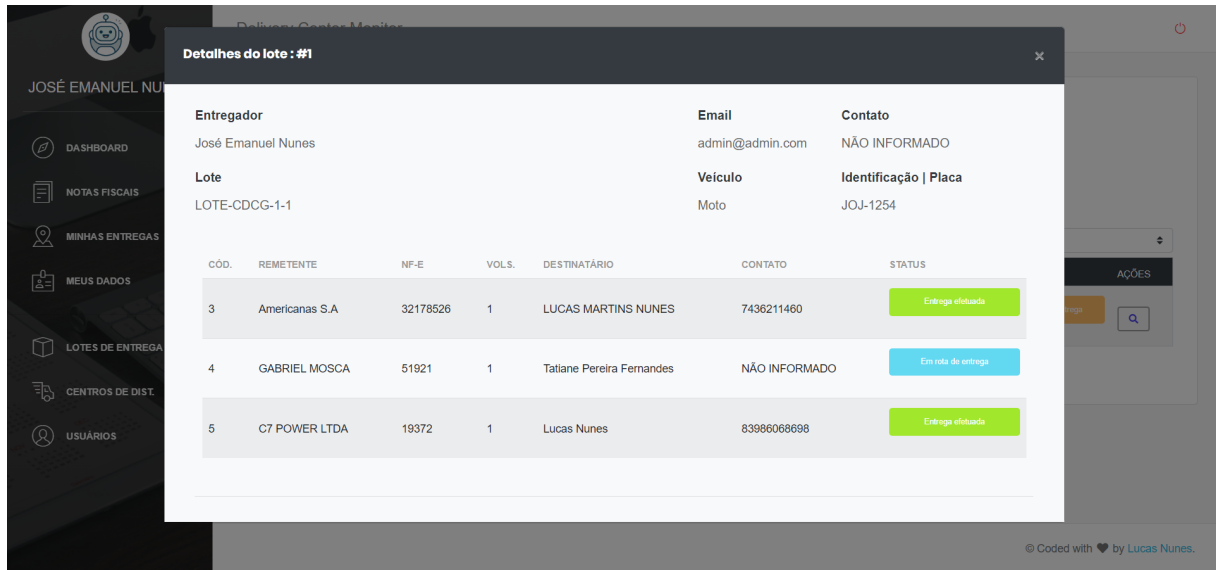


Figura 21 – Visualizar Lote de entrega

Fonte: Elaborado pelo autor, 2023.

Centros de Distribuição: Nessa funcionalidade, os administradores podem cadastrar, alterar e excluir centros de distribuição conforme necessário. Além disso, é possível realizar a entrada de notas fiscais no sistema, importando arquivos XML, agilizando o registro e a organização das notas fiscais.

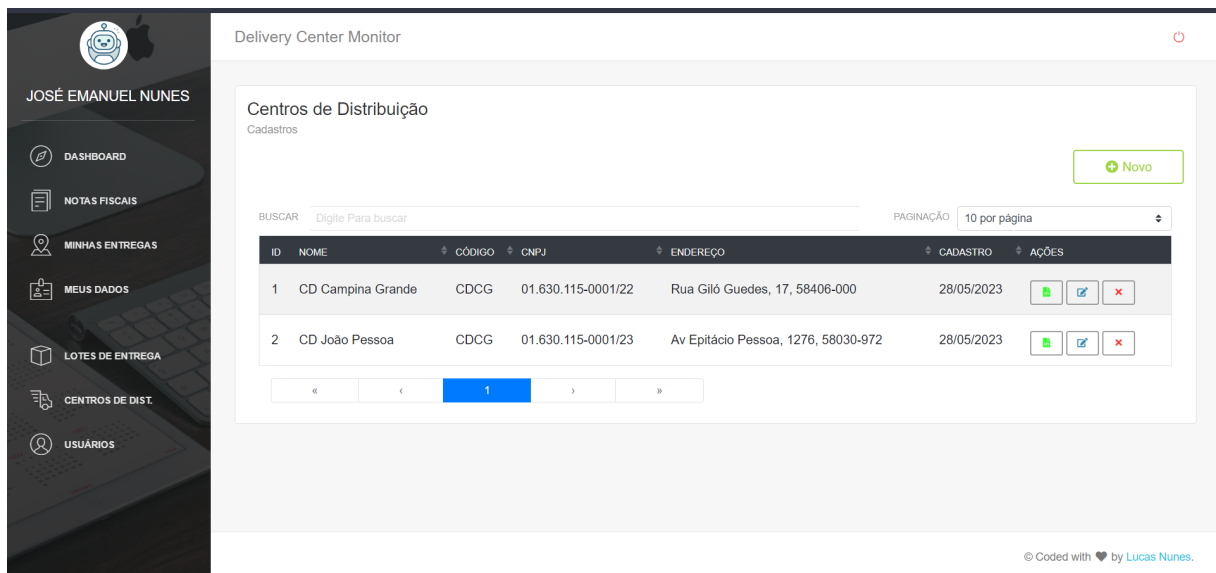


Figura 22 – Listar Centros de distribuição

Fonte: Elaborado pelo autor, 2023.

Formulário - Cadastro de Centros de Distribuição

Nome	Nome Fantasia	Código	CNPJ
<input type="text" value="Nome Centro"/>	<input type="text" value="Nome Fantasia"/>	<input type="text" value="Código"/>	<input type="text" value="CNPJ"/>
Logradouro	Número	Referência	CEP
<input type="text" value="Logradouro"/>	<input type="text" value="Número"/>	<input type="text" value="Referência"/>	<input type="text" value="CEP"/>
Cidade	Estado		
<input type="text" value="Cidade"/>	<input type="text" value="Estado"/>		

© Coded with ❤️ by Lucas Nunes.

Figura 23 – Criar Centro de distribuição

Fonte: Elaborado pelo autor, 2023.

Formulário - Cadastro de Centros de Distribuição

Nome	Nome Fantasia	Código	CNPJ
<input type="text" value="CD Campina Grande"/>	<input type="text" value="Delivery Center CG LTDA"/>	<input type="text" value="CDCG"/>	<input type="text" value="01.630.115-0001/22"/>
Logradouro	Número	Referência	CEP
<input type="text" value="Rua Giló Guedes"/>	<input type="text" value="17"/>	<input type="text" value="Referência"/>	<input type="text" value="58406-000"/>
Cidade	Estado		
<input type="text" value="Campina Grande"/>	<input type="text" value="Paraíba"/>		

© Coded with ❤️ by Lucas Nunes.

Figura 24 – Editar Centro de distribuição

Fonte: Elaborado pelo autor, 2023.

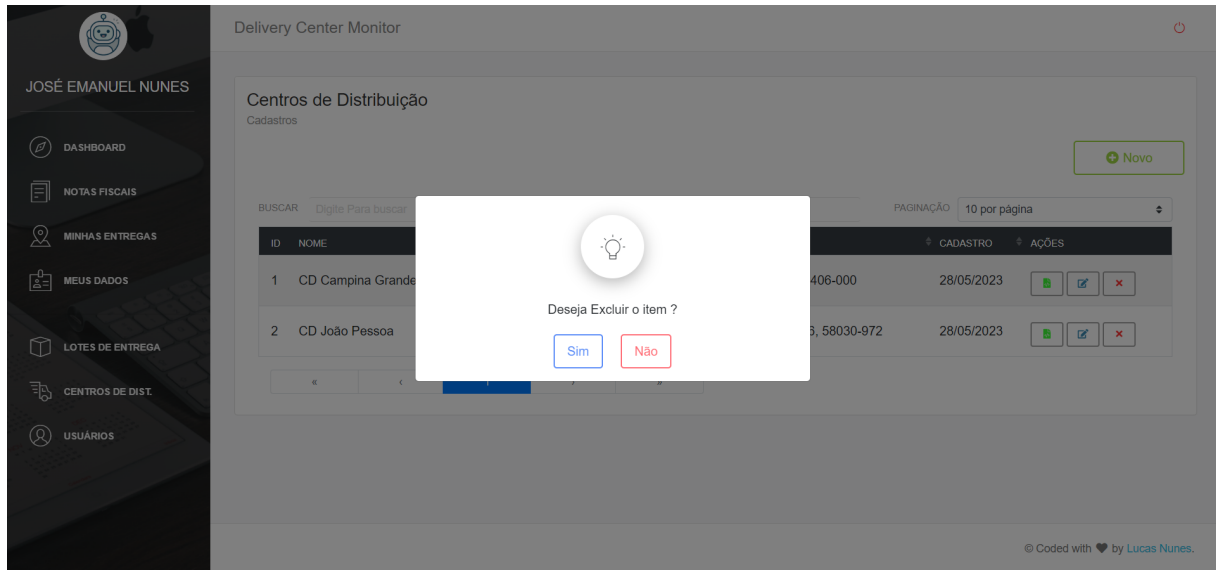


Figura 25 – Excluir Centro de distribuição

Fonte: Elaborado pelo autor, 2023.

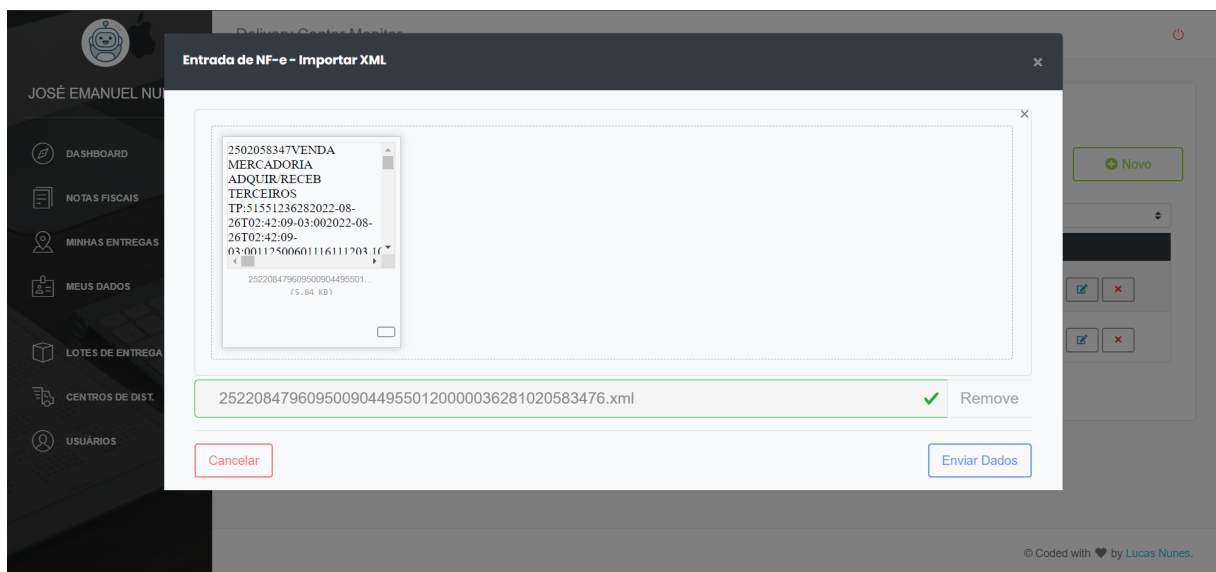


Figura 26 – Entrada de notas fiscais - importação de XML

Fonte: Elaborado pelo autor, 2023.

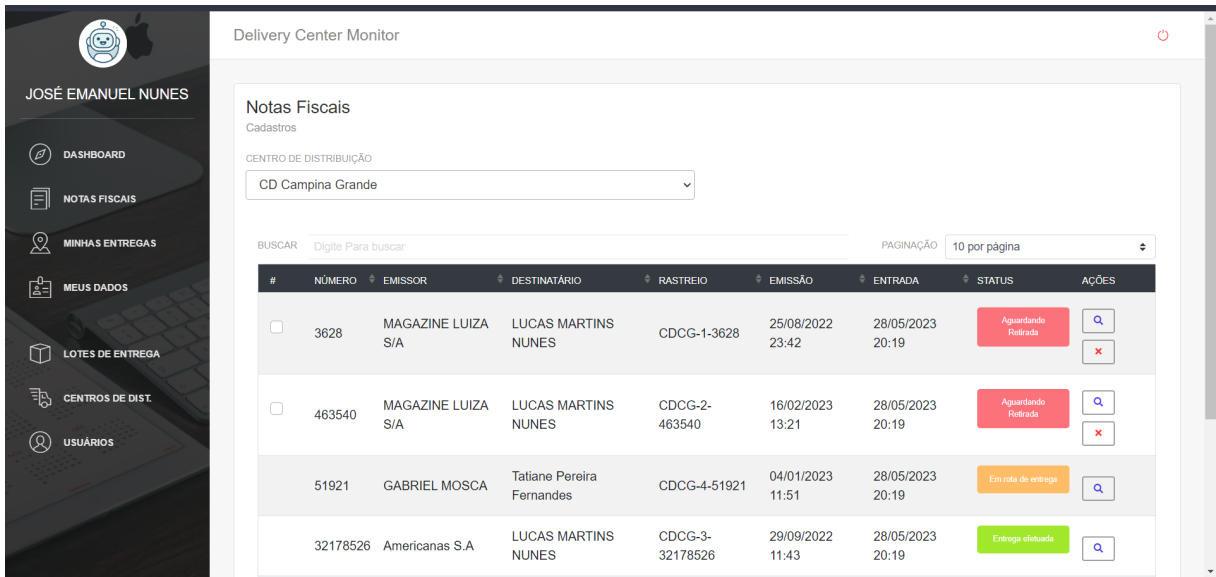


Figura 27 – Listar Notas Fiscais

Fonte: Elaborado pelo autor, 2023.

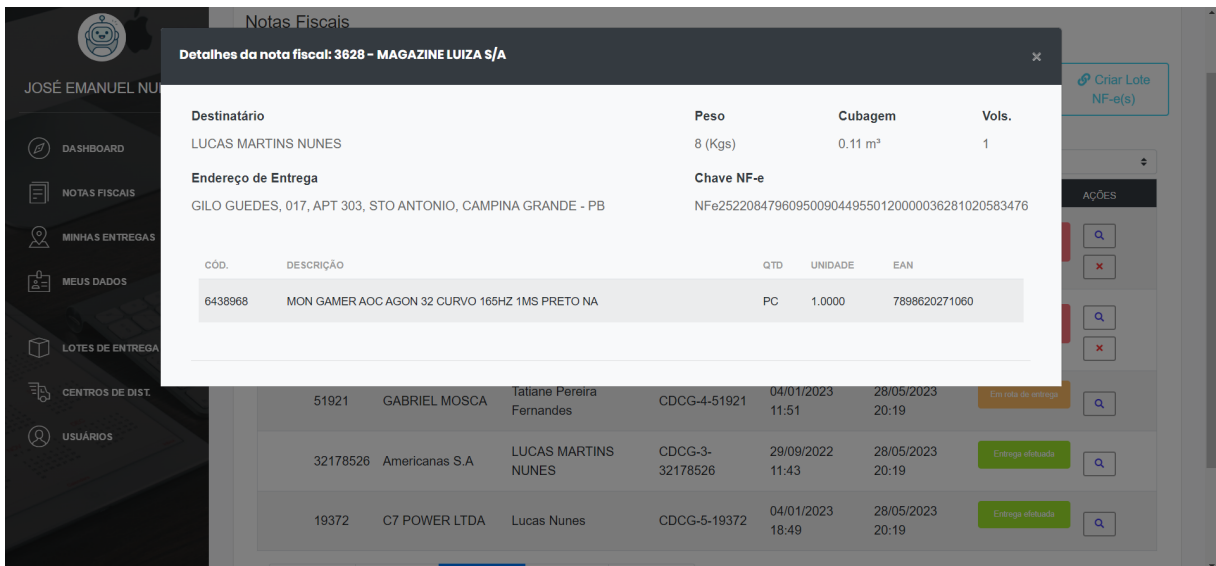


Figura 28 – Vizualizar detalhe de Nota Fiscal

Fonte: Elaborado pelo autor, 2023.

Usuários: Os administradores têm o controle completo sobre os usuários do sistema. Podem cadastrar, alterar e excluir usuários, além de vincular informações de veículos aos entregadores, simplificando o gerenciamento das atividades de entrega.

Delivery Center Monitor

JOSÉ EMANUEL NUNES

Dashboard

NOTAS FISCAIS

MINHAS ENTREGAS

MEUS DADOS

LOTES DE ENTREGA

CENTROS DE DIST.

USUÁRIOS

Usuários

Cadastros

Novo

BUSCAR Digite Para buscar PAGINAÇÃO 10 por página

AVATAR	NOME	E-MAIL	TIPO	CADASTRO	AÇÕES
	José Emanuel Nunes	admin@admin.com	Administrador	28/05/2023	
	Jonathas Nunes	jonathas@delivery.com	Entregador Parceiro	28/05/2023	

« ‹ 1 › »

© Coded with ♥ by Lucas Nunes.

Figura 29 – Listar Usuários

Fonte: Elaborado pelo autor, 2023.

Delivery Center Monitor

JOSÉ EMANUEL NUNES

Dashboard

NOTAS FISCAIS

MINHAS ENTREGAS

MEUS DADOS

LOTES DE ENTREGA

CENTROS DE DIST.

USUÁRIOS

Formulário - Cadastro de Usuários

Nome: Nome Completo

E-mail: E-mail

CPF: CPF

Logradouro: Logradouro

Número: Número

Referência: Referência

CEP: CEP

Cidade: Cidade

Estado: Estado

Contato: Contato

Tipo: Tipo

Senha: Senha

Confirmar Senha: Confirmar Senha

Senha aleatória: Gerar

Cancelar

Enviar Dados

© Coded with ♥ by Lucas Nunes.

Figura 30 – Criar Usuários

Fonte: Elaborado pelo autor, 2023.

Formulário - Cadastro de Usuários

Nome: José Emanuel Nunes | E-mail: admin@admin.com | CPF: 920.983.920-01

Logradouro: Logradouro | Número: Número | Referência: Referência | CEP: CEP

Cidade: Campina Grande | Estado: Paraíba

Contato: NÃO INFORMADO | Tipo: Administrador

Buttons: Cancelar, Enviar Dados

Figura 31 – Editar Usuários

Fonte: Elaborado pelo autor, 2023.

Formulário - Cadastro de Veículo do usuário : José Emanuel Nunes

Tipos de Veículo: Moto

Identificador | Placa: JOJ-1254 | Cubagem Máxima (m³): 10.00 | Peso Máximo (kg): 166.20

Buttons: Cancelar, Enviar Dados

Figura 32 – Dados do veículo

Fonte: Elaborado pelo autor, 2023.

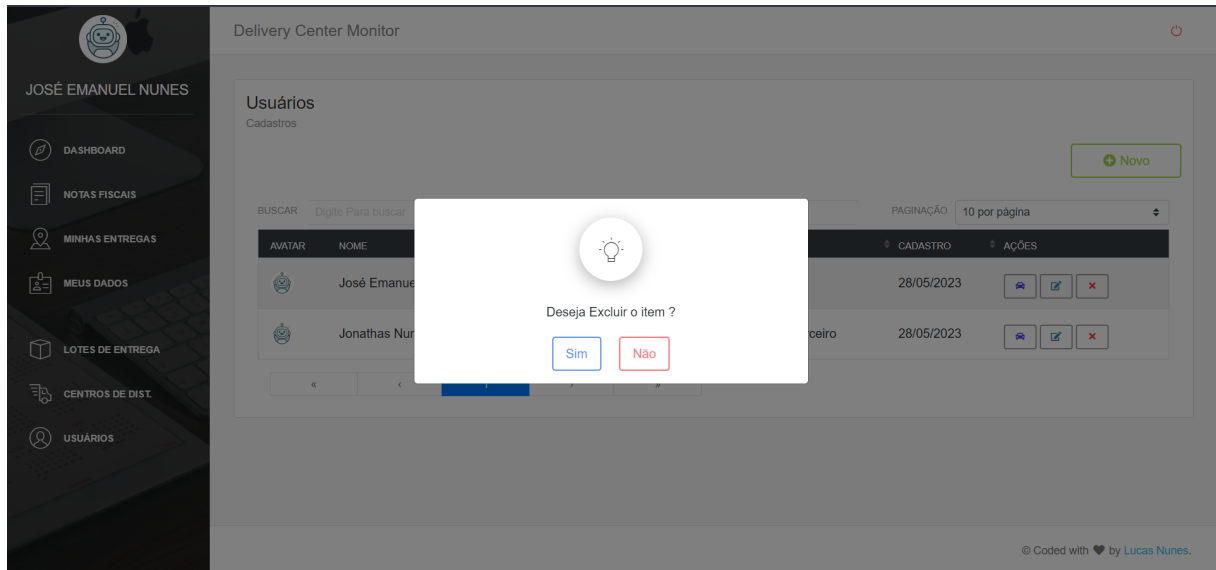


Figura 33 – Excluir Usuários

Fonte: Elaborado pelo autor, 2023.

Exclusão de Notas Fiscais e Lotes de Entrega: Os administradores têm a capacidade de excluir notas fiscais que não estão vinculadas a um lote de entrega e também excluir lotes de entrega que ainda não tiveram a entrega iniciada.

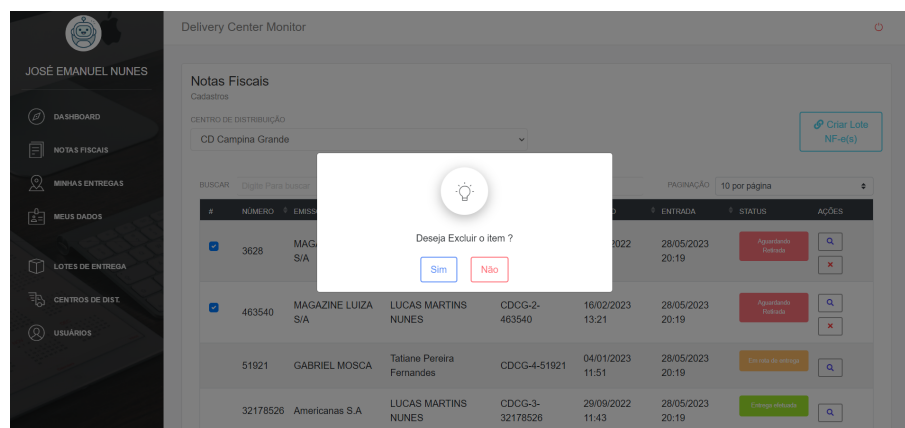


Figura 34 – Excluir Nota Fiscal

Fonte: Elaborado pelo autor, 2023.

Essas funcionalidades abrangentes do sistema de entregas do centro de distribuição proporcionam uma experiência completa, permitindo aos usuários gerenciar eficientemente as atividades de entrega, acompanhar o status e realizar as operações necessárias para um processo de entrega seguro e transparente.

6 RESULTADOS E DISCUSSÃO

No presente estudo, o software de entregas do centro de distribuição foi submetido a testes iniciais com um grupo restrito de usuários. Durante essa fase de testes, o software demonstrou um desempenho satisfatório, cumprindo as funcionalidades propostas e atendendo às necessidades dos usuários. Os resultados obtidos até o momento indicam que o software possui uma interface intuitiva e de fácil utilização, permitindo aos usuários administradores realizar o cadastro de novos usuários, vincular veículos e gerenciar as notas fiscais de forma eficiente. Da mesma forma, os entregadores puderam gerar lotes de entrega a partir das notas fiscais e utilizar a funcionalidade de rastreamento em um tempo de resposta de baixa latência, proporcionando aos destinatários a possibilidade de acompanhar a localização das entregas.

6.1 Testes e Validação do Software

Embora os testes tenham sido conduzidos com um número limitado de usuários, a expectativa é que o software seja liberado em breve para o público em geral. Com o lançamento para um maior número de usuários, espera-se obter um feedback mais abrangente e identificar possíveis melhorias e ajustes necessários para otimizar, ainda mais, a experiência do usuário. A disponibilização do software para o público em geral permitirá uma análise mais ampla de sua eficácia e funcionalidade em diferentes cenários e condições.

Neste estágio inicial, é importante ressaltar que os resultados e discussões apresentados são preliminares e baseados em testes limitados. Portanto, é necessário um acompanhamento contínuo e a coleta de feedbacks dos usuários para aprimorar e aperfeiçoar o software de entregas do centro de distribuição. Com a implementação de melhorias e ajustes com base nos resultados e discussões obtidos, espera-se que o software atinja um alto nível de satisfação dos usuários e contribua para uma eficiente gestão de entregas em centros de distribuição.

No próximo estágio do projeto, será realizado um processo de avaliação mais abrangente, com a participação de um número maior de usuários representativos do público-alvo. Esse processo permitirá coletar dados adicionais sobre a usabilidade, desempenho e eficácia do software, possibilitando uma análise mais aprofundada e embasada para

validação e refinamento do sistema.

6.2 Avaliação das Funcionalidades Apresentadas

A avaliação das funcionalidades apresentadas no software de entregas do centro de distribuição é fundamental para verificar a eficácia e a qualidade do sistema desenvolvido. Para isso, foram conduzidos testes e análises com o objetivo de avaliar cada uma das funcionalidades implementadas.

Em relação ao painel administrativo, foi verificado se o cadastro de usuários, tanto administradores quanto entregadores, ocorreu de maneira correta e se o vínculo entre usuários e veículos foi estabelecido adequadamente. Além disso, foi avaliada a facilidade de uso e a navegabilidade do painel, verificando se as informações são apresentadas de forma clara e organizada.

No que diz respeito à entrada de notas fiscais no centro de distribuição, foi testada a funcionalidade de importação de arquivos XML e verificado se o sistema processa e armazena corretamente as informações das notas fiscais. Além disso, foi também avaliado se o usuário administrador possui as permissões adequadas para realizar essa operação.

A geração de lotes de entrega a partir das notas fiscais foi testada para verificar se o sistema consegue agrupar as notas fiscais de forma eficiente, considerando critérios como destino, volume, peso, entre outros. Assim como, foram avaliadas a facilidade de uso e a clareza das informações apresentadas durante o processo de geração dos lotes.

A funcionalidade de rastreamento foi avaliada tanto do ponto de vista do entregador quanto do destinatário. Foi verificado se o entregador consegue enviar suas coordenadas para o sistema e se as atualizações de localização são refletidas corretamente no mapa da tela de rastreamento. Da mesma forma, foi avaliada a facilidade de uso e a clareza das informações apresentadas para o destinatário, permitindo que ele acompanhe o status da entrega com um tempo de resposta de baixa latência.

Por fim, a funcionalidade de finalização da entrega, com a possibilidade de anexar um comprovante de sucesso e uma observação opcional, também foi avaliada. Foi verificado se o entregador consegue concluir a entrega de forma adequada, registrando todas as informações necessárias para o encerramento do processo.

Durante a avaliação das funcionalidades, foram observados e registrados os resultados obtidos, identificando-se possíveis problemas, erros ou dificuldades encontradas pelos usuários. Esses feedbacks são essenciais para aprimorar o sistema, corrigindo eventuais

falhas e ajustando as funcionalidades de acordo com as necessidades dos usuários.

Em linhas gerais, com base nos testes realizados e nos feedbacks coletados, será possível realizar melhorias e otimizações no software de entregas do centro de distribuição, visando aperfeiçoar a experiência do usuário e garantir um sistema eficiente e confiável.

7 CONSIDERAÇÕES FINAIS

7.1 Contribuições do Trabalho

O desenvolvimento e implementação do software de entregas para o centro de distribuição poderá apresentar diversas contribuições significativas. Através da aplicação, será possível aprimorar o processo de logística, proporcionando uma melhor experiência para os usuários ao permitir o rastreamento das entregas. Além disso, o software contribuirá para o fortalecimento das pequenas empresas, oferecendo uma solução acessível e eficiente para a gestão de suas entregas. Portanto, por meio da análise dos resultados obtidos, foi possível identificar as funcionalidades bem-sucedidas e as áreas que necessitam de melhorias, fornecendo um ponto de partida para futuras otimizações.

7.2 Limitações e possíveis melhorias

Apesar dos avanços alcançados com o Delivery Center, é importante reconhecer as limitações encontradas durante o desenvolvimento e implementação do projeto. Algumas das limitações identificadas incluem a necessidade de uma maior divulgação e adoção do software por parte das empresas de menor porte e a disponibilidade de recursos tecnológicos por parte dos usuários finais. Em vista disso, reconhecemos que a integração com outras plataformas e sistemas existentes pode ser um desafio em termos de compatibilidade e interoperabilidade. Para melhorar o sistema, é recomendado investir em estratégias de marketing para promover a adoção do software por um maior número de empresas, bem como aprimorar a usabilidade e a interface do usuário. Além disso, a integração com outras plataformas e serviços populares de compras pode ser explorada para expandir ainda mais a funcionalidade do software. Com isso, outro ponto a ressaltar é sempre buscar por melhorias e correções de possíveis bugs.

7.3 Conclusão

O presente trabalho teve como objetivo desenvolver um software de entregas para um centro de distribuição, visando melhorar a eficiência e a transparência do processo logístico. Através da revisão bibliográfica, análise de requisitos e implementação do sistema, foi possível constatar que o software apresenta funcionalidades relevantes para otimizar as operações de entrega. A partir dos testes e avaliações realizados, observou-se que

o software atendeu às expectativas, fornecendo um rastreamento em um tempo de resposta de baixa latência, facilidade na geração de lotes de entrega e melhor controle administrativo. Embora algumas limitações tenham sido identificadas, o software demonstrou potencial para auxiliar empresas de menor porte e estabelecer um vínculo de trabalho com a comunidade. Portanto, com a futura liberação para o público geral, espera-se que o software seja amplamente adotado, contribuindo para aprimorar a logística de entregas e proporcionar uma melhor experiência aos usuários.

REFERÊNCIAS

BAUDUINO, Plínio. **Dominando JavaScript com JQuery**. São Paulo: Casa do código, 2012.

BRASIL, Daniel. **Centro de Distribuição: o que é, como funciona, desafios e por que investir**. HiveCloud, 2023.

Disponível em <<https://www.hivecloud.com.br/post/centro-de-distribuicao>>. Acesso em Junho, 2023.

Comó nació Rappi explicado por su fundador Simon Borrero. **Rappi**, 2018.

Disponível em <<https://blog.rappi.com/como-nacio-rappi>>. Acesso em Junho, 2023.

FERREIRA, Adriano. **Aplicativos delivery: veja cinco apps para entrega de comida e produtos**. TechTudo, 2019.

Disponível em <<https://www.techtudo.com.br/listas/2019/04/aplicativos-delivery-veja-cinco-apps-para-entrega-de-comida-e-produtos.ghhtml>>. Acesso em Junho, 2023.

FLANAGAN, David. **JavaScript: O Guia Definitivo**. 6ª ed. Porto Alegre: Bookman, 2012.

JUNIOR, Francie. Delivery transformou tendência em necessidade e continua em crescimento. **JORNAL USP**, 2021.

Disponível em: <<https://jornal.usp.br/atualidades/delivery-transformou-tendencia-em-necessidade-e-continua-em-crescimento/>> Acesso em: 28 de junho de 2023.

LARAVEL Meet Laravel. **Laravel**, 2011.

Disponível em <<https://laravel.com/docs/10.x>>. Acesso em Junho, 2023.

LAZOTI, Rodrigo. **Armazenando Dados com Redis**. São Paulo: Casa do código, 2014.

LIMA, Cleyson. Fluxo de autenticação baseado em JWT. **Treina Web**, 2021.

Disponível em

<<https://www.treinaweb.com.br/blog/fluxo-de-autenticacao-baseado-em-jwt>>. Acesso em Junho, 2023.

Manual de Referência do MySQL 8.1. **MYSQL**, 2011.

Disponível em <<https://dev.mysql.com/doc/refman/8.0/en/>>. Acesso em Junho, 2023.

OTWELL, Taylor; BORGES, Pedro. **Laravel: De Aprendiz a Artesão (Brazilian Portuguese)**. Leanpub, 2013.

O que é o PHP?. **PHP**, 2001.

Disponível em <<https://www.php.net/manual/en/intro-whatism.php>>. Acesso em Junho, 2023.

O que é o iFood? Conheça a história e a operação da empresa. **Ifood**, 2023.

Disponível em <<https://www.news.ifood.com.br/o-que-e-o-ifood>>. Acesso em Junho, 2023.

O que é Vue.js?. **VUEJS** 2023.

Disponível em <<https://br.vuejs.org/v2/guide>>. Acesso em Junho, 2023.

RESS, Dayle; DALMOLIN, Luís; LAGUNA, Antônio. **Laravel: Code Bright:**

Desenvolvimento de aplicações web com framework Laravel 4 para iniciantes. Leanpub, 2014.

RODRIGUES, Horácio Wanderlei; BECHARA, Gabriela Natacha; GRUBBA, Leilane Serratine. Era digital e controle da informação. Revista Em Tempo, [n.p.], v. 20, n. 1, nov. 2020.

Disponível em: <<https://revista.univem.edu.br/emtempo/article/view/3268>>. Acesso em: 25 de maio de 2023.

SILVA, Maurício Samy. **HTML5 e CSS3: Domine a web do futuro**. São Paulo: Casa do código, 2014.

TAURION, César. **Cloud Computing - computação em nuvem: transformando o mundo da tecnologia da informação**. Rio de Janeiro: Brasport, 2009.

Tudo sobre a Amazon - História e Notícias. **CanalTech**, 2023.

Disponível em <<https://canaltech.com.br/empresa/amazon>>. Acesso em Junho, 2023.

Tudo o que você precisa saber sobre o Mercado Livre. **Mercado Livre**, 2023.
Disponível em <<https://www.mercadolivre.com.br/institucional/nos-comunicamos/noticia/tudo-sobre-o-mercado-livre>>. Acesso em Junho, 2023.

What Socket.IO is. **SOCKET.IO**, 2021.
Disponível em <<https://socket.io/docs/v4/>>. Acesso em Junho, 2023.

APÊNDICE A – MODELO ENTIDADE RELACIONAMENTO

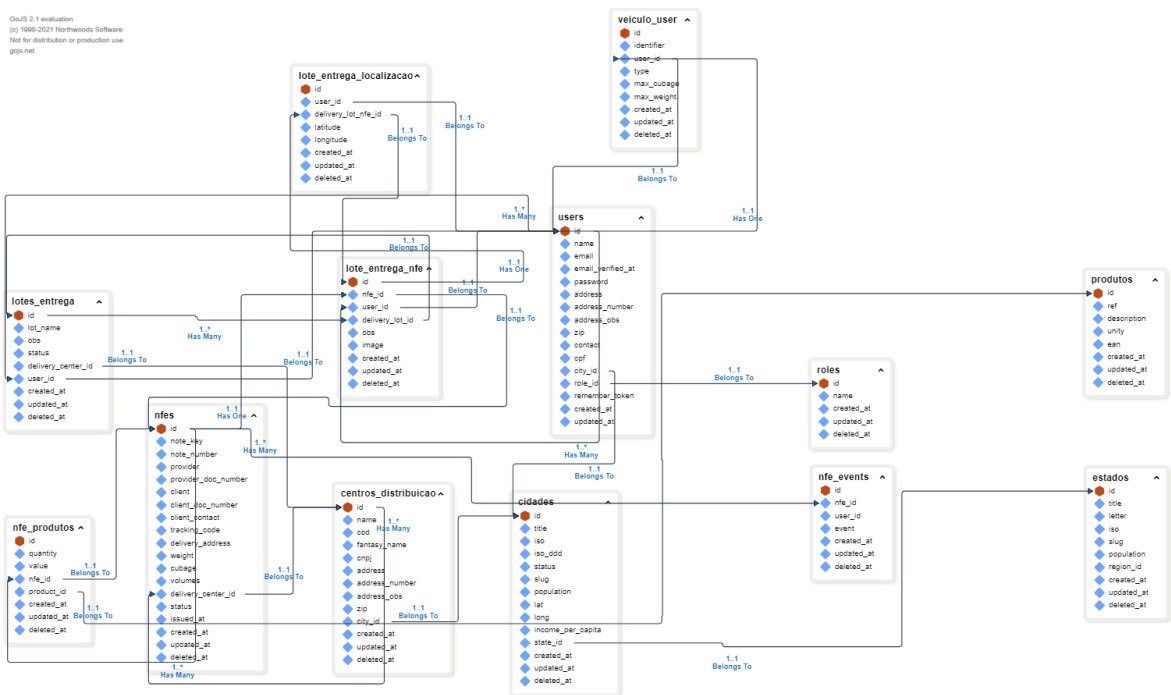


Figura 35 – Modelo Entidade Relacionamento

Fonte: Elaborado pelo autor, 2023.



Figura 36 – Diagrama de tabelas do banco de dados

Fonte: Elaborado pelo autor, 2023.

APÊNDICE B – DEMAIS TELAS DO SISTEMA



Figura 37 – Tela de rastreamento por código

Fonte: Elaborado pelo autor, 2023.



Figura 38 – Tela de rastreamento por documentos

Fonte: Elaborado pelo autor, 2023.

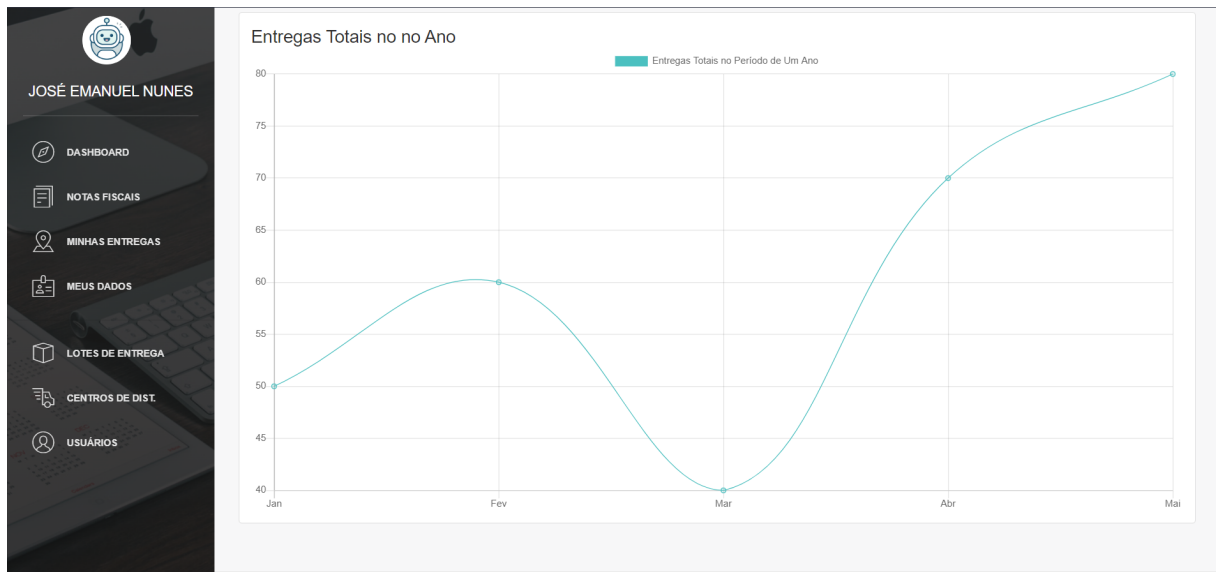


Figura 39 – Painel administrativo (Admin) - Dashboard

Fonte: Elaborado pelo autor, 2023.

Notas Fiscais

Cadastros

CENTRO DE DISTRIBUIÇÃO

CD Campina Grande

USUÁRIO: JOSÉ EMANUEL NUNES

USUÁRIO: DASHBOARD

USUÁRIO: NOTAS FISCAIS

USUÁRIO: MINHAS ENTREGAS

USUÁRIO: MEUS DADOS

USUÁRIO: LOTES DE ENTREGA

USUÁRIO: CENTROS DE DIST.

USUÁRIO: USUÁRIOS

USUÁRIO: BUSCAR Digite Para buscar

USUÁRIO: PAGINAÇÃO 10 por página

#	NÚMERO	EMISSOR	DESTINATÁRIO	RASTREIO	EMIÇÃO	ENTRADA	STATUS	AÇÕES
<input checked="" type="checkbox"/>	3628	MAGAZINE LUIZA S/A	LUCAS MARTINS NUNES	CDCG-1-3628	25/08/2022 23:42	28/05/2023 20:19	Aguardando Retirada	<input type="button" value="Q"/> <input type="button" value="X"/>
<input checked="" type="checkbox"/>	463540	MAGAZINE LUIZA S/A	LUCAS MARTINS NUNES	CDCG-2-463540	16/02/2023 13:21	28/05/2023 20:19	Aguardando Retirada	<input type="button" value="Q"/> <input type="button" value="X"/>
	51921	GABRIEL MOSCA	Tatiane Pereira Fernandes	CDCG-4-51921	04/01/2023 11:51	28/05/2023 20:19	Em rota de entrega	<input type="button" value="Q"/>
	32178526	Americanas S.A	LUCAS MARTINS NUNES	CDCG-3-32178526	29/09/2022 11:43	28/05/2023 20:19	Entrega efetuada	<input type="button" value="Q"/>
	19372	C7 POWER LTDA	Lucas Nunes	CDCG-5-19372	04/01/2023 18:49	28/05/2023 20:19	Entrega efetuada	<input type="button" value="Q"/>

USUÁRIO: Criar Lote NF-e(s)

Figura 40 – Gerar Lote de entrega - Notas Fiscais (Admin/Entregador)

Fonte: Elaborado pelo autor, 2023.

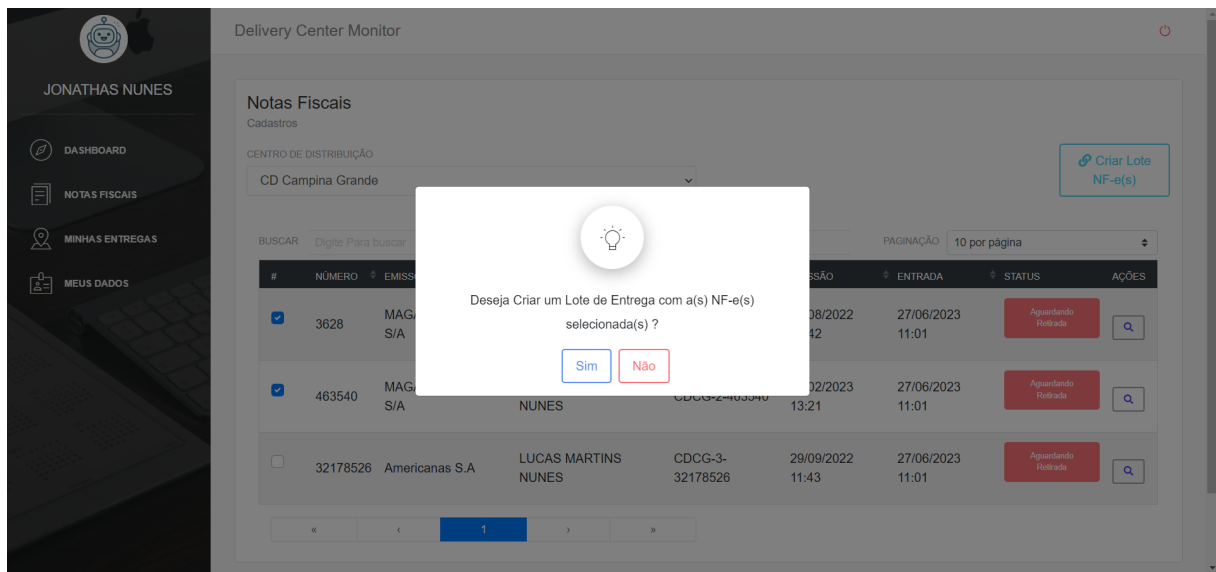


Figura 41 – Gerar Lote de entrega - Notas Fiscais (Admin/Entregador)

Fonte: Elaborado pelo autor, 2023.

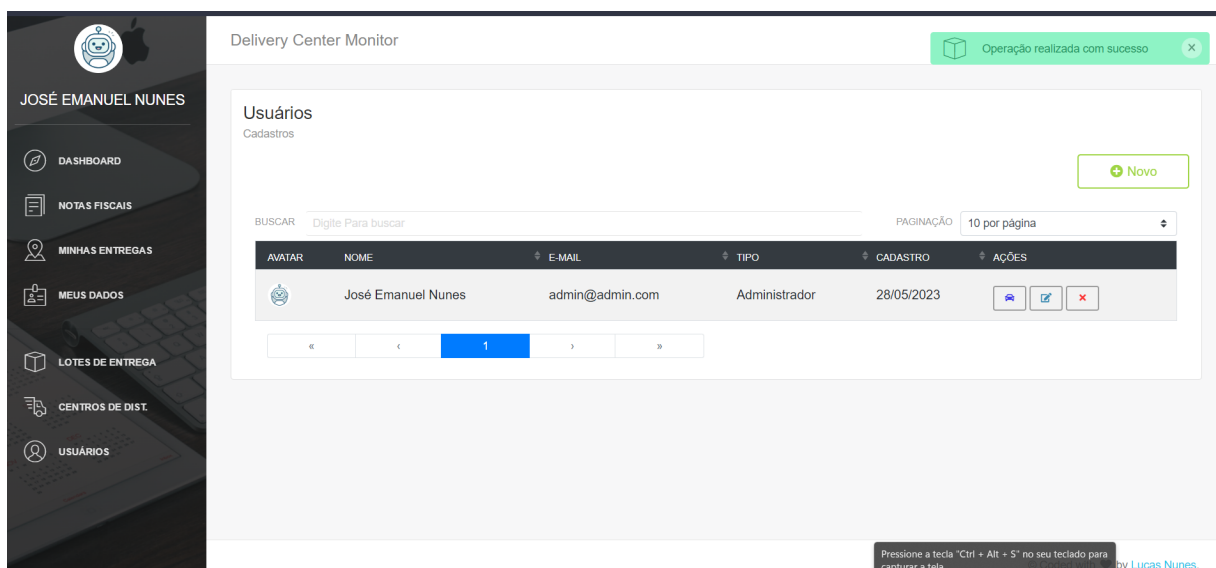


Figura 42 – Exemplo de feedback de sucesso (Admin/Entregador)

Fonte: Elaborado pelo autor, 2023.

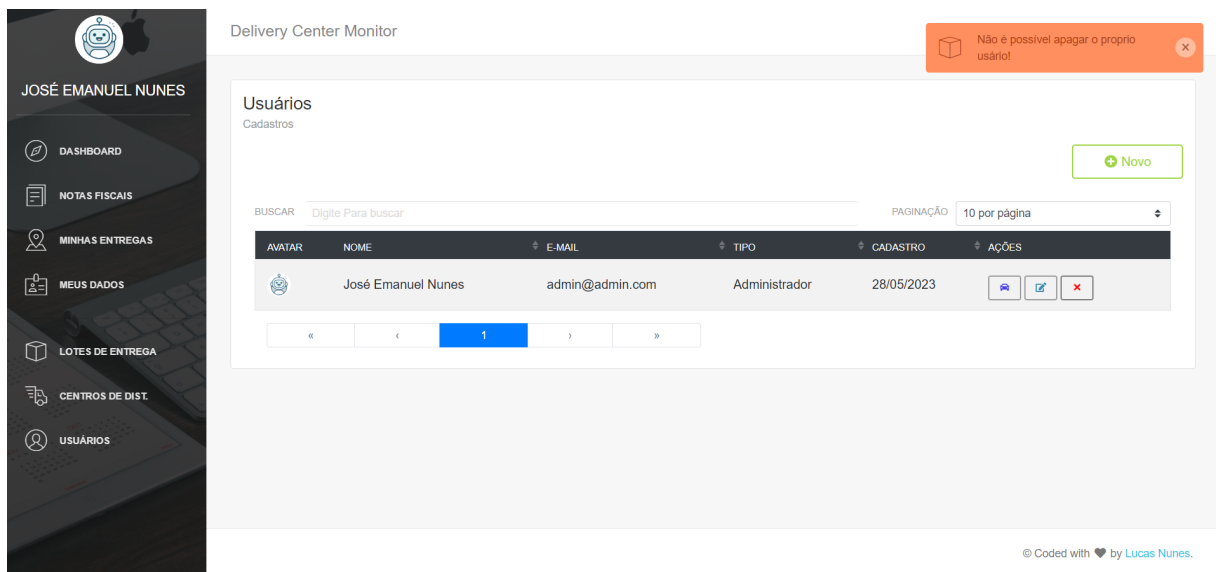


Figura 43 – Exemplo de Feedback de erro (Admin/Entregador)

Fonte: Elaborado pelo autor, 2023.

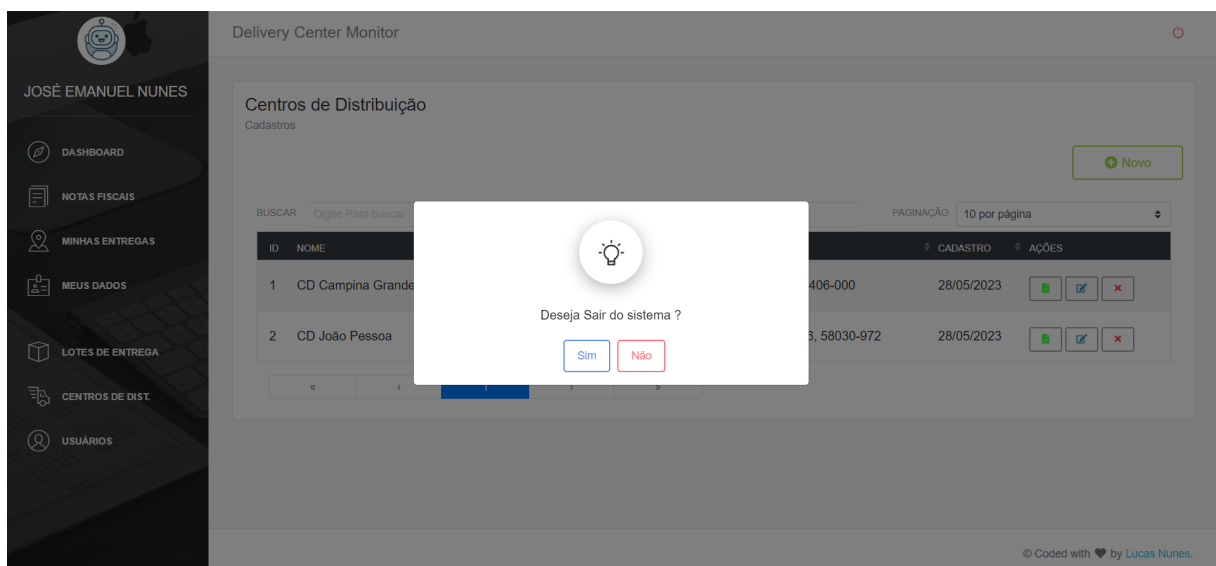


Figura 44 – Logout do sistema (Admin/Entregador)

Fonte: Elaborado pelo autor, 2023.

APÊNDICE C – REPOSITÓRIO GIT

<https://gitlab.com/luk4smn/delivery-center-monitor>