

**COLÉGIO ESTADUAL DE EDUCAÇÃO PROFISSIONAL  
PEDRO BOARETTO NETO  
CURSO TÉCNICO EM INFORMÁTICA**

**LUIZ ANTONIO COMIRAN BUENO**

**CONFEITARIA PAPAQUI**

**CASCADEL - PR  
2023**

**LUIZ ANTONIO COMIRAN BUENO**

**CONFEITARIA PAPAQUI**

Projeto de Desenvolvimento de Software  
do Curso Técnico em Informática do  
Colégio Estadual de Educação  
Profissional Pedro Boaretto Neto –  
Cascavel, Paraná.

Orientadores: Prof<sup>a</sup> Aparecida S.Ferreira<sup>1</sup>  
Prof. Rony Peterson Cordeiro<sup>2</sup>  
Prof. Cleonice Maria Pereira<sup>3</sup>

**CASCADEL - PR**  
**2023**

---

<sup>1</sup>Especialista em Educação Permanente: Saúde e educação pela FioCruz – Fundação Osvaldo Cruz. Especialista em tecnologias da Informação pela UNIVEL – União Educacional de Cascavel. Pedagoga formada pela UNIPAR – Universidade Paranaense. Professora do núcleo técnico do Estado do Paraná – Ensino médio técnico.

<sup>2</sup>

<sup>3</sup>

**LUIZ ANTONIO COMIRAN BUENO**

**CONFEITARIA PAPAQUI**

Este Projeto de Conclusão de Curso foi julgado e aprovado pelo Curso Técnico em Informática do Colégio Estadual de Educação Profissional Pedro Boaretto Neto.

Cascavel, Pr., xx de Xxxxx de 2023

**COMISSÃO EXAMINADOR**

---

Prof<sup>a</sup>. Aparecida da S. Ferreira<sup>1</sup>  
Especialista em Tecnologia da  
Informação  
*Faculdade de Ciências Sociais Aplicadas  
de Cascavel*  
Orientadora

---

Prof. Rony Peterson Cordeiro

Web Design

---

Prof<sup>a</sup>. Cleonice Maria Pereira  
Especialização em  
PSICOPEDAGOGIA.

Banco de dados

---

Prof<sup>a</sup> Márcio Alves dos Santos  
Especialista em Gestão e Docência  
no ensino superior, médio e técnico.  
Coordenador de curso

## Sumário

1	INTRODUÇÃO	5
1.1	Apresentação do Problema .....	6
2	OBJETIVOS	8
3	METODOLOGIA	9
4	REFERENCIAL TEÓRICO	11
5	DOCUMENTAÇÃO do projeto	13
	<b>Coleta de dados.....</b>	<b>15</b>
	Coleta de dados .....	15
	Treinamento .....	15
	Validação .....	16
	Deploy .....	16
	Monitoramento .....	16
5.1	Requisitos .....	17
5.1.1	Requisitos funcionais .....	17
	<b>5.1.2 Requisitos não funcionais .....</b>	<b>18</b>
5.2	Diagrama de Contexto.....	19
5.3	Diagrama de Fluxo de dados .....	21
5.4	Diagrama de Entidade e relacionamento .....	22
5.5	Dicionário de Dados .....	24
5.6	Diagrama de Caso de Uso .....	25
5.7	Diagrama de Classe .....	26
5.8	Diagrama de Sequência .....	27
5.9	Diagrama de Atividade .....	28
6	Telas	30
7	Conclusão	33
8	REFERÊNCIAS	35

## 1 INTRODUÇÃO

GEMINI (2024), na era digital em constante evolução, a demanda por plataformas que facilitem a venda e a gestão de pedidos para confeitarias nunca foi tão alta. Diante desse cenário, surge a necessidade de um espaço virtual que não apenas permita às confeitarias exporem e venderem seus produtos, mas também ofereça uma experiência integrada e interativa para a gestão de pedidos e relacionamento com os clientes. É nesse contexto que se insere nosso projeto: a criação de um site dedicado a auxiliar confeitarias na venda de seus produtos e no gerenciamento de pedidos, proporcionando uma plataforma completa e especializada para impulsionar seus negócios.

Para MONTAGNE (1938), a confeitaria é uma arte culinária dedicada à criação de sobremesas, bolos, tortas, e outros doces que encantam tanto pelo sabor quanto pela aparência visual. Esta área da gastronomia exige conhecimento técnico, criatividade e uma compreensão profunda dos ingredientes para criar receitas que são verdadeiras obras de arte. Segundo ROUX (2007.) A confeitaria, também conhecida como pâtisserie, é a área da culinária focada na preparação de doces e sobremesas. Profissionais dessa área, chamados confeitadores, dominam técnicas que incluem desde o preparo de massas, cremes, recheios e coberturas até a decoração sofisticada de bolos e tortas. Eles transformam ingredientes como açúcar, farinha, ovos, e manteiga em produtos que não apenas são deliciosos, mas também visualmente atraentes. A confeitaria possui uma grande importância cultural, com cada região do mundo possuindo suas próprias receitas tradicionais. Por exemplo, o Brasil é conhecido por seus brigadeiros, enquanto a França é famosa por seus macarons e éclairs. Essas sobremesas muitas vezes carregam histórias e memórias, tornando-as uma parte essencial de celebrações e tradições.

Para UDE (2017.) A confeitaria em casa tem se tornado cada vez mais popular, especialmente com o aumento de tutoriais online e programas de culinária que inspiram pessoas a experimentar novas receitas e técnicas. Ao contrário da confeitaria profissional, onde há acesso a equipamentos especializados e ingredientes em grande quantidade, a confeitaria em casa geralmente envolve o uso de utensílios domésticos comuns e ingredientes acessíveis. Apesar das limitações, confeitaria em casa oferece uma grande liberdade criativa e pode ser uma atividade muito gratificante. As pessoas podem adaptar receitas para atender às suas preferências e

dietas, e muitas vezes o processo de criação é tão prazeroso quanto o resultado final. Confeitar em casa também permite que os indivíduos criem sobremesas personalizadas para ocasiões especiais, adicionando um toque pessoal às festividades. Conforme REYNOLDS (2020) com o crescimento da internet e do comércio eletrônico, ter uma presença online tornou-se essencial para qualquer negócio, incluindo a confeitaria. Um site de vendas permite que confeitadeiras alcancem um público mais amplo, ofereçam seus produtos de forma conveniente e aumentem suas vendas. Este documento explora a importância de sites de vendas para confeitadeiras e como eles podem ser criados para maximizar o sucesso no mercado digital.

SMITH (2019) Fala que para confeitadeiras, um site de vendas não é apenas uma vitrine virtual, mas uma ferramenta estratégica que pode impulsionar o crescimento do negócio. Ao ter um site, as confeitadeiras podem exibir seus produtos, como bolos, tortas, cupcakes, e outros doces, para um público muito maior do que seria possível em uma loja física. Isso é especialmente relevante em um mundo onde os consumidores estão cada vez mais comprando online.

Além de aumentar a visibilidade, um site permite que as confeitadeiras gerenciem melhor suas operações, como recebimento de pedidos, processamento de pagamentos, e até mesmo a coordenação de entregas. Com a integração de ferramentas de marketing digital, como SEO e redes sociais, as confeitadeiras podem atrair mais clientes e fidelizá-los ao longo do tempo.

Segundo a fala de SMITH (2019) A criação de um site de vendas é um passo crucial para confeitadeiras que desejam expandir seus negócios e alcançar novos clientes. Um site bem projetado, aliado a estratégias eficazes de marketing digital, pode transformar uma pequena confeitaria em uma marca conhecida e respeitada. Portanto, investir em uma presença online não é mais uma opção, mas uma necessidade para quem quer prosperar no mercado competitivo de hoje.

## **1.1 Apresentação do Problema**

O projeto Confeitaria PAPAQUI tem como objetivo saciar a falta de alcance de confeitadeiras de casa, podendo atingir um número quase infinito de pessoas com uma loja online, onde poderá se encontrar grande variedade de bolos, doces, entre outras delicias de confeitarias, também podendo agendar e fazer pedidos grandes e fazendo contato diretamente com a confeitadeira, assim garantindo uma alta facilidades de

encontrar clientes, e os clientes encontrar sua confeitaria.

## 2 OBJETIVOS

A criação de um site intuitivo e responsivo de uma confeitaria.

Experiência Envolvente e Intuitiva: Garantir que o site seja fácil de usar e visualmente agradável, proporcionando uma navegação fluida e uma experiência de usuário (UX) positiva.

Destacar a Qualidade e Criatividade dos Produtos: Utilizar imagens de alta qualidade e descrições detalhadas para apresentar os produtos de confeitaria de forma atraente, destacando seus diferenciais e a criatividade dos itens oferecidos.

Facilitar a Navegação e a Compra Online: Implementar funcionalidades de e-commerce que permitam aos clientes explorar o catálogo de produtos, realizar pedidos e pagamentos de maneira simples e segura.

Promover a Marca e os Valores: Refletir os valores e a identidade da confeitaria através do design do site, conteúdos e campanhas de marketing, reforçando a imagem da marca e criando uma conexão com os clientes.



### 3 METODOLOGIA

O método comparativo compreende em estabelecer paralelos entre dois ou mais objetos de estudo, para analisar semelhanças e diferenças. Em outras palavras, é um método para comprovar ou refutar teorias e hipóteses que se baseia em comparações. Nas ideias de COELHO (2022), o método comparativo se baseia no método das semelhanças e das diferenças ou na observação de variações concomitantes“. Nessa perspectiva, no processo de comparação, identificar as semelhanças permite organizar e relacionar o novo conceito com o conhecimento que já existe. Por outro lado, estabelecer as diferenças permite discriminar o novo conceito de outros parecidos, para evitar confusão.

A Modelagem de Dados para DRUCKER (2024), é uma prática do campo da gestão e da análise de informações que busca representar visualmente a estrutura e as relações entre os dados em um sistema. Essencialmente, ela cria um mapa que descreve como as informações são organizadas e interagem dentro de um ambiente específico. Ao empregar técnicas e ferramentas adequadas, a modelagem de dados permite uma compreensão mais clara e sistemática deles, facilitando a interpretação, manipulação e tomada de decisões. Com isso ela desempenha um importante papel na integração de sistemas, no desenvolvimento de bancos de dados eficientes e na otimização de processos analíticos. Como funciona a modelagem de dados? A modelagem de dados é um processo que opera por meio de uma sequência de passos organizados, utilizando técnicas e ferramentas específicas para representar graficamente a estrutura, relacionamentos e características dos dados em um sistema. O primeiro passo é identificar os requisitos do sistema, ou seja, compreender as necessidades dos stakeholders em termos de dados. Os profissionais de modelagem de dados, em seguida, coletam informações relevantes para o contexto do sistema, incluindo dados sobre entidades, atributos, relacionamentos e restrições. Com base nos requisitos levantados são identificadas as entidades principais do sistema, em outras palavras, os objetos ou conceitos sobre os quais desejamos armazenar informações. Atributos, que são as características específicas das entidades, são associados a essas. Em seguida, estabelece-se relações entre diferentes entidades para representar como elas se conectam e interagem entre si. O processo de normalização é aplicado para eliminar redundâncias e melhorar a eficiência do banco de dados, organizando as informações de maneira lógica e eficaz.

Utilizando ferramentas gráficas, como diagramas de entidade-relacionamento (ER) ou diagramas UML com objetivo visualizar e comunicar a estrutura de dados de forma compreensível.

A implementação no banco de dados real ocorre a partir dos conceitos gerados durante o processo, traduzindo-os em esquemas físicos. Importante destacar que a modelagem de dados não é um processo estático, dessa forma, à medida que os requisitos evoluem, ajustes são realizados para garantir que ela continue a atender às necessidades em constante mudança do sistema.

## 4 REFERENCIAL TEÓRICO

**HTML** conforme EICH (2024), a base fundamental da web é o HTML (HyperText Markup Language). Essa linguagem define a estrutura e o significado do conteúdo que vemos nas páginas da internet. Imagine o HTML como os ossos de um corpo: ele dá forma e sustentação. Para entender melhor, podemos comparar uma página web a um livro. O HTML seria a estrutura do livro, definindo capítulos, parágrafos, títulos e imagens. Outras tecnologias, como o CSS, seriam responsáveis pela aparência do livro, como a fonte, as cores e o layout. Já o JavaScript adicionaria funcionalidades interativas, como menus suspensos ou animações. O termo "hipertexto" se refere à capacidade de conectar diferentes partes de um texto ou até mesmo páginas inteiras. Os links que você clica para navegar entre sites são exemplos de hipertexto em ação.

O **CSS** (Cascading Style Sheets ou Folhas de Estilo em Cascata) para EICH (2024), é como uma roupa para as páginas da internet. Ele define como os elementos de uma página, como títulos, parágrafos e imagens, serão exibidos na tela. Por exemplo, o CSS pode determinar a cor do texto, o tamanho das fontes, o espaçamento entre os elementos e até mesmo criar efeitos visuais como animações. Imagine uma página da web como um livro. O HTML (HyperText Markup Language) é como a estrutura do livro, definindo os capítulos, parágrafos e imagens. O CSS, por sua vez, é como a diagramação do livro, escolhendo as fontes, as cores e o layout das páginas. Para que serve o CSS? Estilização básica: Muda a cor e o tamanho dos títulos, links e outros elementos. Criação de layouts: Transforma uma simples coluna de texto em um layout mais complexo com colunas laterais e áreas de conteúdo. Efeitos visuais: Cria animações e outros efeitos para tornar a página mais interativa e interessante.

**JavaScript** para EICH (2024), é uma linguagem de programação que possibilita a criação de elementos sofisticados em sites. Sempre que uma página web vai além de exibir apenas informações estáticas — apresentando conteúdo que se renova periodicamente, mapas interativos ou gráficos animados em 2D/3D, por exemplo — é bem provável que o JavaScript esteja por trás disso. Essa linguagem faz parte da terceira camada da estrutura das tecnologias web, sendo que as duas primeiras (HTML e CSS) foram exploradas em mais profundidade em outras seções da Área de Aprendizado.

**PHP**, segundo CRISTO (2024), PHP: Hypertext Preprocessor, é uma

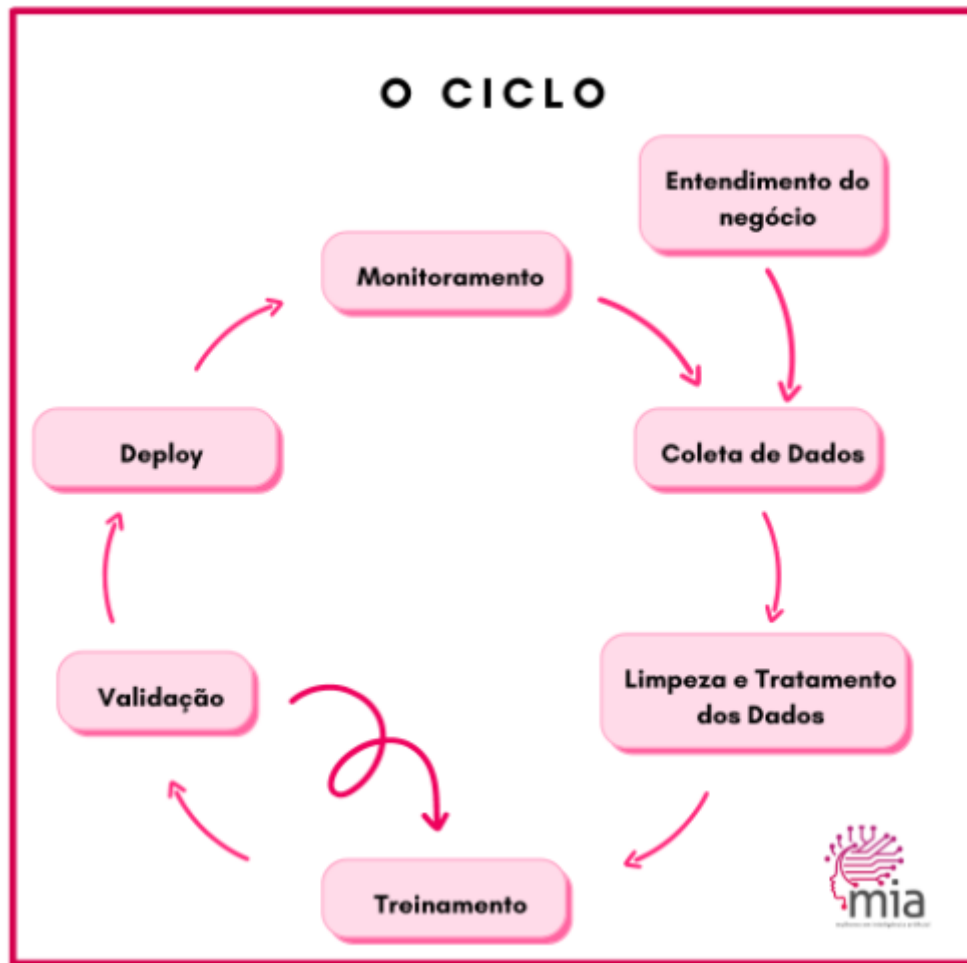
linguagem de programação de código aberto muito popular, amplamente utilizada e ideal para criar aplicativos web, podendo ser integrada diretamente ao HTML. Ao invés de utilizar diversos comandos para exibir HTML (como em C ou Perl), as páginas PHP apresentam HTML integrado ao código, realizando determinada ação (como, por exemplo, exibir a mensagem "Olá, eu sou um script PHP!"). O código PHP é identificado pelas tags de início e fim das instruções de processamento, que possibilitam a transição para dentro e fora do "modo PHP". A diferença entre o PHP e o JavaScript no lado do cliente é que o PHP é executado no servidor, produzindo o HTML que é enviado para o navegador. O navegador apenas recebe os resultados da execução do script, sem ter acesso ao código fonte original. Uma opção é configurar o servidor para processar todos os arquivos HTML com PHP, o que impede que os usuários saibam o que está por trás das páginas. Uma das vantagens de adotar o PHP é a sua facilidade de uso para quem está começando garantindo ao mesmo tempo uma gama de funcionalidades avançadas para os programadores mais experientes. Não é preciso sentir receio ao se deparar com a extensa lista de recursos disponíveis no PHP. Aproveite ao máximo, aprenda rapidamente e comece a desenvolver scripts básicos em questão de poucas horas.

**MySQL** para CIRNE (2024), é um sistema de gerenciamento de banco de dados relacional de código aberto. O software opera em um modelo arquitetônico, permitindo que vários clientes se conectem ao servidor e acessem e manipulem os dados armazenados no banco de dados. Os dados são organizados em tabelas que consistem em linhas e colunas. Usando Structured Query Language (SQL), você pode realizar operações de criação, modificação e consulta de dados. A ferramenta oferece uma ampla gama de recursos e funcionalidades, incluindo suporte a transações, indexação eficiente, controle de acesso e segurança de dados. Além disso, o MySQL é conhecido por sua estabilidade, confiabilidade e facilidade de uso. Possui uma grande comunidade de desenvolvedores e usuários em todo o mundo, o que o ajuda a ser continuamente aprimorado e suportado.

## 5 DOCUMENTAÇÃO DO PROJETO

Para MESQUITA (2024) a documentação de projetos desempenha um papel fundamental no sucesso e na eficiência dos empreendimentos. Esses acompanhamentos, baseados na gestão empresarial, são o divisor de águas entre uma companhia amadora ou profissional. Independentemente do tamanho ou da natureza do projeto, a documentação abrange o registro de todas as informações relevantes, desde o planejamento inicial até a conclusão. Por isso, é como se resumisse e reunisse todas as informações importantes dele. Para entender melhor o conceito de documentação de projetos, continue lendo. Conheça a sua importância e confira orientações sobre como realizar esta documentação de maneira eficiente.

O que é a documentação de projetos? A documentação de projetos refere-se ao processo de registrar e organizar todas as informações relacionadas a um projeto. Isso inclui objetivos, escopo, cronograma, recursos, requisitos, estratégias, riscos, decisões e resultados alcançados. O objetivo principal é criar um histórico completo e estruturado do projeto, facilitando a compreensão, a colaboração e a referência futura. Importância da documentação de projetos. A documentação de projetos é essencial para o sucesso das atividades da empresa, produtividade e mais. Uma documentação clara e detalhada permite que todas as partes interessadas compreendam o projeto de maneira consistente. Ela serve como uma fonte confiável de informações, facilitando a comunicação entre a equipe do projeto, os clientes, os patrocinadores e outros envolvidos. Assim, com uma documentação adequada, evitam-se mal-entendidos e aumenta-se a eficiência das interações entre as partes. Aprendizado organizacional. A documentação de projetos permite que as organizações acumulem conhecimento e experiência ao longo do tempo.



Fonte: Bueno (2024).

Para OLIVEIRA (2024) tem está ideia sobre o ciclo de vida: Entendimento do negócio essa é uma das partes fundamentais do projeto. Afinal, para criar um modelo, é necessário entender o seu objetivo e um pouco do contexto no qual nos encontramos. No Projeto A, precisamos conversar com a área de negócios para entender, por exemplo: Qual o fluxo de vida do cliente dentro do negócio; O que, na visão deles, pode impactar esse cancelamento; Se existem variáveis (features, em inglês) sazonais — por exemplo, um período de reajuste de preço — ou regionais (como bairros em que existe um melhor ou pior atendimento). Além disso, precisamos entender quando e como esse modelo poderia ser usado. Se a predição for feita com um mês de antecedência é suficiente? As ações tomadas estariam de acordo?

## **Coleta de dados**

Reunir os dados para criar a modelagem pode ser simples, bastando executar uma *query SQL* em uma base de dados. Mas, dependendo do negócio e da necessidade, pode ser uma tarefa bastante trabalhosa. Pode ser necessário capturar dados de fontes diversas e, em alguns casos, fazer raspagem de dados (web scraping), ou seja, ter que criar um script que faça download de dados da internet de maneira automatizada, como por exemplo, baixar tweets para criar um modelo que analise fake news. Então, tendo entendido quais as principais variáveis que precisamos para a criação do modelo, é hora de conversar com a equipe responsável pelos dados, que pode ser a equipe de Engenharia de Dados ou a própria equipe de Produtos, dependendo da estruturação da empresa.

## **Coleta de dados**

Reunir os dados para criar a modelagem pode ser simples, bastando executar uma *query SQL* em uma base de dados. Mas, dependendo do negócio e da necessidade, pode ser uma tarefa bastante trabalhosa. Pode ser necessário capturar dados de fontes diversas e, em alguns casos, fazer raspagem de dados (web scraping), ou seja, ter que criar um script que faça download de dados da internet de maneira automatizada, como por exemplo, baixar tweets para criar um modelo que analise fake news. Então, tendo entendido quais as principais variáveis que precisamos para a criação do modelo, é hora de conversar com a equipe responsável pelos dados, que pode ser a equipe de Engenharia de Dados ou a própria equipe de Produtos, dependendo da estruturação da empresa.

## **Treinamento**

Chegou a etapa do ciclo que leva toda a fama! Aqui, já com os dados organizados, limpos e tratados, os separamos em três grupos: treino, teste e validação. O treino e a validação são usados durante o treinamento, sendo o treino para treinar o algoritmo em si e a validação para testar os parâmetros do algoritmo. Devemos guardar os dados que ficaram em teste para a etapa seguinte. Existem outras formas de fazer essa separação entre treino e teste, como, por exemplo, com validação cruzada, separando os dados em vários folds. Tendo feito essa última organização, é hora de você brilhar com os seus conhecimentos de aprendizado da máquina. O ideal é escolher algoritmos para testes que se encaixem com o seu

problema, começando com os mais simples e, somente caso necessário, passando para algoritmos mais complexos.

### **Validação**

Agora, com o seu modelo treinado, está na hora de avaliar os resultados!

É importante ter os dados separados para teste para garantir que o seu modelo não está com overfitting, ou seja, que ele não tenha se super ajustado aos dados de treino. Por isso, testamos nos dados de teste e calculamos as métricas necessárias de acordo com o nosso problema. Se for um problema de classificação, como é o caso dos nossos dois projetos, as métricas normalmente são: curva ROC; área sob a curva ROC (AUC – do inglês “area under the curve”); acurácia; precisão; sensibilidade (ou recall). É importante notar qual métrica é mais relevante de acordo com o seu problema. Se estou tratando de uma classificação de um problema de saúde, a sensibilidade é mais importante, pois é melhor capturar a maior quantidade de doentes corretamente, mesmo que algumas pessoas saudáveis sejam incorretamente classificadas como doentes. Porém, eu também não posso ter uma precisão muito ruim, para que pessoas saudáveis não passem pelo stress de acharem que estão doentes. Por isso, é preciso definir um bom limiar (threshold) de probabilidade que dê o equilíbrio que estamos procurando.

### **Deploy**

O deploy consiste em deixar o seu modelo disponível para que outros serviços possam acessá-lo. Dessa forma, ele poderá de fato ser utilizado dentro do ecossistema da empresa ou diretamente pelos clientes do seu serviço. O deploy normalmente consiste na criação de uma API com um endpoint específico para a predição, mas também pode ser feito de algumas outras formas, como uma comunicação por mensageria ou até mesmo uma predição por batch em script.

### **Monitoramento**

Avaliar o seu modelo enquanto em produção é uma etapa ainda pouco valorizada. Os dados não são fixos, portanto uma mudança de comportamento pode alterar todos os resultados. Por isso, é importante monitorar o andamento e entender quando é hora de capturar novos dados e retreinar o seu modelo. Independentemente do projeto, essa etapa consiste em capturar os dados e as predições do modelo que estão em produção, e fazer as mesmas análises de validação do modelo. Se percebemos que o modelo está caindo em relação às métricas definidas — por



exemplo, acurácia — e está muito pior do que com os dados utilizados na validação, é hora de pegar mais dados e retreinar o modelo.

## **5.1 Requisitos**

Para HIGOR (2013), antigamente dizia-se que requisitos eram sinônimos de funções, ou seja, tudo que o software deveria fazer funcionalmente. No entanto, atualmente assumiu-se que requisitos de software é muito mais do que apenas funções. Requisitos são, além de funções, objetivos, propriedades, restrições que o sistema deve possuir para satisfazer contratos, padrões ou especificações de acordo com o(s) usuário(s). De forma mais geral um requisito é uma condição necessária para satisfazer um objetivo. Portanto, um requisito é um aspecto que o sistema proposto deve fazer ou uma restrição no desenvolvimento do sistema. Vale ressaltar que em ambos os casos devemos sempre contribuir para resolver os problemas do cliente e não o que o programador ou um arquiteto deseja. Dessa forma, o conjunto dos requisitos como um todo representa um acordo negociado entre todas as partes interessadas no sistema. Isso também não significa que o programador, arquiteto ou um analista bem entendido no assunto de tecnologia não possam contribuir com sugestões e propostas que levem em conta o desejo do cliente. Além disso, ainda temos um documento de requisitos que é uma coleção dos requisitos. Por fim, os requisitos possuem alguns objetivos centrais como estabelecer e manter uma concordância com os clientes e outros envolvidos sobre o que o sistema deve fazer, deve oferecer aos desenvolvedores, projetistas e testadores do sistema uma compreensão melhor dos requisitos do sistema, definir fronteiras do sistema definindo o que deve ser incluído e o que não deve fazer parte do sistema, fornecer uma base para estimar o custo e o tempo de desenvolvimento do sistema e por fim definir uma interface de usuário para o sistema.

### **5.1.1 Requisitos funcionais**

Os requisitos funcionais são aqueles que visam atingir a solução dos problemas do usuário. Desse modo, eles trabalham diretamente no objetivo para o qual uma solução foi escrita. Na engenharia de software, um requisito funcional visa atender as necessidades dos usuários, materializando estes conceitos em forma de funções no novo sistema. (DESENVOLVEDOR, 2024).

REQUISITOS FUNCIONAIS		
RF001	Cadastramento de usuário	Permite que o usuário faça o preenchimento dos seus dados para realizar o cadastramento.
RF002	Login de usuário	Permite que o login do usuário seja feito a partir do CPF e senha.
RF003	Pesquisa de produto	Permite o usuário procurar o produto por nome, ou outras especificações e também filtrar o conteúdo.
RF004	Catálogo de produto	Exibir um catálogo dos produtos separados por sessões.
RF005	Acesso administrador	Permite que o administrador insira, altere, e exclua informações de usuários e agendamentos.
RF006	Finalização de agendamento	Após o usuário preencher as informações do agendamento, ele deve preencher uma ficha médica, informando alergias, uso de medicamentos, etc, para a finalização do agendamento.
RF007	Carrinho de compras	Adicionar e remover itens; mostrar o valor da soma de todos os produtos do carrinho;
RF008	Chekout e Pagamento	Checkout simplificado em várias etapas;
RF009	Gestão de Pedidos	O usuário conseguira ver o andamento do pedido.
RF010	Avaliação e Comentários	O cliente poderá comentar e avaliar os produtos, mas com moderação.

Fonte: Bueno (2024).

### 5.1.2 Requisitos não funcionais

Segundo o site TRTPR (2024) Requisitos não-funcionais são os requisitos relacionados ao uso da aplicação em termos de desempenho, usabilidade, confiabilidade, segurança, disponibilidade, manutenibilidade e tecnologias envolvidas. Não é preciso o cliente dizer sobre eles, pois eles são características mínimas de um software de qualidade, ficando a cargo do desenvolvedor optar por atender esses

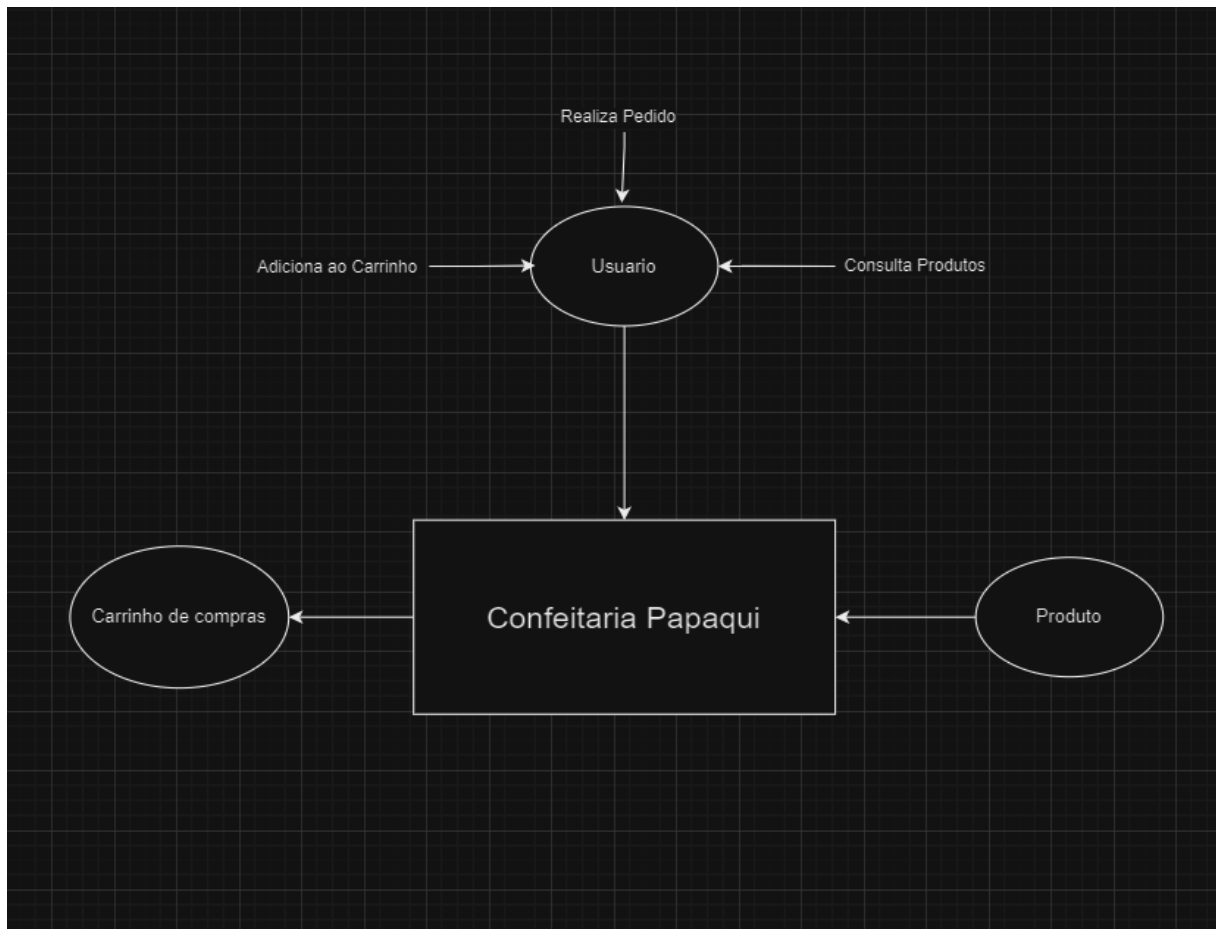
requisitos ou não.

REQUISITOS NÃO FUNCIONAIS		
<b>RNF001</b>	<b>Velocidade</b>	Que o usuário não precise esperar mais de 4 segundos para obter retorno.
<b>RNF002</b>	<b>Facilidade de uso</b>	O treinamento para utilização deve ser feito no máximo em 1 dia.
<b>RNF003</b>	<b>Confiabilidade</b>	Terá administrador tomando conta para caso ocorra falhas no sistema, e seja corrigido o mais breve possível.
<b>RNF004</b>	<b>Robustez</b>	O tempo para correção de falhas deve ocorrer em até 5 minutos.
<b>RNF005</b>	<b>Segurança</b>	O sistema deve colocar como exigência, que a senha precise de no mínimo 9 caracteres, sendo elas: 6 letras (entre elas, uma maiúscula), 1 caractere especial e 2 dígitos numéricos.
<b>RNF006</b>	<b>Compatibilidade</b>	O site deve ser compatível com os principais navegadores: Google Chrome, Mozilla Firefox, etc...

Fonte: Bueno, 2024

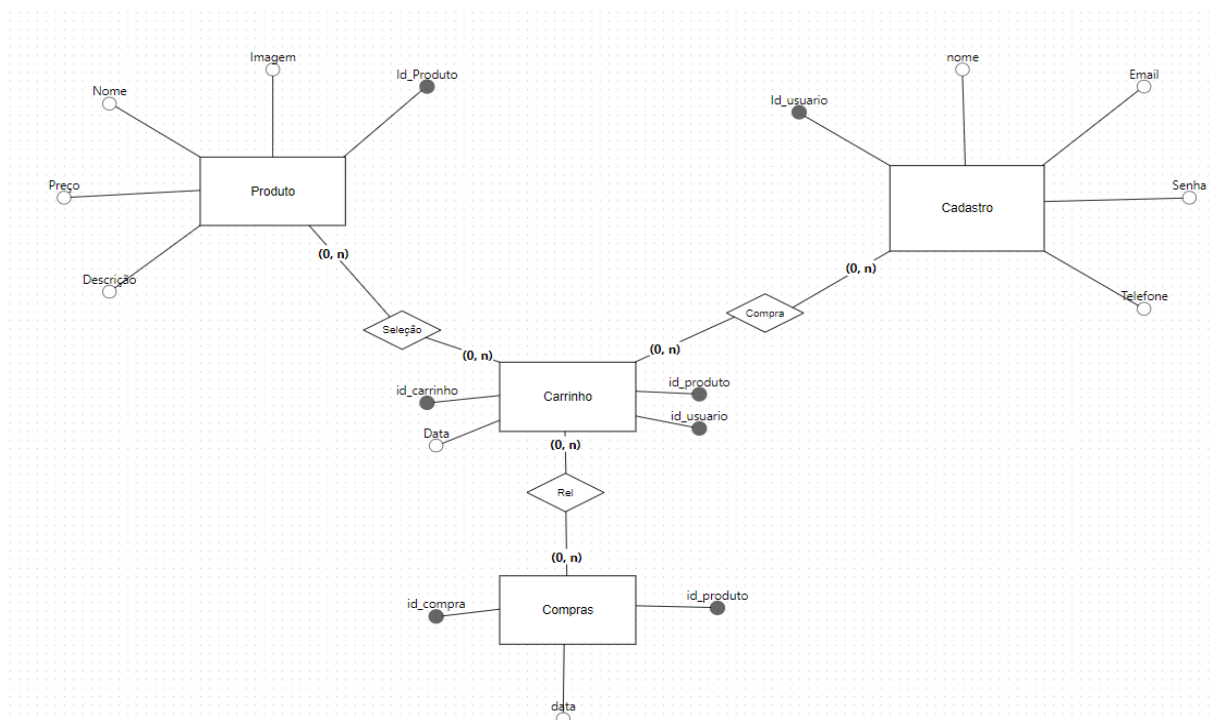
## 5.2 Diagrama de Contexto

O diagrama de contexto é uma representação de alto nível de um sistema, funcionando como um esboço inicial para definir o projeto, delimitando seu alcance e mostrando como ele se conecta a componentes externos, como stakeholders. Chamado também de diagrama de fluxo de dados de nível 0, ele destaca as interações do sistema com elementos externos, sem entrar nos processos internos, que costumam ser detalhados em diagramas de fluxo mais específicos.



Fonte: Bueno, 2024

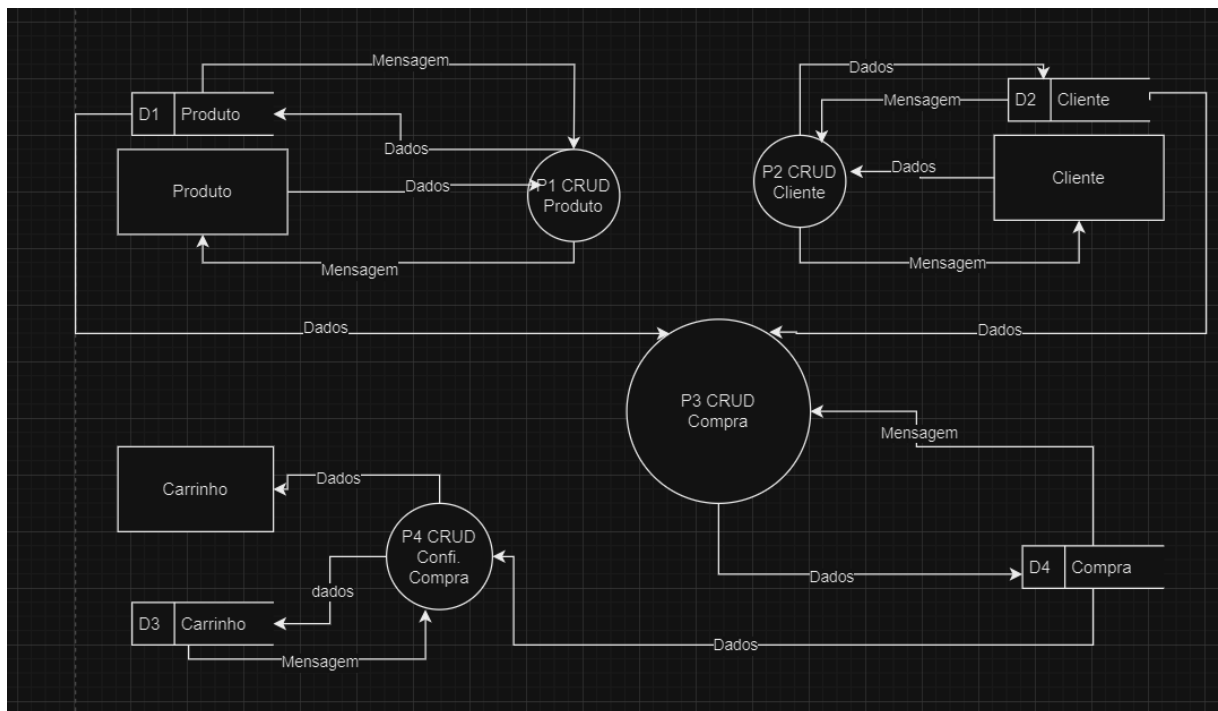
## MODELO CONCEITUAL



Fonte: Bueno, 2024

### 5.3 Diagrama de Fluxo de dados

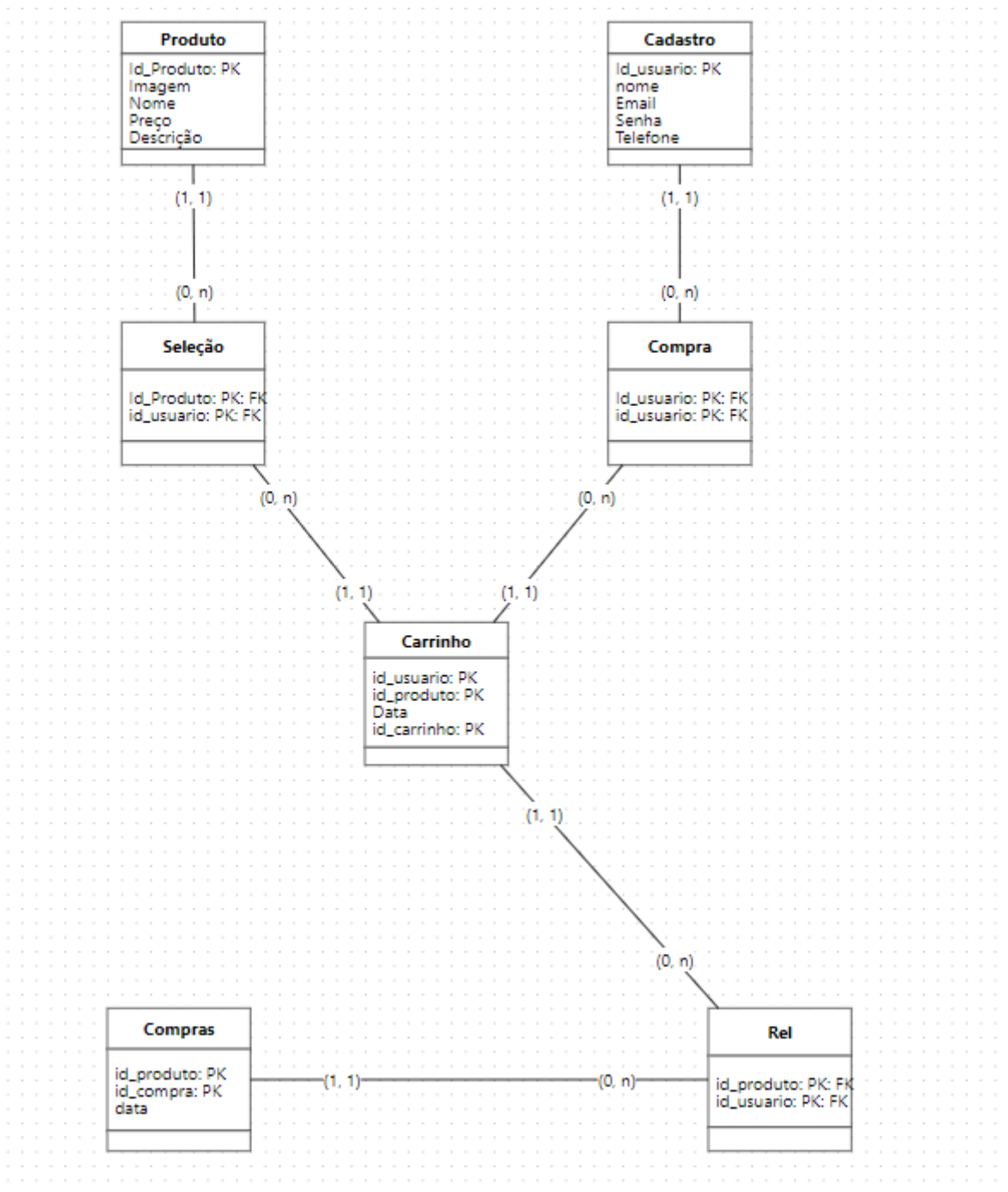
Para Khusid (2024), um diagrama de fluxo de dados mapeia a sequência de informações, atores e passos dentro de um processo ou sistema. Ele usa um conjunto de símbolos e formas padronizadas representando as diferentes etapas e pessoas que são necessárias para que um processo seja realizado corretamente. O diagrama pode ser tão simples ou tão complexo quanto o sistema que ele representa, mas a maneira mais fácil é usar uma ferramenta para fazer Diagramas de Fluxo de Dados Online. Normalmente, os diagramas de fluxo de dados são usados para representar visualmente o fluxo de dados de um sistema de uma empresa. Como mostrado no exemplo do diagrama de fluxo de dados abaixo, eles ilustram como os dados são processados dentro de um sistema usando um conjunto de entradas (inputs) e saídas (outputs).



Fonte: Bueno, 2024

#### **5.4 Diagrama de Entidade e relacionamento**

Conforme diz KHUSID (2024) um diagrama de entidade e relacionamento (também conhecido como diagrama ER ou simplesmente DER) mostra como as entidades (pessoas, objetos e conceitos) interagem. Estes modelos de dados conceituais ajudam desenvolvedores e designers a visualizar as relações entre os elementos-chave do software. Criado pelo cientista da computação Peter Chen nos anos 70, os diagramas ERs são comumente usadas por empresas ao projetar e analisar bancos de dados. Entretanto, os DERs também podem mapear relações entre outros elementos empresariais, tais como funções (como a relação de funções de um gerente de produto com um desenvolvedor), objetos empresariais tangíveis (como um produto ou serviço), e objetos empresariais intangíveis (como um backlog de produtos). O diagrama em si é um tipo de fluxograma. É um modelo lógico que mostra como os dados fluem de uma entidade para a outra. Com este formato fácil de seguir, os desenvolvedores de software e designers podem visualizar claramente a estrutura de um sistema.



Fonte: Bueno, 2024

## 5.5 Dicionário de Dados

### cadastros\_usuarios

Coluna	Tipo
id ( <i>Primária</i> )	int(11)
user_id	int(11)
full_name	varchar(100)
phone	varchar(15)
email	varchar(100)
password	varchar(255)
created_at	timestamp

#### Índices

Nome da chave	Tipo
PRIMARY	BTREE
user_id	BTREE

### users

Coluna	Tipo
id ( <i>Primária</i> )	int(11)
username	varchar(50)
password	varchar(255)
email	varchar(100)
created_at	timestamp

#### Índices

Nome da chave	Tipo
PRIMARY	BTREE

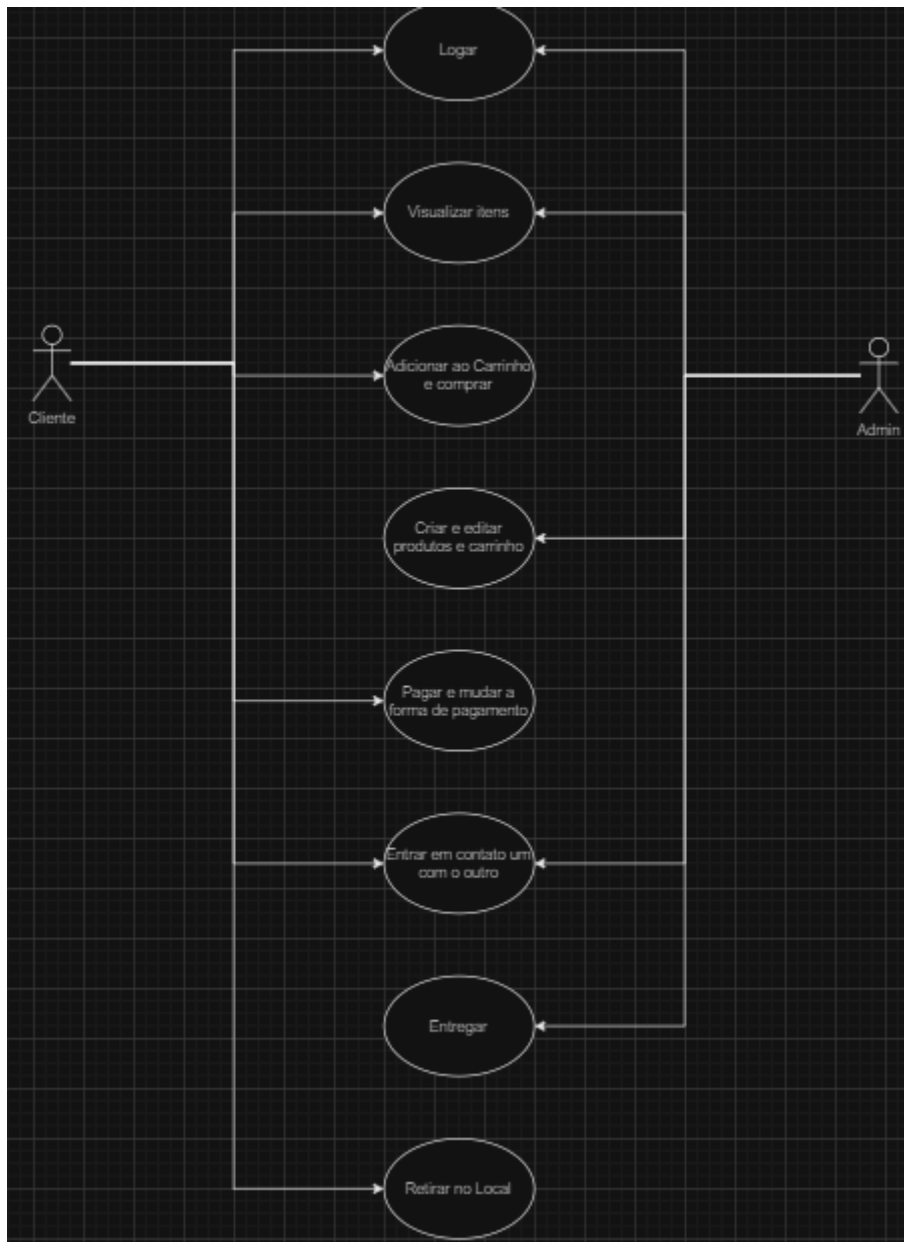
Fonte: Bueno, 2024



## 5.6 Diagrama de Caso de Uso

Para MATOS (2012), esse diagrama documenta o que o sistema faz do ponto de vista do usuário. Em outras palavras, ele descreve as principais funcionalidades do sistema e a interação dessas funcionalidades com os usuários do mesmo sistema. Nesse diagrama não nos aprofundamos em detalhes técnicos que dizem como o sistema faz.

Este artefato é comumente derivado da especificação de requisitos, que por sua vez não faz parte da UML. Pode ser utilizado também para criar o documento de requisitos.



#### Visualizar Produtos:

O cliente acessa a página inicial e visualiza os produtos disponíveis, com imagens, descrição, preço e ingredientes principais.

#### Adicionar ao Carrinho:

O cliente seleciona um produto e o adiciona ao carrinho de compras.

Opcional: Adiciona observações como preferências de sabor, cor, etc.

#### Finalizar Pedido:

O cliente revisa os itens no carrinho e preenche o formulário com informações de contato.

Escolhe a forma de pagamento (simples ou parcelado).

Envia o pedido, que é enviado por e-mail ou WhatsApp para a confeitaria.

#### Gerenciar Produtos (Admin):

O administrador faz login no painel de controle e adiciona/edita/remove produtos, definindo preço, descrição e estoque.

#### Gerenciar Pedidos (Admin):

O administrador visualiza os pedidos realizados, confirma e informa o cliente sobre o tempo de preparo.

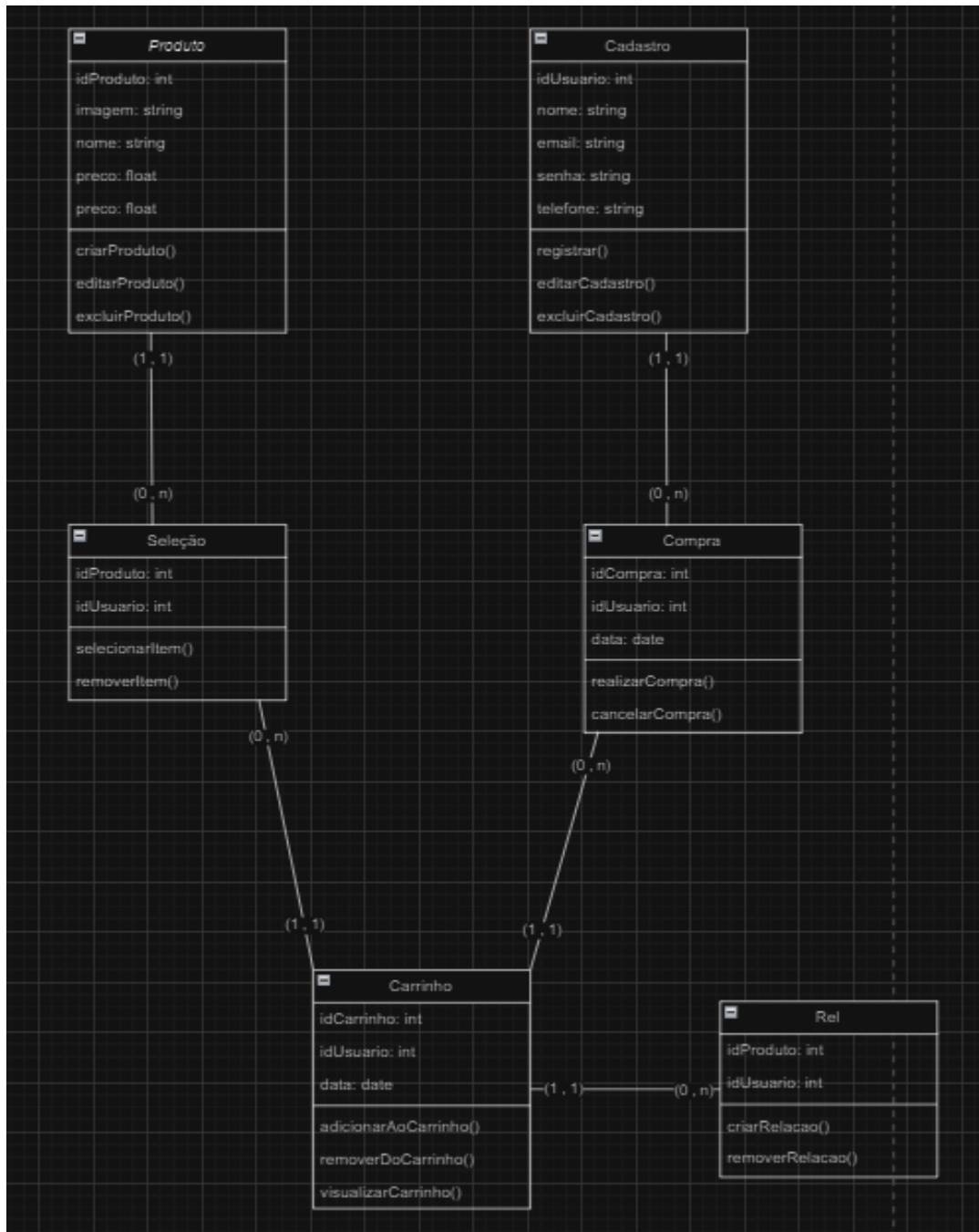
#### Consultar Informações de Contato:

O cliente acessa a página de "Contato" para entrar em contato direto com a confeitaria via WhatsApp ou e-mail.

**Fonte: Bueno, 2024**

## 5.7 Diagrama de Classe

Segundo MATOS (2016) na programação, um diagrama de classes é uma representação da estrutura e relações das classes que servem de modelo para objetos. Podemos afirmar de maneira mais simples que seria um conjunto de objetos com as mesmas características, assim saberemos identificar objetos e agrupá-los, de forma a encontrar suas respectivas classes. Na Unified Modeling Language (UML) em diagrama de classe, uma classe é representada por um retângulo com três divisões, são elas: O nome da classe, seus atributos e por fim os métodos.



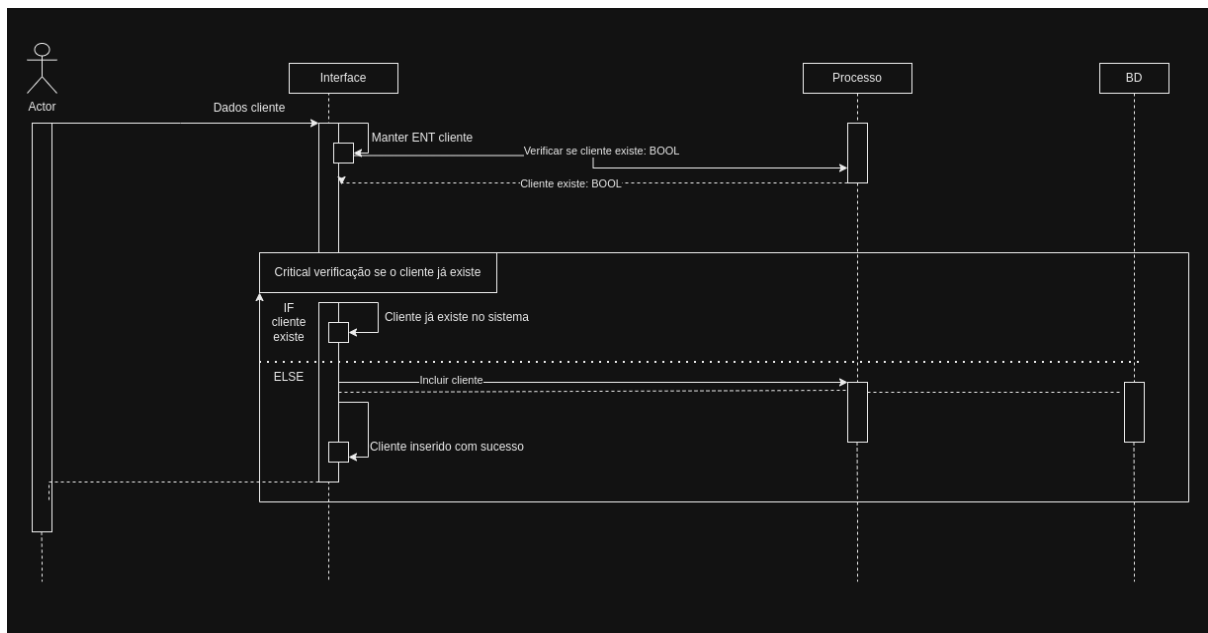
Fonte: Bueno, 2024

## 5.8 Diagrama de Sequência

Para MATOS (2009) o diagrama de sequência preocupa-se com a ordem temporal em que as mensagens são trocadas entre os objetos envolvidos em determinado processo, ou seja, quais condições devem ser satisfeitas e quais métodos devem ser disparados entre os objetos envolvidos e em que ordem durante

um processo. Dessa forma, determinar a ordem em que os eventos ocorrem, as mensagens que são enviadas, os métodos que são chamados e como os objetos interagem entre si dentro de um determinado processo é o principal objetivo deste diagrama.

O diagrama de sequência baseia-se no diagrama de casos de uso. É importante deixarmos claro que o fato de existir apenas um diagrama de casos de uso não implica na existência de apenas um único diagrama de sequência. Segundo Guedes em seu livro “UML – Uma Abordagem Prática”, normalmente existem diversos diagramas de sequência em um projeto, um para cada processo específico do sistema



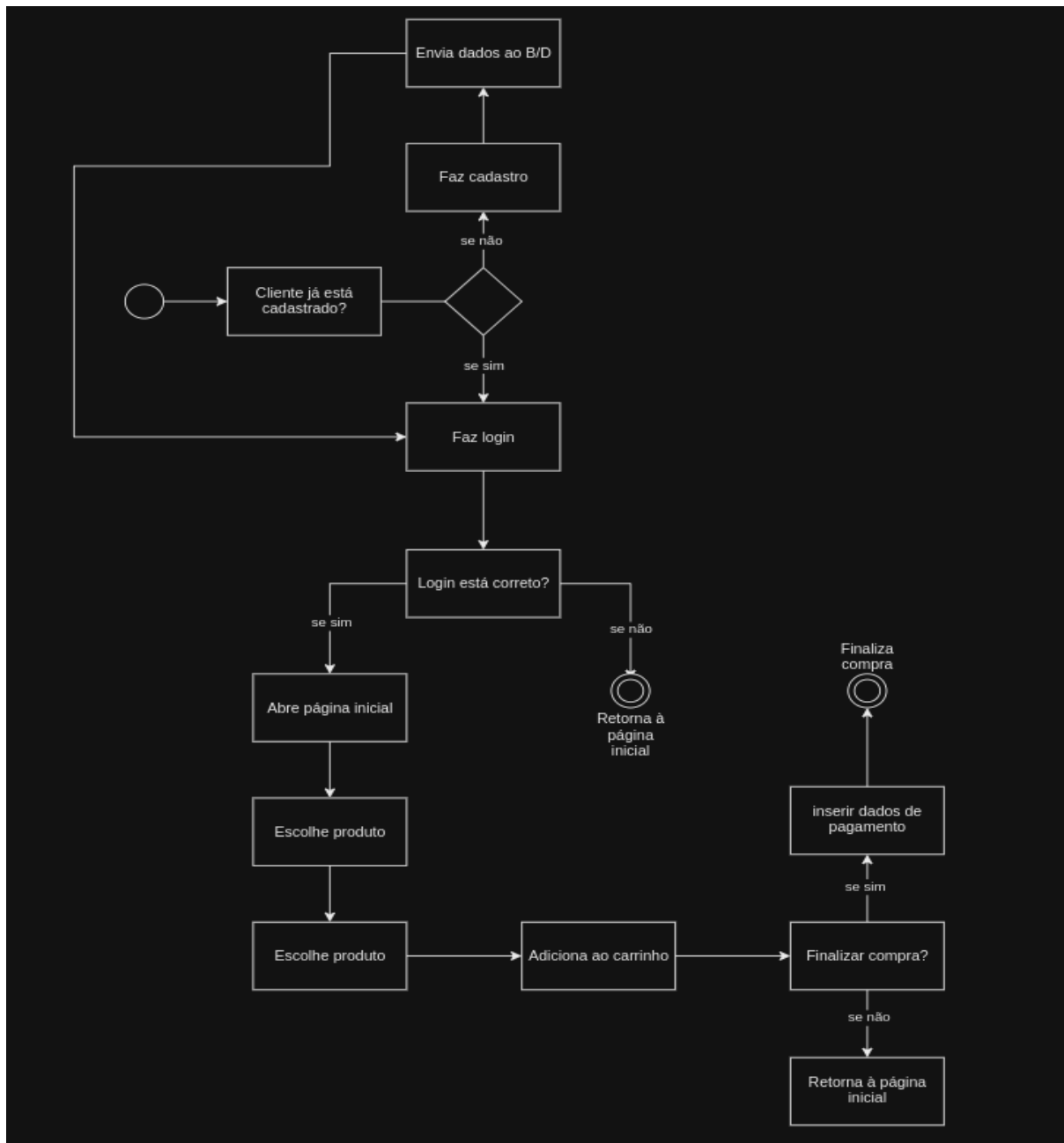
Fonte: Bueno, 2024

## 5.9 Diagrama de Atividade

Segundo a citação de MATOS (2010) aparentados com os diagramas de transição de estados, diagramas de atividades compartilham com aqueles uma parte significativa da notação. Estados de ação e estados de atividade são representados por um retângulo de cantos arredondados. De forma coerente com os diagramas de estados, ações são atômicas e atividades são conjuntos de ações e/ou atividades

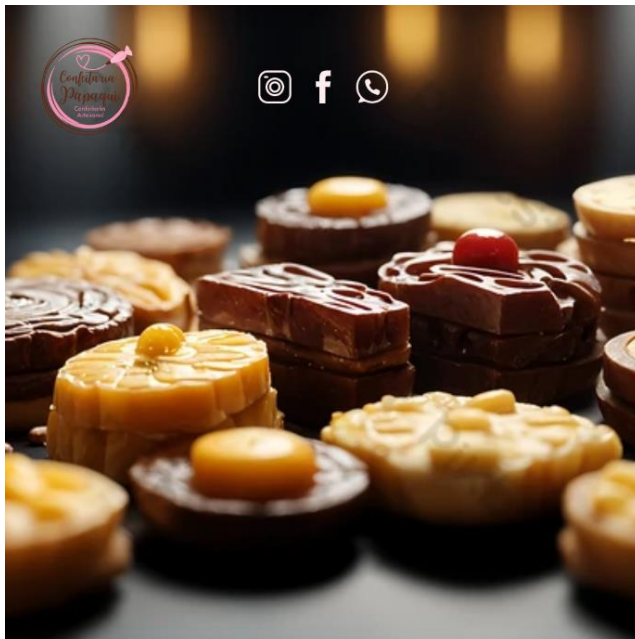
encadeadas.

Um diagrama de atividades pode constituir uma grande atividade (não atômica) formada por diversas ações (atômicas) e/ou sub-atividades. Entre duas ações e/ou atividades, uma seta simboliza simplesmente o término de uma atividade e o início da outra. Em princípio não é necessário rotular esse elemento, mas uma condição de guarda poderá estar eventualmente associada a ele. Os símbolos de início e final são idênticos àqueles utilizados em diagramas de estados.



Fonte: Bueno, 2024

## 6 TELAS



### Registrar

Nome:

Telefone:

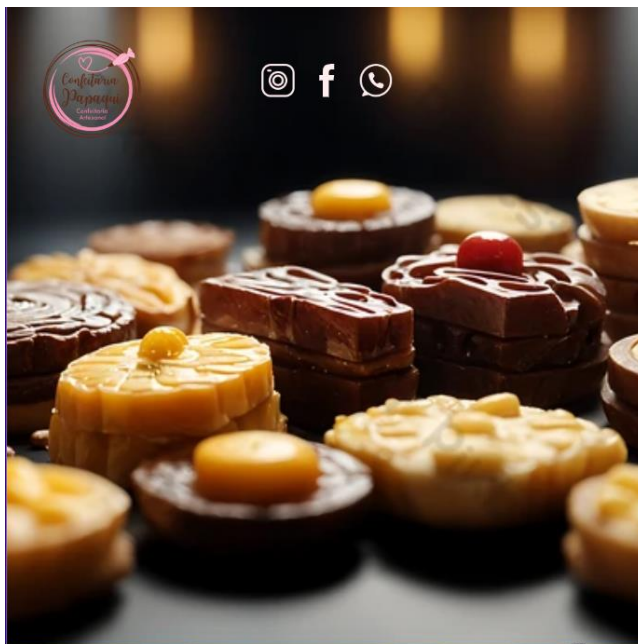
Email:

Senha:

Confirmar senha:

Entrar

Já tem cadastro? Clique em [Login](#)



### Login

Email:

Senha:



Entrar

Manter logado ☐

Não tem cadastro? Clique em [Cadastrar-se](#)



Siga-nos:



Confeitaria  
Papauki

 Carrinho:  
00,00

Início | Vitrine | Promoções

 Buscar...

Categorias | Localização

Anuncio

Anuncio

Anuncio

Bolos:



bolo de Morango  
R\$ 39,90



bolo de Morango  
R\$ 39,90



bolo de Morango  
R\$ 39,90



bolo de Morango  
R\$ 39,90



bolo de Morango  
R\$ 39,90

Doces:



Cento beijinho  
R\$ 39,90



Cento Brigadeiro  
R\$ 39,90



Cento casadinho  
R\$ 39,90



Cento Romeu e Julieta  
R\$ 39,90




Cento Olho de sogra  
R\$ 39,90



marketing • info • suporte

Termos de uso • Política de Privacidade

© 2024 Lacob



Escolher imagem

Nome do Produto:

digite aqui...

ID Produto:

ID adicionado automaticamente

Preço do produto:

R\$...

Ingredientes chaves:

+

Descricao do produto:

Digite aqui...

Promocoes

Salvar Postar Apagar Sair

### Carrinho:

	Nome do item	— Qtd: 1 +
	Preço: xxx,xx	
	Selecionar <input type="checkbox"/>	
	Nome do item	— Qtd: 1 +
	Preço: xxx,xx	
	Selecionar <input type="checkbox"/>	
	Nome do item	— Qtd: 1 +
	Preço: xxx,xx	
	Selecionar <input type="checkbox"/>	
<b>Valor total: R\$: XXX,XX</b>		



## 7 CONCLUSÃO

Este projeto teve como objetivo principal desenvolver um sistema eficiente para a gestão de produtos e compras, integrando funcionalidades como cadastro de usuários, seleção de itens, carrinho de compras e finalização de pedidos. A ideia surgiu da necessidade de atender demandas reais, como a criação de uma plataforma funcional para apoiar pequenos negócios, promovendo organização e acessibilidade.

Durante o desenvolvimento, abordamos desde a modelagem do banco de dados até a implementação prática das funcionalidades. Foram utilizados conceitos de normalização, relacionamentos entre tabelas e aplicação de princípios de programação orientada a objetos. Cada etapa trouxe desafios importantes, como a construção de tabelas intermediárias e a definição de regras de negócio para manter a integridade dos dados, mas também resultou em aprendizados que enriqueceram minha trajetória como programador.

Os resultados obtidos confirmaram a viabilidade do projeto e mostraram que é possível, com dedicação e planejamento, criar uma solução que atenda às necessidades iniciais. O sistema desenvolvido possibilita não apenas uma gestão simplificada de produtos e usuários, mas também abre portas para melhorias futuras, como a integração de métodos de pagamento e análises de vendas.

As implicações deste projeto são significativas para o campo de estudo e para a prática. Ele reflete como tecnologias acessíveis e bem planejadas podem ser ferramentas poderosas para digitalizar processos e melhorar a experiência dos usuários. Além disso, o trabalho desenvolvido reforça a importância de unir teoria e prática para resolver problemas reais.

Para o futuro, penso em expandir as funcionalidades deste sistema, incluindo relatórios avançados de vendas, integração com redes sociais para divulgação e suporte a múltiplos idiomas, ampliando sua aplicabilidade. Esses são caminhos promissores que podem fortalecer ainda mais o impacto positivo desta solução.

Agradeço imensamente a todos que contribuíram para o sucesso deste projeto: professores, colegas e familiares que ofereceram apoio e incentivo em cada etapa. Este trabalho não seria possível sem a dedicação e o esforço coletivo.

Finalizo com a convicção de que este projeto é apenas o começo de uma jornada. Ele representa não só o conhecimento adquirido, mas também a vontade de continuar aprendendo, criando e inovando. Espero que esta iniciativa inspire outros a

também buscarem soluções criativas para os desafios do dia a dia.

## 8 REFERÊNCIAS

EICH, Brendan. HTML: Linguagem de Marcação de Hipertexto. 2024. Disponível em: <https://developer.mozilla.org/pt-BR/docs/Web/HTML>. Acesso em: 20 set. 2024.

EICH, Brendan. O que é CSS? . 2024. Disponível em: [https://developer.mozilla.org/pt-BR/docs/Learn/CSS/First\\_steps/What\\_is\\_CSS](https://developer.mozilla.org/pt-BR/docs/Learn/CSS/First_steps/What_is_CSS). Acesso em: 20 set. 2024.

EICH, Brendan. O que é JavaScript?. 2024. Disponível em: [https://developer.mozilla.org/pt-BR/docs/Learn/JavaScript/First\\_steps/What\\_is\\_JavaScript](https://developer.mozilla.org/pt-BR/docs/Learn/JavaScript/First_steps/What_is_JavaScript). Acesso em: 20 set. 2024.

CRISTO, Adriel. O que é o PHP? 2024. Disponível em: [https://www.php.net/manual/pt\\_BR/intro-what.php](https://www.php.net/manual/pt_BR/intro-what.php). Acesso em: 20 set. 2024. LOCABWEB,

CIRNE, Fernando. MySQL: guia completo de como utilizar o sistema. 2024. disponível em: <https://www.locaweb.com.br/blog/temas/codigo-aberto/mysql-guia-completo-de-como-utilizar-o-sistema>. Acesso em: 20 set. 2024.

MESQUITA, Aline. Documentação de projetos: o que é, por que e como fazer? 2024. Disponível em: <https://www.oitchau.com.br/blog/documentacao-de->

OLIVEIRA, Jéssica dos Santos de. **O ciclo de vida de um projeto de dados**. 2022. Disponível em: <https://www.programaria.org/o-ciclo-de-vida-de-um-projeto-de-dados/>. Acesso em: 07 out. 2024.

HIGOR. **Introdução a Requisitos de Software**: requisitos de software. Requisitos de Software. 2013. Disponível em: <https://www.devmedia.com.br/introducao-a-requisitos-de-software/29580>. Acesso em: 21 out. 2024

TRTPR. **Conceito: Requisitos Não-Funcionais**: definição. Definição. 2024. Disponível em: [https://www.trt9.jus.br/pds/pdstrt9/guidances/concepts/supporting\\_requirements\\_B2C4D610.html](https://www.trt9.jus.br/pds/pdstrt9/guidances/concepts/supporting_requirements_B2C4D610.html). Acesso em: 23 out. 2024.

KHUSID, Andrey. **Diagrama de fluxo de dados**: o que é um diagrama de fluxo de dados?. O que é um diagrama de fluxo de dados?. 2024. Disponível em: <https://miro.com/pt/diagrama/o-que-e-diagrama-fluxo-dados/>. Acesso em: 25 out. 2024.

KHUSID, Andrey. **Diagrama de Entidade e Relacionamento**: o que é um diagrama de entidade e relacionamento (er)?. O que é um diagrama de entidade e relacionamento (ER)?. 2024. Disponível em: <https://miro.com/pt/diagrama/o-que-e-diagrama-entidade-relacionamento/>. Acesso em: 25 out. 2024.

Montagne, Prosper. 'Larousse Gastronomique'. Éditions Larousse, 1938.

Roux, Michel. 'Pastry: Savory and Sweet'. Quadrille Publishing, 2007.

(GEMINI), Openia. Na era digital. 2021. Disponível em: <https://mindthegraph.com/blog/pt/chatgpt-citacoes/>. Acesso em: 08 ago. 2024.

Ude, Johann. 'The Art of Pastry Making'. Reprint by Forgotten Books, 2017.

Matos, Gladstone O que é UML e Diagramas de Caso de Uso: Introdução Prática à UML: diagrama de casos de uso. Diagrama de Casos de Uso. 2012. Disponível em: <https://www.devmedia.com.br/o-que-e-uml-e-diagramas-de-caso-de-uso-introducao-pratica-a-uml/23408>. Acesso em: 25 nov. 2024.

Matos, Gladstone. Orientações básicas na elaboração de um diagrama de classes: o que são os diagramas de classe. O que são os Diagramas de Classe. 2016. Disponível em: <https://www.devmedia.com.br/orientacoes-basicas-na-elaboracao-de-um-diagrama-de-classes/37224>. Acesso em: 25 nov. 2024.

Matos, Gladstone Artigo SQL Magazine 64 - Utilizando UML. 2009. Disponível em: <https://www.devmedia.com.br/artigo-sql-magazine-64-utilizando-uml/12665>. Acesso em: 25 nov. 2024.

Matos, Gladstone Diagramas de Atividades - Engenharia de Software 31. Definições básicas e notação. 2010. Disponível em: <https://www.devmedia.com.br/diagramas-de-atividades-engenharia-de-software-31/18744>. Acesso em: 25 nov. 2024.