

**CENTRO PAULA SOUZA
FACULDADE DE TECNOLOGIA DE FRANCA
“DR. THOMAZ NOVELINO”**

TECNOLOGIA EM ANÁLISE E DESENVOLVIMENTO DE SISTEMAS

**CAMILY ALEIXO RANDI
LIDIANE BARBOSA DA SILVA**

**SOLUÇÃO SISTÊMICA PARA GERENCIAMENTO DE UMA BUTIQUE
DE CARNES**

Trabalho de Graduação apresentado à Faculdade de Tecnologia de Franca - “Dr. Thomaz Novelino”, como parte dos requisitos obrigatórios para obtenção do título de Tecnólogo em Análise e Desenvolvimento de Sistemas.

Orientador: Prof. Me. Carlos Alberto Lucas

FRANCA/SP

2023

SOLUÇÃO SISTÊMICA PARA GERENCIAMENTO DE UMA BUTIQUE DE CARNES

Camilly Aleixo Randi¹

Lidiane Barbosa da Silva²

Resumo

As lojas especializadas em carnes são famosas por oferecerem uma grande variedade de produtos e excelente qualidade aos clientes. É indispensável que essas butiques busquem uma gestão eficiente das suas atividades cotidianas, tais como o controle de estoque, vendas e relatórios. Como forma de resolver esse problema, emerge a necessidade do desenvolvimento de um Sistema de Gestão Integrada, e este é um dos objetivos deste Trabalho, qual seja, o desenvolvimento de uma solução sistêmica para butiques de carnes. Os outros objetivos são o de proporcionar uma maior eficiência, organização e qualidade nos serviços oferecidos através do aprimoramento dos processos administrativos. Os resultados conquistados com esta solução incluem a maior eficiência nos processos diários e um controle de estoque mais aprimorado. Esta gestão integrada ainda contribui na melhoria contínua do atendimento ao cliente, agregando valor ao negócio e gerando satisfação constante.

Palavras-chave: Atendimento ao cliente. Butique de Carnes. Eficiência. Qualidade. Organização. Gestão Integrada.

Abstract

Stores specializing in meat are famous for offering a wide variety of products and excellent quality to customers. It is essential that these boutiques seek efficient management of their daily activities, such as inventory control, sales and reports. As a way to solve this problem, the need to develop an Integrated Management System emerges, and this is one of the objectives of this Work, that is, the development of a systemic solution for meat boutiques. The other objectives are to provide greater efficiency, organization and quality in the services offered through the improvement of administrative processes. The results achieved with this solution include greater efficiency in daily processes and improved inventory control. This integrated management also contributes to the continuous improvement of customer service, adding value to the business and generating constant satisfaction.

Keywords: Customer service. Efficiency. Meat Boutique. Quality. Organization.

¹ Graduando em Análise e Desenvolvimento de Sistemas pela Fatec Dr Thomaz Novelino – Franca/SP. Endereço eletrônico: camilly.randi@fatec.sp.gov.br

² Graduando em Análise e Desenvolvimento de Sistemas pela Fatec Dr Thomaz Novelino – Franca/SP. Endereço eletrônico: lidiane.silva19@fatec.sp.gov.br

1 Introdução

O constante consumo de carne pela população, a situação econômica da maioria das famílias, o critério de ter uma carne de baixo preço e boa qualidade têm sido cada vez mais exigido no mercado. Trazendo assim vários fatores, como preço, qualidade, aspectos nutricionais, preferências e sabor.

Diante de uma pandemia mundial, o setor comercial passou por uma mudança drástica adequando aos novos tipos de tecnologia e a diminuição do contato humano no ato da venda. Um dos principais avanços tecnológicos na época da pandemia foi a adoção de aplicativos de vendas e entregas (*delivery*) como parte natural da nossa vida em sociedade. É fato que eles já eram muito populares antes, mas se tornaram absolutamente vitais durante a pandemia.

Nos últimos anos houve um aumento de 103% de consumo nesses aplicativos, uma forma mais segura de trazer os produtos necessários, como os alimentos, para casa em meio ao lockdown.

Conforme a pesquisa "Consumo Online no Brasil," conduzida pela agência Edelman e promovida pela empresa PayPal, o hábito de realizar pedidos de refeições pela internet ou por meio de aplicativos apresentou um aumento significativo durante a pandemia da COVID-19.

O processo de operação de uma butique de carne envolve várias etapas essenciais, desde a aquisição da carne até o atendimento ao cliente. Controlar o estoque é um processo essencial, juntamente com a promoção do negócio para atrair novos clientes por meio de campanhas de *marketing*, descontos ou programas de fidelidade.

Observando esse mercado surgiu a ideia de desenvolver uma solução sistêmica com o objetivo de contribuir com tais processos, facilitando o gerenciamento das compras, estoque e as entregas de uma butique de carnes.

O ponto de destaque do sistema buscará atender e permitir um maior direcionamento de tomada de decisão pelas lideranças, seja em questões técnicas ou voltadas para o monitoramento da maneira como os dados são passados, tudo levando em conta a facilidade de compreensão e retorno exato do que foi solicitado.

Através dos artefatos de engenharia de software, será construída toda a documentação necessária para demonstrar a proposta descrita, permitindo o acompanhamento dos artefatos em criação, gerenciamento e controle geral.

Conseqüentemente, faz-se necessário dividir o projeto em dois escopos de desenvolvimento, sendo o front-end e back-end, com aplicação de conceitos de prototipação de tela, design patterns e componentização de elementos, mesmo que futuramente este possa ser escalável e utilizado em outras soluções.

Por fim, algumas ferramentas serão importantes na implementação e criação desta solução, que proporcionarão atributos como performance, qualidade e confiabilidade. Estas ferramentas são: React, Node.js, TypeScript, PostgreSQL e Prisma.

1.1 Termo de Abertura do Projeto (TAP)

O objetivo deste documento é registrar as informações levantadas na fase inicial, documentando formalmente a necessidade de organizar as vendas online de uma butique de carne. Propondo uma resolução sistêmica desse problema e descrevendo as etapas necessárias para que esse projeto seja construído.

A etapa de conhecimento do projeto e a equipe, foram realizadas pelos seguintes integrantes: Lidiane Barbosa {Gerente do Projeto (EAP, TAP, BPMN, Regras de Negócio, Revisão, Monitoramento, Supervisão de etapas, Integração e Organização de Prazos)}; Camilly Aleixo {Integração, escopo, Risco, Interface do Sistema e Projetos, Tempo de desenvolvimento, Custos.}

Justificativa do projeto: Elaborar um software para auxiliar e organizar as rotinas das compras de carnes feitas online. Melhorando assim, a experiência do cliente, facilitando os processos e o controle da empresa em relação ao seu estoque, vendas e entregas.

Objetivos e metas do projeto: Objetivo é documentar e elaborar um aplicativo para gerenciamento de vendas online, entregas de pedidos, qualidade dos produtos e experiência do cliente. O sistema também visa melhorar a comunicação interna dos seus colaboradores. Tendo em mente os seguintes focos: Ter controle do estoque; manter a qualidade dos produtos; ter rastreabilidade dos pedidos; melhorar a jornada de trabalho dos colaboradores; ter um plano de fidelização dos clientes.

Descrição do projeto: O aplicativo de vendas online é desenvolvido em etapas, que são divididas em outros tópicos. O primeiro passo foi montar a equipe, escolher um gestor, definir o cliente e os prazos. Na segunda etapa, as respostas são coletadas e analisadas. O terceiro começa com a preparação *SWOT*, *5W2H*, *EAP*, *TAP*, *BPMN*.

No quarto passo foi feita a elicitación dos requisitos. Já na quinta etapa, começamos o desenvolvimento do caso de uso, o diagrama de atividade, de máquina, de estados, os diagramas de sequência e por fim matriz de rastreabilidade. A etapa final é a conclusão do projeto, ou seja, apresentação de documentos, protótipos e a proposta comercial e o desenvolvimento do código com integração no banco de dados.

Restrições do projeto: As restrições a serem consideradas no desenvolvimento do projeto são: Os prazos de entregas planejadas que não devem ser ultrapassadas; nenhum membro da equipe deve ser sobrecarregado.

Stakeholders do projeto: Empresário da Butique de Carne analisada e de outras empresas do segmento que estiverem interessados no produto.

Riscos do projeto: Os riscos do projeto podem ser separados em duas categorias: risco interno e externo. Os riscos internos dizem respeito aos acontecimentos como o desentendimento entre os integrantes da equipe, baixa produtividade em relação ao esperado e planejado e uma entrega com falta de qualidade. Já os riscos externos, são ocorrências como a falta de compromisso dos integrantes da equipe, problemas com os equipamentos necessários para a realização do projeto, falta de conhecimento do segmento ou dos processos imprescindíveis da área, devida à falta de informações advindas da empresa, por exemplo.

Entretanto, algumas medidas podem ser tomadas para mitigar esses riscos: manter todos os integrantes da equipe sempre informados sobre o projeto e quaisquer mudanças que venham a acontecer, buscando sempre a aprovação dos colaboradores; em caso de imprevisto ou obstáculo, deve ser comunicado aos integrantes, para que se encontre uma solução o mais rápido possível; o gerente do grupo deve estar sempre atento ao andamento do projeto e ter autoridade para tomar medidas corretivas quando for necessário, pensando sempre no bem coletivo, assim como todos os indivíduos devem respeitar aquele que atua no papel de gerente.

Prazos estimados do projeto: Data do início do projeto: 08/2022; Data da conclusão do projeto: 11/2023.

Marco Principal: O projeto se inicia em agosto de 2022. Nos reunimos para elaborar e obter conhecimento para desenvolver o projeto:

- Durante o mês de setembro/2022, organizamos encontros para desenvolver um questionário para levantar as necessidades da empresa;

- Em outubro/2022 começamos a desenvolver o BPMN, Caso de Uso, os diagramas de estado, atividade e sequência;
- Nas reuniões de novembro/2022, trabalhamos em cima de matrizes, rastreabilidade, métricas e por fim prototipação de telas;
- A partir de janeiro/2023 começamos o desenvolvimento do software e integrações com o banco de dados;
- Em outubro/2023 finalização de algumas telas desta solução.

2 Viabilidade do Projeto

A questão-chave da viabilidade é se o projeto pode ser financeiramente sustentável e lucrativo. A análise financeira pode incluir projeções de fluxo de caixa, cálculos de retorno sobre o investimento (ROI) e estimativas de custos operacionais.

Na área dos negócios, analisar cuidadosamente a viabilidade desempenha um papel crucial na tomada de decisões embasadas em relação à alocação eficiente de recursos. Destina-se a ajudar na mitigação dos riscos e no aumento das chances de alcançar êxito.

O *Business Model Canvas*, uma ferramenta de planejamento estratégico, foi concebido para desmembrar uma ideia de negócio em diversos componentes, permitindo uma visão completa do modelo de negócios.

A abordagem visual do Canvas, influenciada pelo design thinking, o torna intuitivo e fácil de compreender à primeira vista. Inicialmente proposto por Alex Osterwalder em sua tese de doutorado, o Canvas evoluiu por meio da colaboração de mais de 200 consultores globais.

Seus nove campos representam os principais pilares de um negócio, preenchidos com anotações que proporcionam uma visão holística e flexível do modelo, revelando diferenciais competitivos e possíveis melhorias para aumentar a eficiência (MINAS, 2023).

É importante destacar a diferença fundamental entre um modelo de negócios e um plano de negócios. O modelo de negócios concentra-se na criação, entrega e captura de valor no mercado, oferecendo uma visão estruturada e simplificada do funcionamento da empresa.

Por outro lado, o plano de negócios é um documento mais abrangente, reunindo todas as informações da organização, sendo crucial para obter investimentos,

formalizar a empresa e executar um planejamento detalhado (MINAS, 2023).

Figura 1 - Business Model Canvas

Segmentos	Proposição de Valor	Relações com o Consumidor	Principais Atividades	Principais Recursos
Boutique de Carnes	<ol style="list-style-type: none"> Qualidade e Variedade de Carnes: A boutique de carne pode se destacar por oferecer uma seleção de carnes de alta qualidade e variedade. Atendimento Personalizado: A equipe da boutique de carne pode fornecer atendimento personalizado, ajudando os clientes a escolher os cortes ideais de carne. Conveniência e Praticidade: Se a boutique de carne oferecer serviços como cortes personalizados, embalagens a vácuo ou marinadas pré-preparadas. Promoções e Programas de Fidelidade: Ofertas especiais, descontos e programas de fidelidade podem criar valor adicional para os clientes, incentivando-os a retornar. 	<ol style="list-style-type: none"> Comunicação Regular: Através de newsletters, mídias sociais ou aplicativos. Feedback e Pesquisa de Satisfação. Resolução de Problemas: Estabelecer um canal eficaz para lidar com reclamações ou problemas relacionados a produtos ou serviços. 	<ol style="list-style-type: none"> Seleção e Compra de Carnes: Armazenamento Adequado: Preparação de Cortes: Atendimento ao Cliente: Limpeza e Higiene: Gestão de Estoques: Controle de Qualidade: Gestão de Preços: Embalagem e Rotulagem: Gestão de Pedidos: Marketing e Promoção: Treinamento e Desenvolvimento: 	Recursos Tangíveis: <ol style="list-style-type: none"> Instalações Físicas: Equipamento: Estoque de Carne: Equipamento de Limpeza: Computadores e Software: Recursos Intangíveis: <ol style="list-style-type: none"> Conhecimento e Habilidade: Relacionamentos com Fornecedores: Marca e Reputação: A Receitas e Métodos de Preparação: Base de Clientes: A Estratégia de Marketing: Sistema de Gestão: Reconhecimento Local:
	1	3	6	7
		Canais <ol style="list-style-type: none"> Loja Física: Ter uma localização física é fundamental para permitir que os clientes visitem a boutique, escolham pessoalmente cortes de carne e recebam um atendimento personalizado. Loja Online: Ter uma plataforma de comércio eletrônico onde os clientes possam fazer pedidos e receber entregas em casa. Uma loja online pode atingir uma audiência mais ampla. 		
		4		
Estrutura de Custo			Fontes de Receita	
<ol style="list-style-type: none"> Custo da Carne: Este é o custo direto da carne que a boutique compra de fornecedores. Ele é uma das maiores despesas e pode variar dependendo dos cortes e da qualidade da carne vendida. Custos de Funcionários: Isso inclui salários e benefícios para os funcionários, como açouqueiros, atendentes, caixas e outros. Os custos de mão de obra são uma parte significativa da estrutura de custos. Utilidades: Contas de serviços públicos, como eletricidade, água e gás, são necessárias para manter as instalações e equipamentos funcionando. Marketing e Publicidade: Gastos com marketing, publicidade local, promoções e eventos para atrair clientes. Seguros: Seguro para proteger contra perdas, danos à propriedade e responsabilidade civil. Manutenção e Reparos: Manutenção regular e reparos em equipamentos, instalações e estruturas. Suprimentos de Escritório: Papelaria, suprimentos de escritório, produtos de limpeza e outros materiais de uso diário. Equipamento de Açougue e Ferramentas: Compra, manutenção e substituição de facas, serras e outros equipamentos de açougue. Despesas Administrativas: Isso inclui despesas gerais de administração, como software de contabilidade, serviços de contabilidade e taxas de licença. Tecnologia e Software: Gastos com computadores, software de ponto de venda (PDV) e sistemas de gestão. 			Preços dos Produtos: O principal fluxo de receita é gerado a partir da venda de produtos à base de carne, como cortes de carne bovina, suína, de frango, linguças, entre outros. Os preços desses produtos podem variar de acordo com o tipo de carne, a qualidade, o corte e a demanda local. Pacotes e Promoções: Oferecer pacotes especiais e promoções em certos produtos ou combinações de carne pode incentivar vendas adicionais e fidelizar os clientes. Serviços de Açougue: Além da venda de produtos de carne, a boutique de carne pode oferecer serviços de açougue, como cortes personalizados. Isso pode ser uma fonte adicional de receita. Produtos Gourmet e Valor Agregado: A boutique pode explorar produtos gourmet, como carnes envelhecidas, marinadas ou preparadas, que geralmente têm preços mais elevados. Assinaturas e Programas de Fidelidade: Um programa de assinatura que permite que os clientes recebam cortes de carne regularmente pode gerar receita recorrente. Além disso, programas de fidelidade podem incentivar clientes a comprar com mais frequência.	
		8		9

Fonte: Elaborado pelos autores

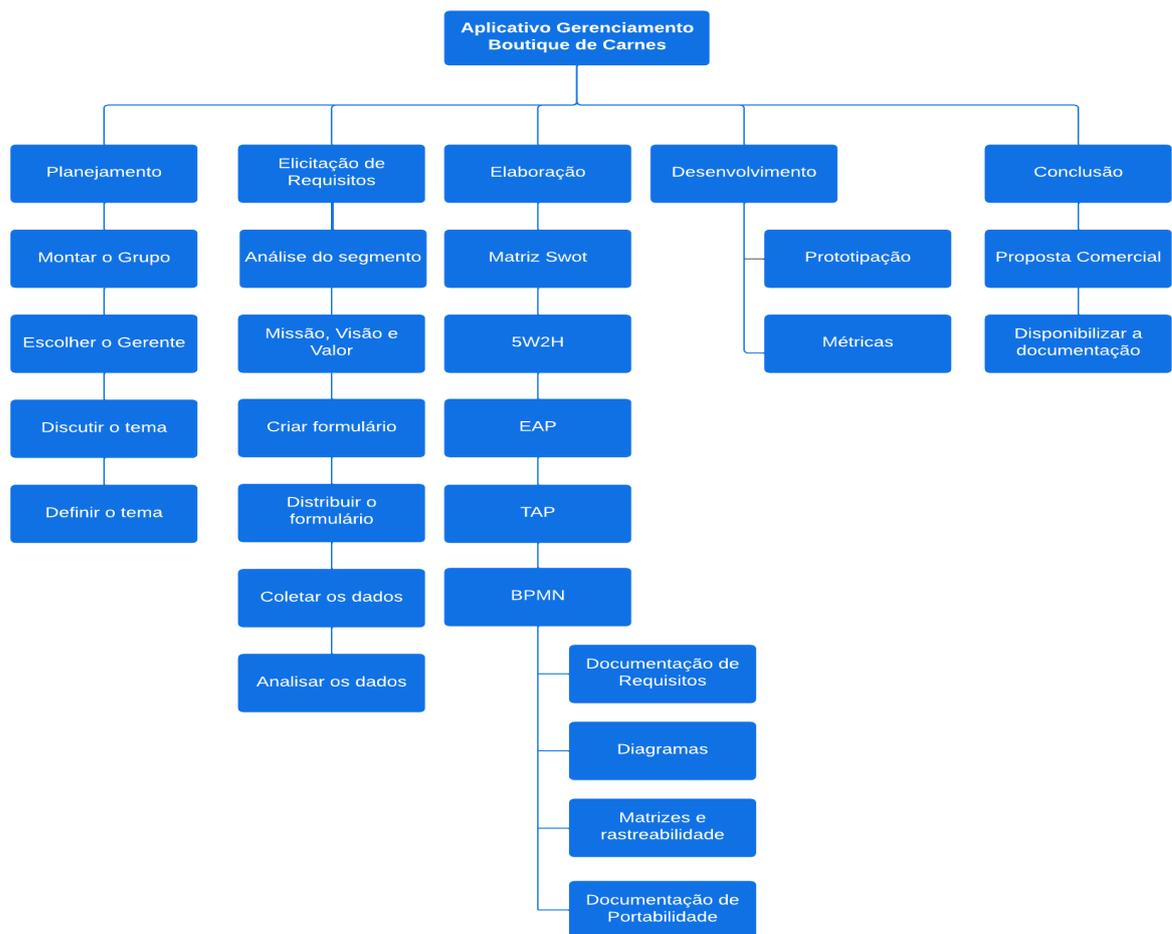
EAP (Estrutura analítica do projeto): A estrutura analítica do projeto nos ajuda a separar as tarefas de acordo com as dependências, como os projetos podem ser muito diferentes, os níveis da sua estrutura também o serão.

O objetivo deste documento é dividir o projeto em partes menores, para facilitar o gerenciamento e o controle. Após a definição da equipe partimos para a escolha de um gerente após algumas reuniões para definição de um tema e assim dividindo as tarefas que cada participante irá desenvolver.

Com base nas informações coletadas foi possível elaborar e definir estratégias para melhorar o funcionamento da empresa.

Com isso, preparamos a documentação de requisitos para apresentar para o cliente, e com sua aprovação, iniciamos os projetos.

Figura 2 - EAP

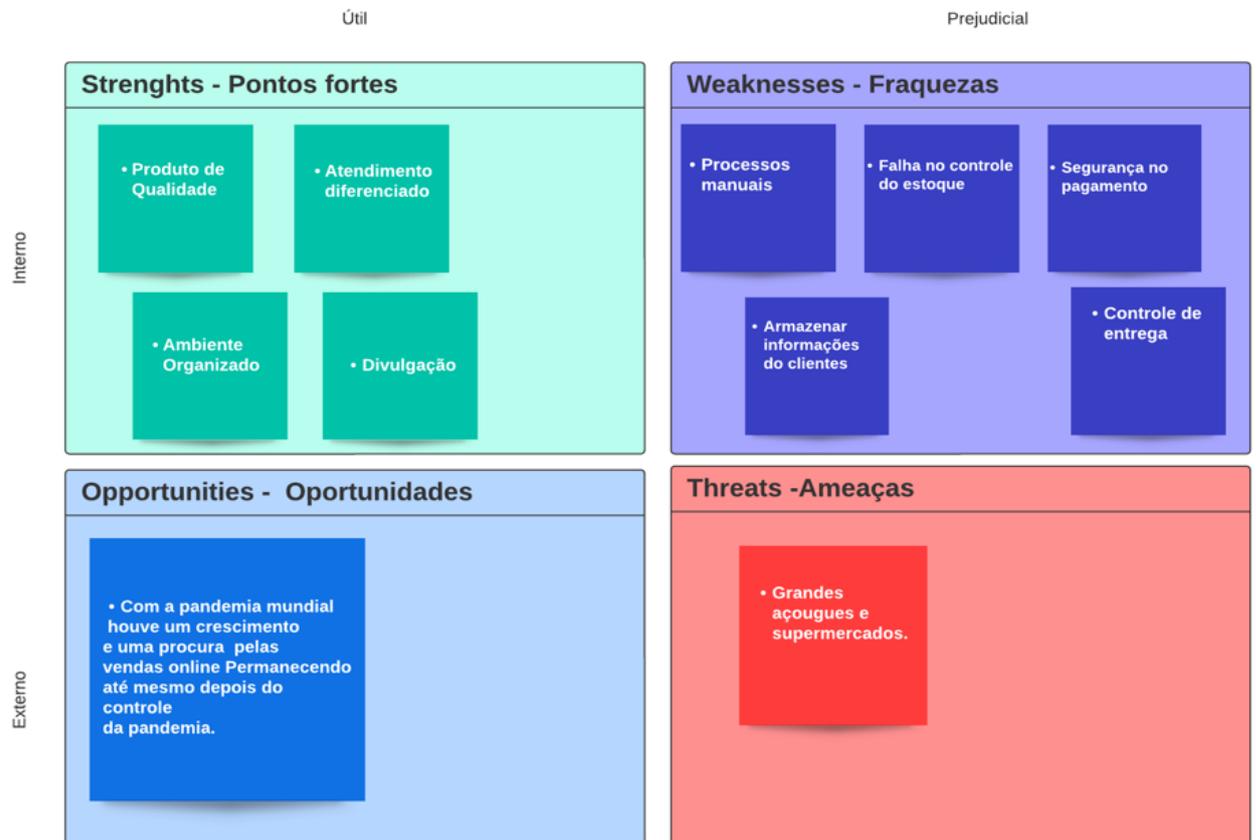


Fonte: Elaborado pelos autores

Matriz SWOT: em português conhecida como FOFA – é um método de planejamento estratégico que engloba a análise de cenários para tomada de decisões, observando 4 fatores. Em inglês: *Strengths, Weaknesses, Opportunities e Threats*. Em português: Forças, Oportunidades, Fraquezas e Ameaças.

A principal função da análise SWOT é avaliar resultados do ambiente externo e interno da empresa, desenvolvendo assim, informações necessárias para identificação dos pontos fracos e fortes, obtendo através disso, conhecimentos adequados para a formalização do planejamento estratégico da empresa visando a análise das oportunidades e fraquezas. (KOTLER; KELLER, 2012, p. 49).

Figura 3 - Matriz Swot



Fonte: Elaborado pelos autores.

Plano de ação – 5W2H: Manter um negócio no mercado não é uma tarefa fácil, dada a intensa concorrência e os desafios que surgem de tempos em tempos. No entanto, por meio do plano de negócios, ele oferecerá a oportunidade de desenvolver um bom plano e a estratégia certa para atender às necessidades dos consumidores e se adaptar às dificuldades iniciais do mercado de trabalho.

O Plano de Ação é uma ferramenta de gestão amplamente utilizada para planejar e monitorar as atividades a serem realizadas para atingir os objetivos da empresa. Isso é essencial para o processo de desenvolvimento do plano de negócios, permitindo simplificar todos os passos que serão dados para tomar a melhor decisão estratégica.

Figura 4 – 5W2H

Ação	O quê? (What?)	Porque? (Why?)	Onde? (Where?)	Quem? (Who?)	Quando? (When?)	Como? (How?)	Quanto? (How Much?)
1	Cadastrar Clientes	Validar e fidelizar dados do cliente e segurança no pagamento	Na tela inicial botão cadastrar	Cliente	Quando um novo cliente for realizar o pedido	Após validação dos dados do cliente, armazenar os dados no banco	Tempo , custo do desenvolvimento e armazenamento no banco de dados
2	Limitar as formas de pagamento	Garantir o recebimento das vendas antes das manipulação das carnes	Botão finalizar pedido	Software	Ao finalizar o pedido	As formas de pagamentos disponíveis no aplicativo serão apenas aquelas validadas através de uma API. Podendo ser liberadas novas formas apenas para clientes fidelizados	Tempo, custo de desenvolvimento
3	Criar pedido	Para automatizar o processo de recebimento dos pedidos	Na aba Pedidos	Software	Ao entrar um novo pedido pelo aplicativo	Assim que um cliente realizar um novo pedido pelo seu aplicativo móvel, a empresa recebe a notificação e o pedido com todos dados especificados no software	Tempo , custo do desenvolvimento e armazenamento no banco de dados
4	Controlar estoque	Para automatizar o controle das mercadorias, evitar erros humanos e demonstrar a quantidade disponível dos produtos no aplicativo	Na Aba Estoque	Açougueiro	A todo momento	Através da chegada de novas mercadorias, incluindo as notas fiscais no estoque, a saídas para manipulação e para os pedidos recebidos no aplicativo.	Tempo , custo do desenvolvimento e armazenamento no banco de dados
5	Controlar entrega	Melhorar a rastreabilidade da mercadoria entregue, controlar a qualidade no resfriamento das carnes e calcular o tempo da entrega	Na aba Entregas	Entregador	Sempre que houver a notificação da confirmação do pedido	Baseado no carrinho de compras será calculado o tempo de entrega, o entregador será notificado, todo seu trajeto será rastreado e se houver alguma ocorrência na temperatura do baú o software vai enviar um notificação.	Tempo , custo do desenvolvimento e do sensor de temperatura e armazenamento no banco de dados

Fonte: Elaborado pelos autores.

A empresa está na cidade de Franca/SP, e atua por doze anos no segmento de manipulação e comércio de cortes de carne. Trabalhando com uma variedade de produtos, sempre visando a qualidade, que excedem as expectativas dos clientes o que fez seu crescimento aumentar nos últimos três anos. O crescimento impulsionado pela pandemia, a empresa despertou uma tendência a entregas na qual não possuía.

Ao longo do tempo, a butique vem se destacando na cidade, com seu diferencial na qualidade dos cortes e personalização do atendimento, conquistando seu lugar dentro do seu segmento.

Entretanto, seus processos ainda permanecem manuais como os pedidos, gerenciamento de estoque, a comunicação interna e as entregas. Com o desenvolvimento no mercado, aumento dos pedidos online, esses processos se tornaram cada vez mais difíceis. O controle e qualidade dos produtos, a experiência com o cliente acabou sendo prejudicados com esse aumento de vendas.

3 Levantamento de Requisitos

3.1 Elicitação e especificação dos Requisitos

O termo “elicitado” pode ser definido como: definir, tornar explícito, obter o máximo de informação sobre o objeto em questão. De acordo com o Dicionário Aurélio (Aurélio, 1999), encontra-se dentre outras as seguintes definições do termo: fazer sair; extrair. A elicitação de requisitos é o processo de buscar, descobrir, adquirir, elaborar requisitos para sistemas baseados em computador.

A elicitação de requisitos é a primeira atividade no processo de engenharia de requisitos, na qual se busca entender quais são as necessidades do usuário que devem ser atendidas pelo software que será desenvolvido.

As técnicas mais comuns utilizadas em elicitação de requisitos são as de introspecção, entrevista e análise de protocolos. A técnica de introspecção se baseia em imaginar que tipo de sistema eu iria querer se eu estivesse executando esta tarefa, utilizando este equipamento, entre outros. Ou seja, imaginar que propriedades um sistema deveria possuir de forma a obter sucesso. As entrevistas podem ser divididas em entrevistas com questionários, entrevistas livres e grupos focais e de desenvolvimento de aplicações.

As entrevistas com questionários são amplamente utilizadas e, segundo, têm a vantagem de parecerem científicas porque podem utilizar análise estatística. A coleta de fatos é feita sobre as fontes de informação que foram identificadas na fase anterior, e a comunicação tem o papel de permitir o diálogo tanto para entender como para confirmar as informações corretas dos fatos.

Iniciamos o projeto com reuniões para discutir, planejar e organizar como iremos abordar o levantamento dos requisitos, optamos pelo modelo de questionário. Elaboramos doze questões para avaliarmos o negócio e implementar as mudanças necessárias.

Concluiremos nosso projeto com a elicitação dos requisitos funcionais e não funcionais.

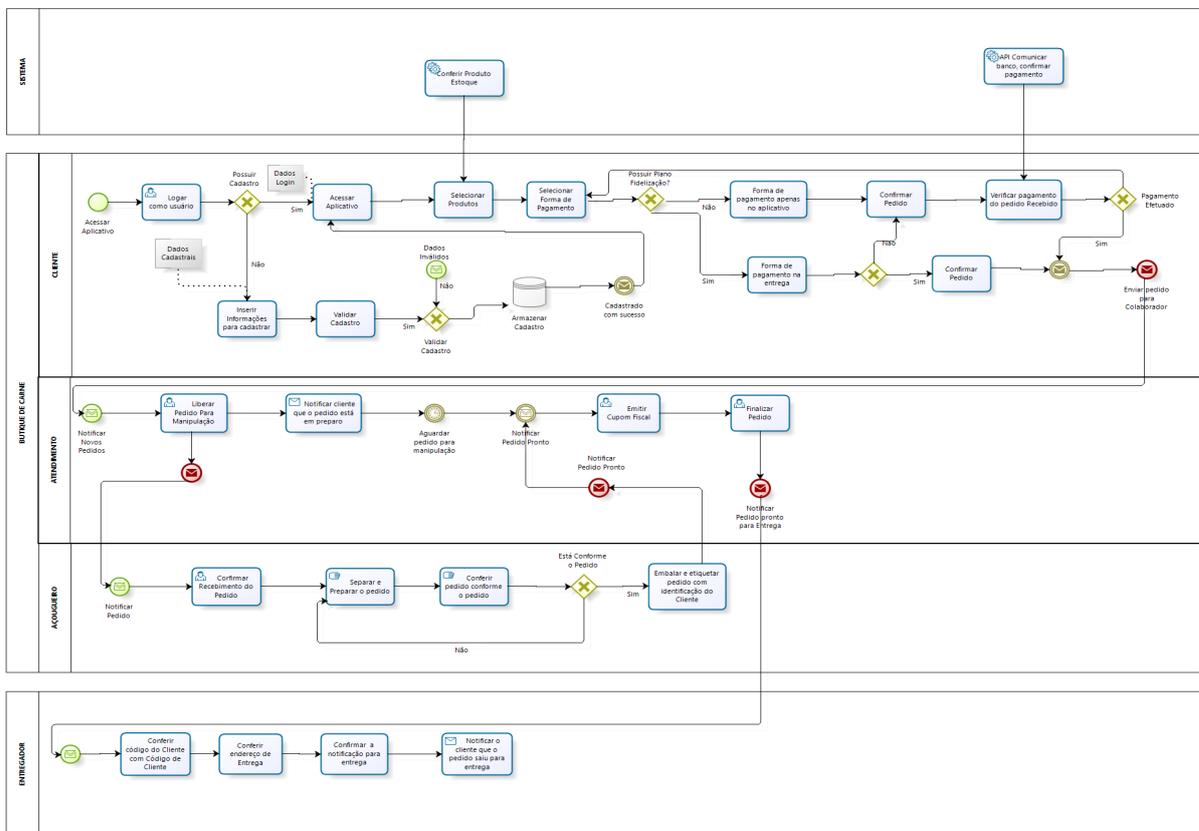
3.2 BPMN

De acordo com Hammer e Champy (1994), "um processo é um grupo de atividades realizadas numa sequência lógica com o objetivo de produzir um bem ou um

serviço que tem valor para um grupo específico de clientes", ou seja, é uma sequência de passos finita que visa definir um conjunto de atividades em que se tem uma entrada, a transformação dessa entrada e uma saída.

A abreviação BPMN traduzida para português significa Modelo de Processo de Negócios. O foco da gestão de processos é permitir que as empresas criem uma cultura de melhoria contínua em seus fluxos de trabalho e desenvolvam a forma como as pessoas trabalham para atingir um objetivo específico da empresa.

Figura 5 – BPMN



Fonte: Elaborado pelos autores.

3.3 Requisitos Funcionais

Os requisitos funcionais são utilizados na engenharia de software eles determinam o modo de operação do sistema e sua interação com os usuários, outros sistemas ou componentes. Requisito define "como" esses serviços serão executados no sistema são imprescindíveis no processo de desenvolvimento de software visto que desempenham um papel fundamental no projeto, na implementação e nos testes do sistema.

Quadro 1: Requisitos Funcionais

ID: RF001	Acessar Aplicativo
Categoria: Evidente	Prioridade: Essencial
Descrição:	O sistema deverá liberar o acesso ao aplicativo
Informações:	Código do Usuário
Regras de Negócio:	RN001 -O cadastro será validado mediante o CPF ou CNPJ.
ID: RF002	Possuir Cadastro
Categoria: Oculto	Prioridade: Alta
Descrição:	O sistema fará a verificação se o cliente possui ou não cadastro
Informações:	Número do CPF verificar no banco de dados se possui cadastro
Regras de Negócio:	RN001-O cadastro será validado mediante o CPF ou CNPJ.
ID: RF003	Cadastrar Clientes
Categoria: Evidente	Prioridade: Alta
Descrição:	O sistema deverá cadastrar os dados pessoais do cliente, caso não possua um cadastro
Informações:	CPF, CNPJ, cod_Id, senha, nome, rua, número, complemento, cidade, estado, celular, e-mail.
Regras de Negócio:	RN001 -O cadastro será validado mediante o CPF ou CNPJ.
ID: RF004	Cadastrar Produtos
Categoria: Evidente	Prioridade: Essencial
Descrição:	O sistema deverá permitir o cadastro dos produtos
Informações:	Código do Produto, Descrição do Produto, Corte, Quantidade, Embalagem, Data de Fabricação, Data de Validade, Preço de Custo, Preço Médio, Preço de Venda, Estoque disponível
Regras de Negócio:	RN002 - O sistema deverá permitir alterar, inativar os produtos e guardar o log das alterações feitas.

ID: RF005	Efetuar Login
Categoria: Evidente	Prioridade: Alta
Descrição:	O sistema deverá permitir o login após validar o cadastro do cliente.
Informações:	Código do Usuário, senha
Regras de Negócio:	RN003 - Possuir o cadastro no aplicativo
ID: RF006	Selecionar produtos
Categoria: Evidente	Prioridade: Alta
Descrição:	O sistema deverá permitir selecionar os produtos desejados
Informações:	-O Acesso será feito com o Código do usuário e senha. -Selecionar o produto que estiver disponível -Selecionar a quantidade -Mostrar no final o total da compra
Regras de Negócio:	RN004 - Deixar selecionar somente os produtos que tiverem quantidades disponíveis
ID: RF007	Selecionar forma de pagamento
Categoria: Evidente	Prioridade: Alta
Descrição:	O sistema deverá permitir selecionar a forma de pagamento
Informações:	Forma_Pgto, Cartão Débito ou Crédito, Pix ou Dinheiro
Regras de Negócio:	RN005: Nos pagamentos on-line só serão aceitos pagamento no cartão ou Pix, exceto se o cliente estiver cadastrado no programa de cliente fidelizado poderá ser em dinheiro:

Fonte: Elaborado pelos autores.

3.4 Requisitos não funcionais

A determinação da qualidade global de um *software* depende do cumprimento dos requisitos não funcionais. Garantindo a confiabilidade, segurança e eficiência do

sistema em uma variedade de situações. Eles também fornecem critérios para avaliar se ele está atendendo aos padrões de desempenho e qualidade requeridos.

Quadro 2: Requisitos Não Funcionais

ID: RNF001	Servidor
Categoria: Segurança	Prioridade: Obrigatório
Descrição:	O servidor deverá armazenar os dados para comunicação de validação com a receita federal e os bancos.
Informações:	Servidor será utilizado para controlar o estoque e o programa de fidelização do cliente
Regras de Negócio:	Nenhuma
ID: RNF002	Dispositivo de rastreabilidade da mercadoria
Categoria: Segurança	Prioridade: Obrigatório
Descrição:	O dispositivo deverá armazenar os dados da entrega
Informações:	Ao concluir o pedido será enviado um código que será os 3 últimos números do celular do cliente para quando o entregador chegar solicitar o número para entregar a mercadoria
Regras de Negócio:	Nenhuma

Fonte: Elaborado pelos autores.

3.5 Diagrama de caso de uso

Um diagrama de caso de uso descreve como os usuários interagem com um sistema para executar um processo. Em outras palavras, ele demonstra os objetivos dos usuários, as interações entre os usuários e o sistema e os processos exigidos pelo sistema para executar esses objetivos.

Em cada caso de uso existe uma descrição das funcionalidades que o sistema precisará fazer. E cada um deles pode ter associações com outros casos de uso que compõem um subconjunto do modelo para um propósito.

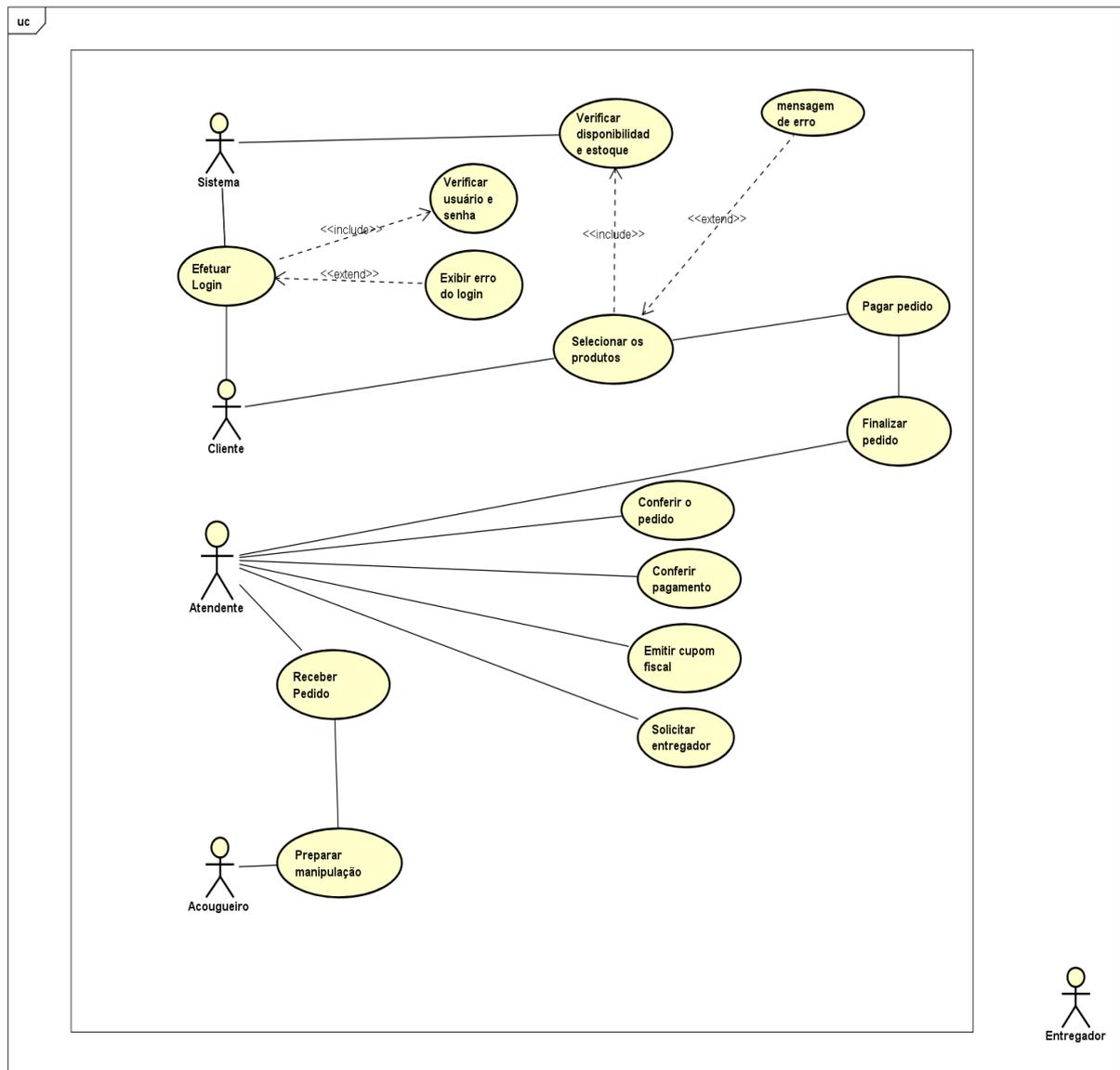
Os modelos de caso de uso se definem em atores, podendo atuar como um ser humano ou até mesmo uma máquina (como uma impressora, um *software*). Cada um

deles possui descrições que especificam o fluxo dos eventos para que possam executar o processo.

Enquanto o BPMN se destaca como uma ferramenta versátil de gestão de processos de negócios, ele difere do mapeamento de processos atuais, que tem um foco mais restrito em padronização, treinamento, qualidade e conformidade.

Além disso, o BPMN possui uma relação estreita com a Linguagem de Modelagem Unificada (UML), que é amplamente utilizada no design de software. Essa conexão entre BPMN e UML permite uma integração eficaz entre a modelagem de processos de negócios e o design de sistemas de software, facilitando a comunicação e a colaboração entre as equipes de negócios e tecnologia (LUCIDCHART, [s.d.] 2023).

Figura 6 – Diagrama de Casos de Uso



Fonte: Elaborado pelos autores

3.5.1 Documentação dos Casos de Uso

A documentação de casos de uso tem por finalidade, descrever de forma simples, completa, e principalmente compreensível aos usuários os processos executados. Sendo assim, os desenvolvedores ou até mesmo usuários mais leigos poderão obter as informações necessárias sobre o projeto e sua implementação.

Na Linguagem de Modelagem Unificada (UML), um diagrama de caso de uso é uma representação que condensa as informações relativas aos usuários de um sistema, também chamados de atores, e suas interações com o sistema em questão.

Através de símbolos e conectores especializados, esse diagrama permite a visualização e discussão de cenários nos quais o sistema interage com pessoas, organizações ou sistemas externos auxiliando na compreensão das metas ajudando os atores a alcançarem uma definição clara do escopo e do sistema, fornecendo um meio eficaz de documentar e comunicar as funcionalidades e interações envolvidas no projeto.

Quadro 3 - Documentação de Caso de Uso

Caso de Uso - Logar	
ID	UC 001
Descrição	Este caso de uso tem o objetivo de cadastrar e logar no sistema
Ator-primário	Cliente
Pré-condição	Ter efetuado o cadastro no sistema - Possuir CPF
Cenário Principal	1.O use case inicia quando o usuário inicia o aplicativo 2. O sistema carrega a tela de login 3. O usuário insere o login e a senha 4. O sistema faz a verificação para validar o login 5. O sistema abre a interface do sistema
Pós-condição	Atualizar banco de dados
Cenário Alternativo	3a - Se o usuário informar os dados errados, o sistema deve notificar sobre o erro. 3b - Caso o usuário tenha esquecido sua senha, ele deve acessar a opção de alterar ou recuperar senha e login. 3c - O sistema deve oferecer a opção de cadastramento caso o cliente não possua. 3c. 1- Caso o cliente já tenha cadastro no sistema, notificar CPF já cadastrado e oferecer opção de recuperar senha
Inclusão	Verificar usuário e senha
Extensão	Exibir erro do login

Caso de Uso - Selecionar Produto	
ID	UC 002
Descrição	Este caso de uso tem o objetivo selecionar produtos para a compra
Ator-primário	Cliente
Pré-condição	Ter efetuado o login no sistema
Cenário Principal	<ol style="list-style-type: none"> 1. O use case inicia quando o usuário acessa o menu de seleção dos produtos 2. O sistema faz a verificação no estoque 3. O usuário seleciona os produtos 4. O usuário verifica a seleção dos produtos 5. O sistema deve oferecer uma caixa de texto para possíveis observações sobre o pedido(corte a ser executado) etc. 6. O usuário finaliza a seleção
Pós-condição	Nenhuma
Cenário Alternativo	<p>2a - O sistema deve verificar se possui produto no estoque, caso não tenha, emitir uma notificação .</p> <p>6a - Caso o usuário não selecionar nenhum item, o sistema notifica que um produto precisa ser selecionado</p>
Inclusão	Verificar disponibilidade do estoque
Extensão	Mensagem de erro estoque não disponível
Caso de Uso - Verificar Estoque	
ID	UC 003
Descrição	Este caso de uso tem o objetivo verificar disponibilidade de produto no estoque
Ator-primário	Sistema
Pré-condição	Ter sido cadastrado no sistema - Possuir NF
Cenário Principal	<ol style="list-style-type: none"> 1. O use case inicia quando o usuário entra no menu de compra 2. O sistema carrega a lista de produtos existentes. 3. O sistema verifica a quantidade no estoque 4. O sistema aguarda o cliente finalizar a seleção os produtos que deseja. 5. O sistema mostra a quantidade disponível e valor. 6. O sistema valida a disponibilidade.
Pós-condição	Nenhuma
Cenário Alternativo	<p>3a - O sistema deve verificar se possui o produto no estoque, caso não emitir uma notificação</p> <p>3 b - Caso não tenho o produto, notificar a falta no estoque do produto</p> <p>4a- O sistema não valida sem a seleção de algum produto</p>
Inclusão	Verificar disponibilidade do estoque
Extensão	Nenhuma

Caso de Uso - Gerar Pedido	
ID	UC 004
Descrição	Este caso de uso tem o objetivo registrar o pedido de venda
Ator-primário	Sistema
Pré-condição	Ter passado pela UC 002 - Ter sido feito a seleção dos produtos no sistema Ter passado pelo UC 003 - Ter feito a consulta no estoque
Cenário Principal	<ol style="list-style-type: none"> 1. O use case inicia quando o usuário seleciona o campo de finalizar pedido no sistema. 2. O sistema carrega o formulário de venda. 3. O sistema salva o pedido. 4. O sistema valida as informações. 5. O sistema salva as informações no banco.
Pós-condição	Nenhuma
Cenário Alternativo	2a - Todos os campos do formulário e obrigatório, e caso não seja preenchido o sistema indicará os campos que estão em branco ou incorreto.
Inclusão	Nenhuma
Extensão	Nenhuma
Caso de Uso - Pagar Pedido	
ID	UC 005
Descrição	Este caso de uso tem o objetivo de baixar o pagamento no sistema
Ator-primário	Sistema
Pré-condição	Ter passado pela UC 002 - Ter sido feito a seleção dos produtos no sistema - Ter sido finalizado a compra no sistema
Cenário Principal	<ol style="list-style-type: none"> 1. O use case inicia quando o cliente seleciona o campo de efetuar pagamento 2. O sistema redireciona para o relatório do pedido com os dados de produto: tipos do produto, quantidade, valores, horário de entrega etc. 3. O sistema verifica se o cliente possui fidelidade 4. Se ele estiver cadastrado no sistema de fidelização, o sistema disponibiliza a opção de pagamento na retirada ou entrega em dinheiro se não possuir disponibiliza a opção de pagamento cartão de débito/crédito 5. O sistema gera o comprovante do pagamento 6. O sistema gera a senha de entrega
Pós-condição	Nenhuma
Cenário Alternativo	<p>2a- o sistema deve ter oferecer opção de retorno caso o cliente tenha se esquecido ou mudado de ideia sobre o pedido.</p> <p>4 a - O sistema notifica quando o cliente ja possuir fidelização ou não.</p>
Inclusão	Nenhuma
Extensão	Nenhuma

Caso de Uso - Finalizar Pedido	
ID	UC 006
Descrição	Este caso de uso tem o objetivo finalizar o pedido e ser enviado para o preparo
Ator-primário	Sistema
Pré-condição	Ter passado pelo UC 003 - Ter sido verificado no estoque - Ter passado pelo UC 005 - Ter sido efetuado o pagamento
Cenário Principal	<ol style="list-style-type: none"> 1. O use case inicia quando o cliente seleciona o campo finalizar pedido 2. É gerado um relatório final de todo processo de compra do produto contendo os dados do pagamento e do produto: comprovante, código de venda, e a previsão da entrega. 3. O sistema redireciona para uma aba de acompanhamento do produto 4. O sistema gera o código para rastreamento do produto para o cliente acompanhar o caminho do produto até a entrega
Pós-condição	Nenhuma
Cenário Alternativo	3 a - O sistema deve oferecer opção de falar com atendente caso tenha algum erro no sistema ou na compra do produto
Inclusão	Nenhuma
Extensão	Nenhuma
Caso de Uso - Receber Pedido	
ID	UC 007
Descrição	Este caso de uso tem o objetivo enviar o pedido para o atendente
Ator-primário	Sistema
Pré-condição	Ter passado pelo UC 005 - ter efetuado o pagamento - Ter passado pelo UC 006 - ter finalizado o pedido
Cenário Principal	<ol style="list-style-type: none"> 1. O use case inicia quando o cliente seleciona o campo de finalizar pedido 2. O sistema atualiza os pedidos 3. O pedido é enviado para o atendente contendo todos os dados do produto e pagamento para a preparação
Pós-condição	Nenhuma
Cenário Alternativo	3a - Quando o pedido não possuir pagamento no ato da compra, o sistema deve destacar esse pagamento a ser efetuado no ato da entrega ou retirada por possuir plano de fidelização.
Inclusão	Nenhuma
Extensão	Nenhuma

Caso de Uso - Conferir Pagamento	
ID	UC 008
Descrição	Este caso de uso tem o objetivo conferir se o pagamento do cliente foi efetuado no sistema ou se será efetuado na entrega
Ator-primário	Atendente
Pré-condição	Ter passado pelo UC 007 - O pedido ter sido enviado ao atendente
Cenário Principal	<ol style="list-style-type: none"> 1. O use case inicia quando o sistema envia o pedido pago para o atendente 2. O Atendente confere se a forma de pagamento será feito no ato da compra na retirada ou entrega 3 - O atendente efetua a baixa do pedido
Pós-condição	Nenhuma
Cenário Alternativo	2a - Caso o pedido tenha alguma divergência, o atendente retorna para o cliente pelo chat ou pelo telefone para correção.
Inclusão	Nenhuma
Extensão	Nenhuma
Caso de Uso - Conferir Pedido	
ID	UC 009
Descrição	Este caso de uso tem o objetivo conferir os dados do pedido
Ator-primário	Atendente
Pré-condição	Ter passado pelo UC 007 - O pedido ter sido enviado ao atendente
Cenário Principal	<ol style="list-style-type: none"> 2. O atendente faz a conferência do pedido 3. O atendente da baixa no pedido
Pós-condição	Nenhuma
Cenário Alternativo	2a - Caso o pedido tenha alguma divergência, o atendente retorna para o cliente pelo chat ou pelo telefone para correção.
Inclusão	Nenhuma
Extensão	Nenhuma

Caso de Uso - Emitir Cupom Fiscal	
ID	UC 010
Descrição	Este caso de uso tem o objetivo gerar cupom fiscal da compra efetuada
Ator-primário	Atendente
Pré-condição	Ter passado pelo UC 006 - ter finalizado o pedido - Ter passado pelo UC 005 - Ter sido efetuado o pagamento
Cenário Principal	<ol style="list-style-type: none"> 1. O use case inicia quando o atendente recebe o pedido 2. O atendente confere os dados do pedido 3. O atendente gera o cupom fiscal para ser impresso e anexado ao pedido
Pós-condição	Nenhuma
Cenário Alternativo	Nenhum
Inclusão	Nenhuma
Extensão	Nenhuma
Caso de Uso - Preparar Manipulação	
ID	UC 011
Descrição	Este caso de uso tem o objetivo fazer o corte das carnes e manipulação
Ator-primário	Açougueiro
Pré-condição	Ter passado pelo UC 009 - Ter sido conferido e enviado pelo atendente
Cenário Principal	<ol style="list-style-type: none"> 1. O use case inicia quando o atendente finaliza a baixa do pedido 2. O atendente envia o pedido com os dados referente os tipos de carnes, cortes e adicionais 3. O atendente notifica o cliente pelo aplicativo que o preparo está sendo feito 4. O sistema inicia o timer de preparo até a entrega do produto. 5. O açougueiro faz o preparo da carne. 6. Notifica no sistema interno sobre a finalização da manipulação e corte. 7. O atendente organiza os pedidos finalizados
Pós-condição	Nenhuma
Cenário Alternativo	<p>4 a - o açougueiro deve notificar o sistema interno quando houver muitos cortes seneo feitos</p> <p>4 a.1 - O sistema deve notificar o cliente quando identificar excesso de pedidos e houver atrasos na preparação ou entregas</p>
Inclusão	Nenhuma
Extensão	Nenhuma

Caso de Uso - Solicitar Entregador	
ID	UC 012
Descrição	Este caso de uso tem o objetivo solicitar um entregador para efetuar a entrega do produto
Ator-primário	Atendente
Pré-condição	Ter passado pelo UC 011 - ter sido finalizado a manipulação - Ter passado pelo UC 009 - Ter sido conferido o pagamento
Cenário Principal	<ol style="list-style-type: none"> 1. O use case inicia quando o atendente recebe a notificação do açougueiro sobre a finalização da manipulação 2. O atendente anexa a encomenda os dados do cliente como endereço, telefone e forma de pagamento. 3. O atendente solicita entregador
Pós-condição	Nenhuma
Cenário Alternativo	Nenhum
Inclusão	Nenhuma
Extensão	Nenhuma

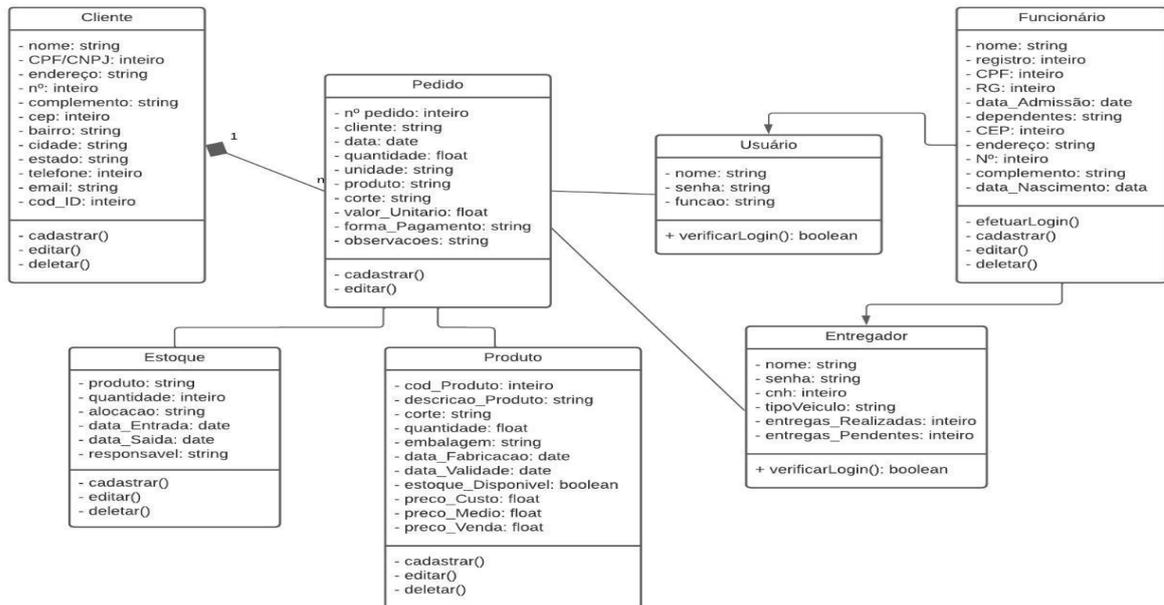
Fonte: Elaborado pelos autores.

3.6 Diagrama de Classe

Um tipo específico de diagrama utilizado na UML (*Unified Modeling Language*) é o diagrama de classe, que permite visualizar os vários tipos distintos de objetos dentro do sistema e suas interconexões estáticas. No intuito de representar visualmente as classes do sistema em questão juntamente com seus atributos, métodos e relação uns com os outros é utilizado o diagrama da classe. Cada retângulo usado como representação de uma classe possui três partes distintas: o nome da classe, os atributos e os métodos.

Para resumir tudo isso em poucas palavras: o diagrama de classe é crucial para a criação da representação visual dos sistemas orientados à objeto. É através dele que podemos observar com clareza quais são as classes envolvidas no projeto assim como seus respectivos atributos e métodos. Ele ainda propicia uma visão geral sobre todas as relações estabelecidas entre esses elementos neste contexto específico. É amplamente empregado em engenharia de software e se destaca como uma das ferramentas mais indispensáveis na modelagem de sistemas orientados a objetos.

Figura 7 – Diagrama de Classe



Fonte: Elaborado pelos autores.

3.7 Diagrama de Atividade

O Diagrama de Atividades na engenharia de software permite uma representação gráfica clara e concisa do fluxo das atividades. É parte da linguagem UML (Unified Modeling Language) e é usado para modelar tanto os processos de negócios quanto os fluxos.

Ao mapear um fluxo de trabalho eficiente é imprescindível incorporar elementos-chave como atividades específicas para cada etapa do processo. Além disso também são importantes as decisões que precisam ser tomadas em determinados momentos cruciais assim como os possíveis desvios ou alternativas (bifurcações) que podem surgir ao longo da execução.

Por último, mas não menos importante estão os pontos onde o fluxograma tem seu início e fim definidos. O controle entre as atividades é visualizado através das setas de controle que indicam a ordem em que devem ser executadas. Para mostrar opções distintas e possíveis percursos, é importante utilizar a representação de decisões e bifurcações. Dar início a partir de um ponto de partida e encerrar em um ponto de chegada.

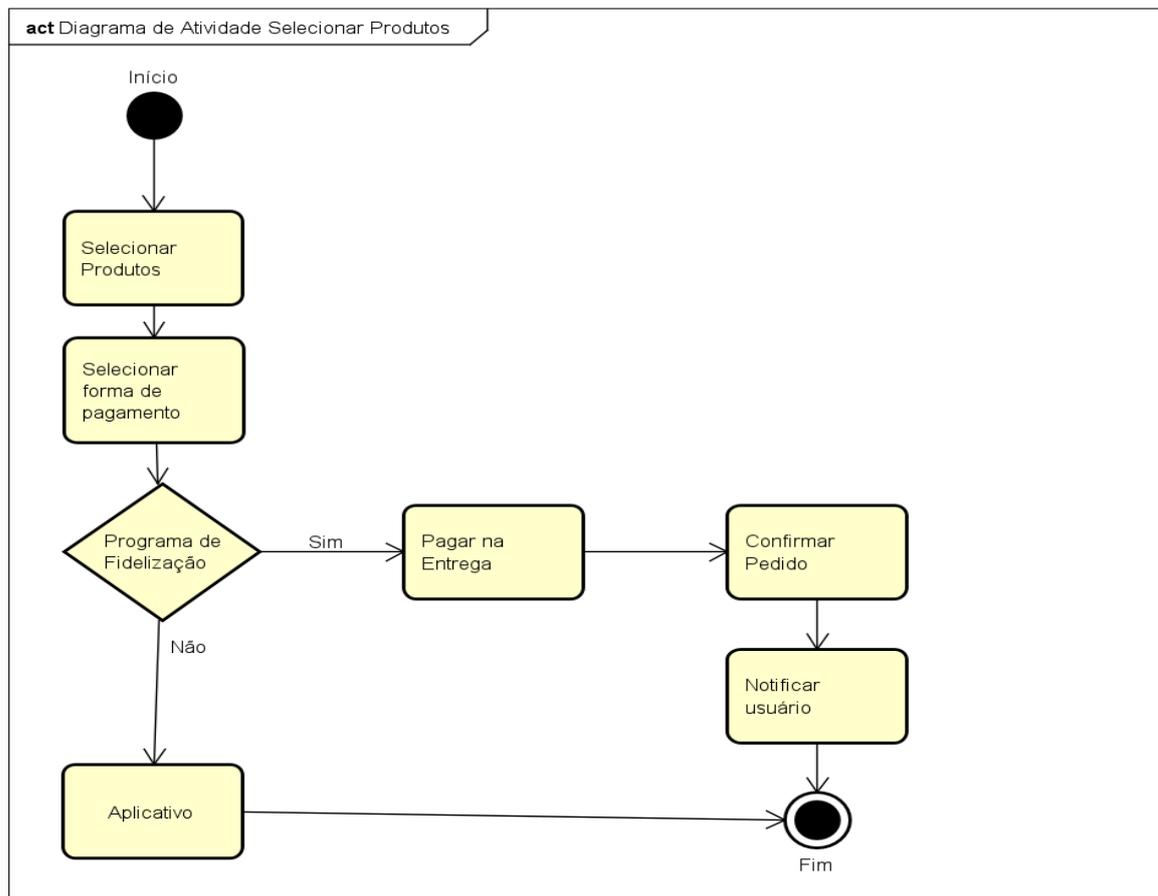
Sua utilização se dá para moldar ações complexas e sua reutilização é viável em diferentes partes do sistema.

Colaborar na documentação e comunicação do comportamento de sistemas complexos. Integrar-se com outros diagramas UML, como Diagrama de Implantação e o Diagrama Classes.

Para modelagem e comunicação claras dos comportamentos em sistemas, processos e algoritmos na área da engenharia de software, o Diagrama de Atividades se destaca como uma valiosa ferramenta.

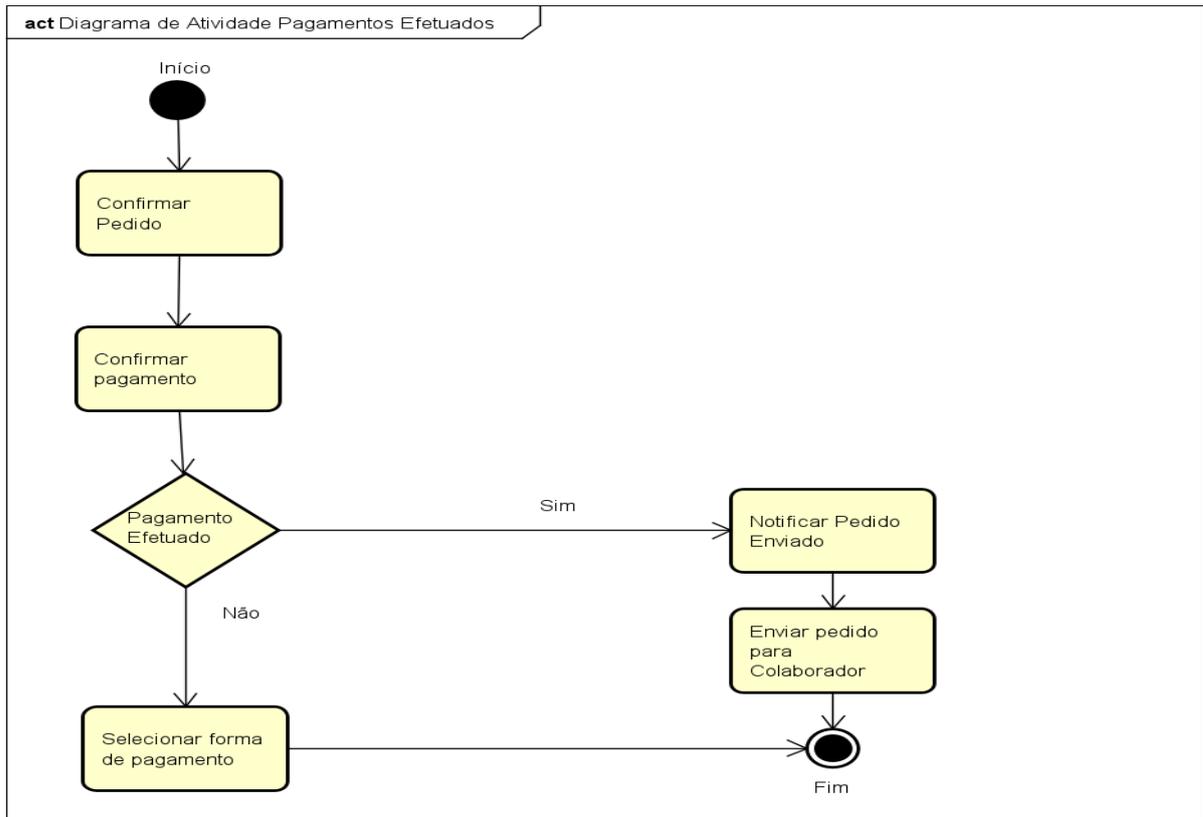
É comumente empregado tanto na análise e desenvolvimento de software quanto para a descrição dos processos comerciais.

Figura 8 – Diagrama de Atividades Selecionar produtos



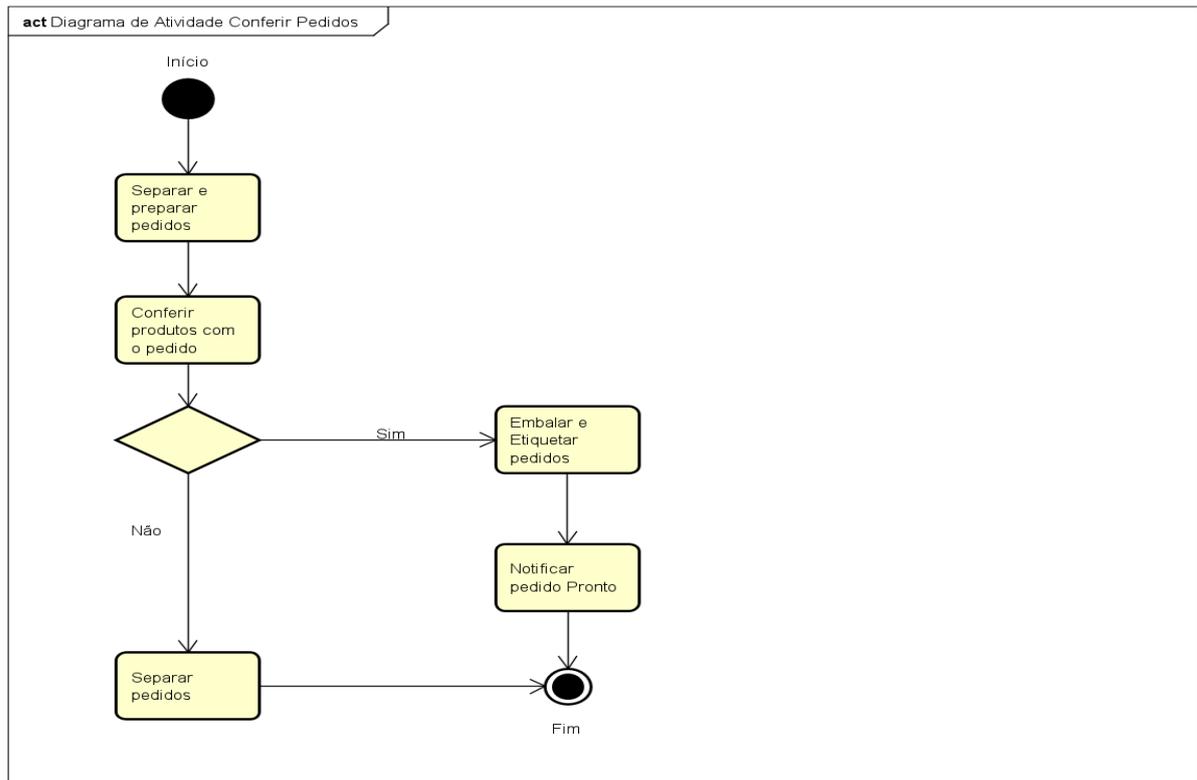
Fonte: Elaborado pelos autores.

Figura 9 – Diagrama de Atividade Pagamentos Efetuados



Fonte: Elaborado pelos autores

Figura 10 – Diagrama de Atividades Conferir Pedidos



Fonte: Elaborado pelos autores.

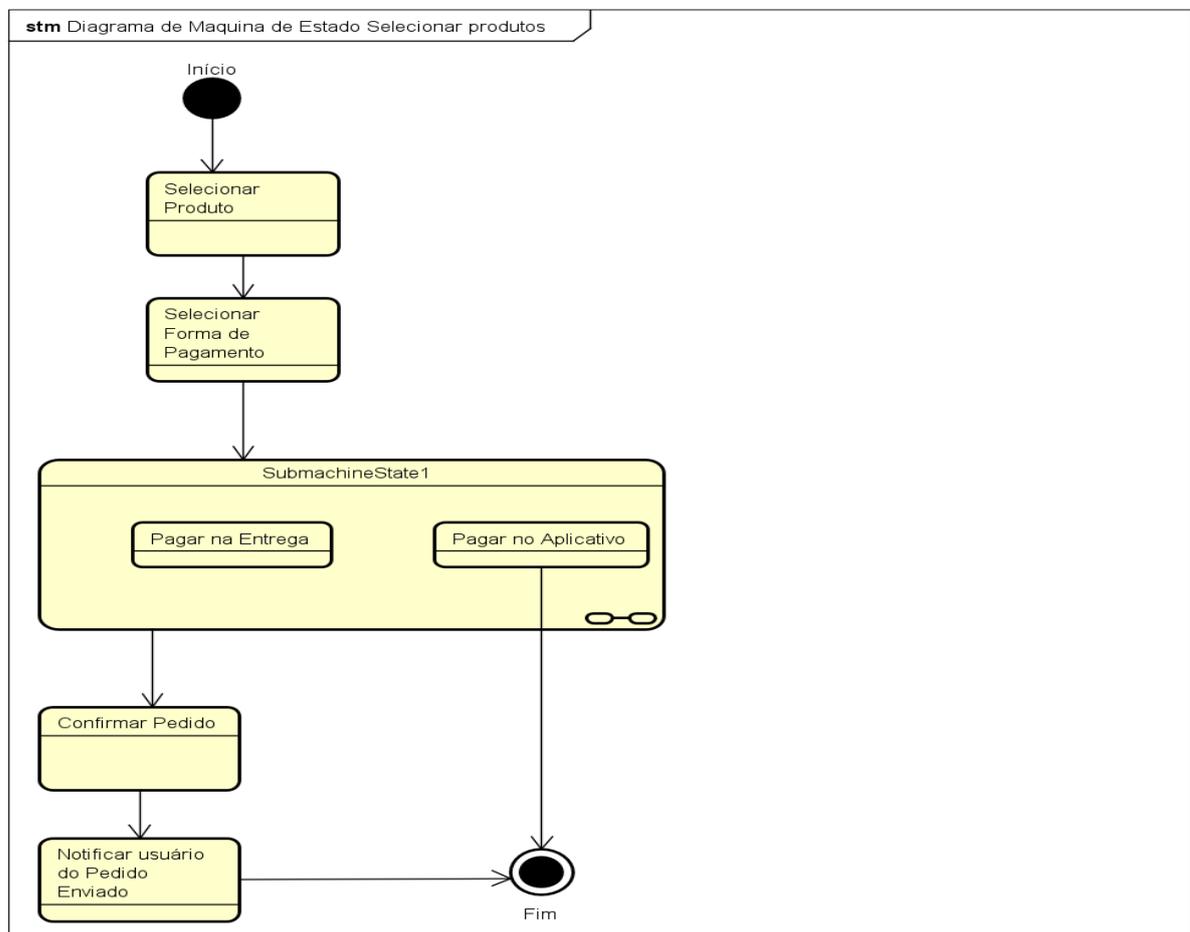
3.8 Diagrama de Máquina de Estado

O diagrama da máquina do estado, na linguagem UML, provê uma visão visual das mudanças nos diferentes objetos através das diversas transições.

O objetivo dele é exibir visualmente todas as opções de estado de um objeto e demonstrar o caminho percorrido pelo objeto até atingir determinado estado.

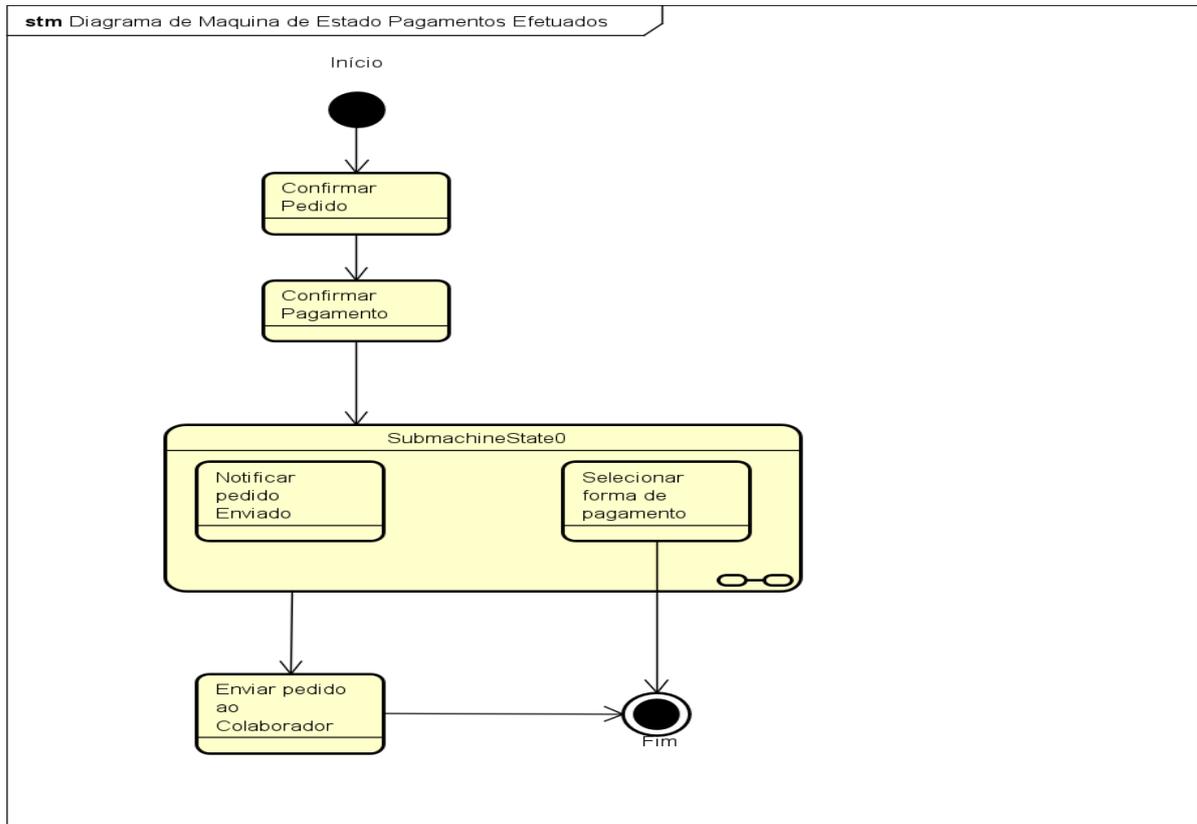
Em engenharia de software, o diagrama de máquina é utilizado para descrever como um objeto se move por vários estados ao longo do seu tempo de vida e mostrar o comportamento geral ou relacionado a máquinas ou conjuntos delas.

Figura 11 – Diagrama Máquina de Estado Selecionar produtos



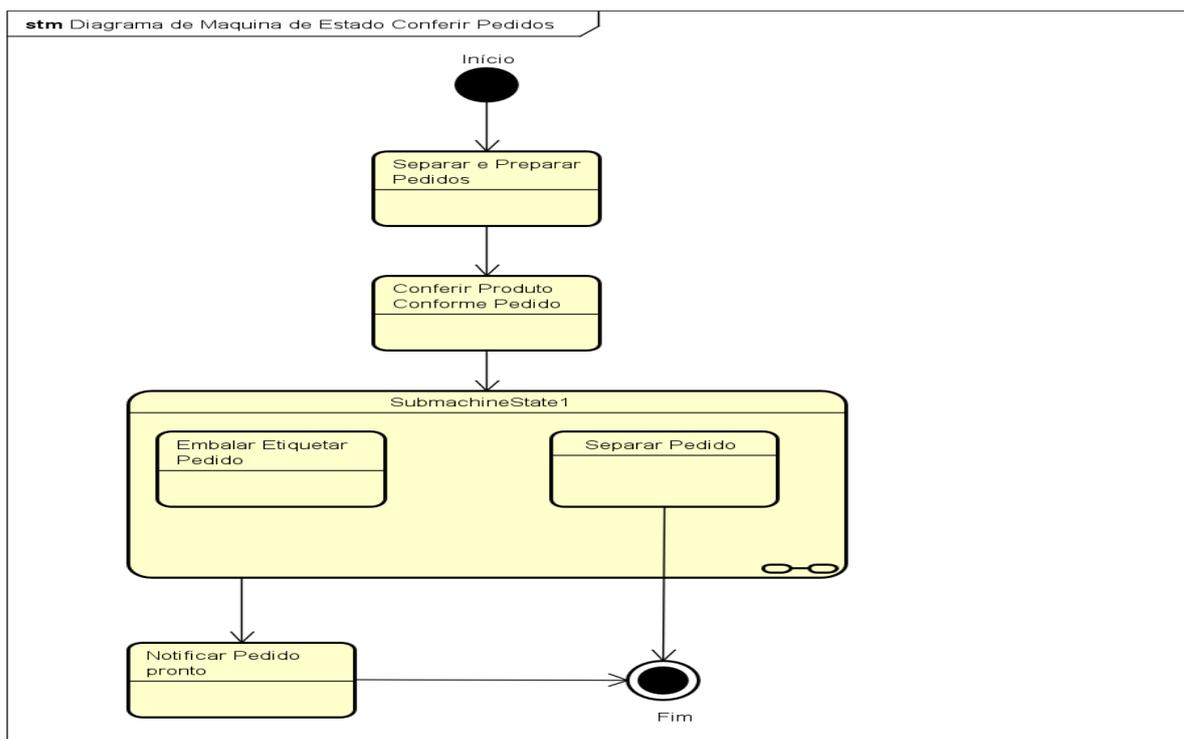
Fonte: Elaborado pelos autores

Figura 12 – Diagrama de Máquina de Estado Pagamentos Efetuados



Fonte: Elaborado pelos autores.

Figura 13 – Diagrama Máquina de Estado Conferir Pedido



Fonte: Elaborado pelos autores.

3.9 Diagrama de Sequência

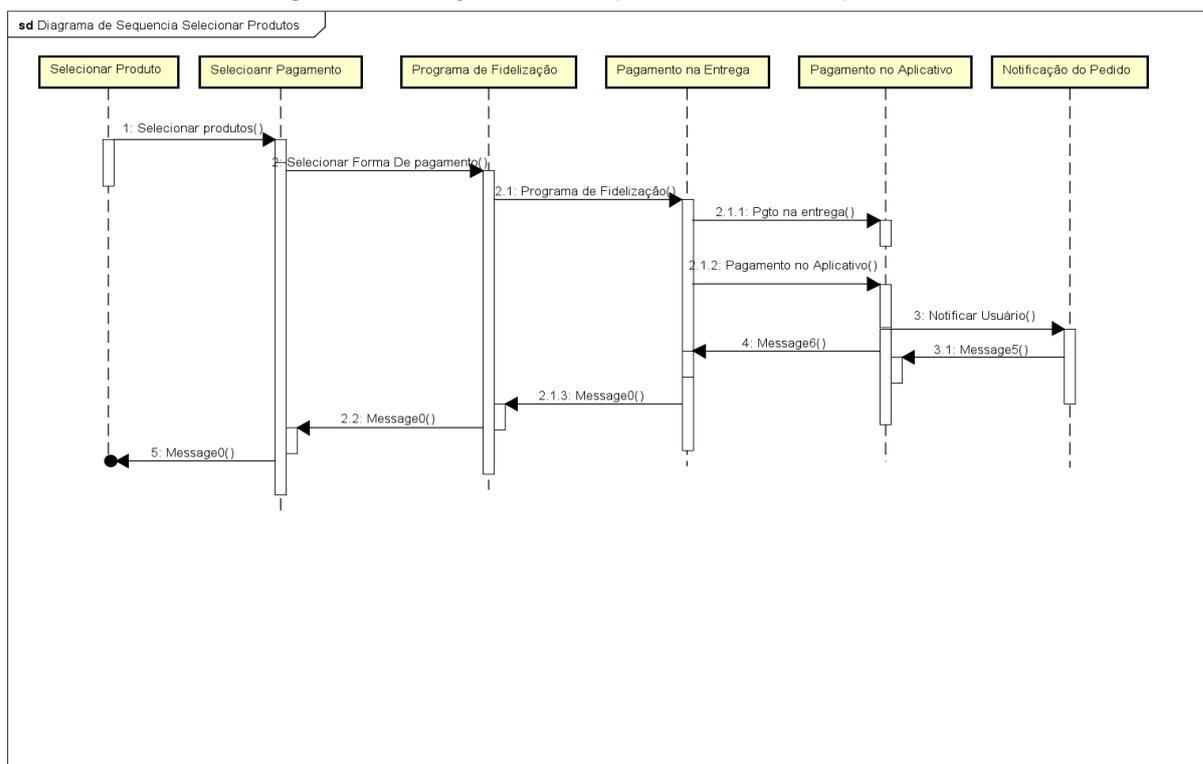
No contexto da engenharia de software, um Diagrama de Sequência, conforme definido por Sommerville (2019), é uma representação gráfica que descreve como as várias partes de um sistema interagem entre si ao longo do tempo.

Este tipo de diagrama pertence à linguagem de modelagem UML (*Unified Modeling Language*) e é amplamente utilizado na fase de modelagem de sistemas de software.

O Diagrama de Sequência é essencialmente uma ferramenta para visualizar e entender o comportamento dinâmico de um sistema, focando nas interações entre objetos individuais ou componentes de software em resposta a eventos específicos. Ele demonstra a ordem das mensagens trocadas entre esses objetos ao longo do tempo.

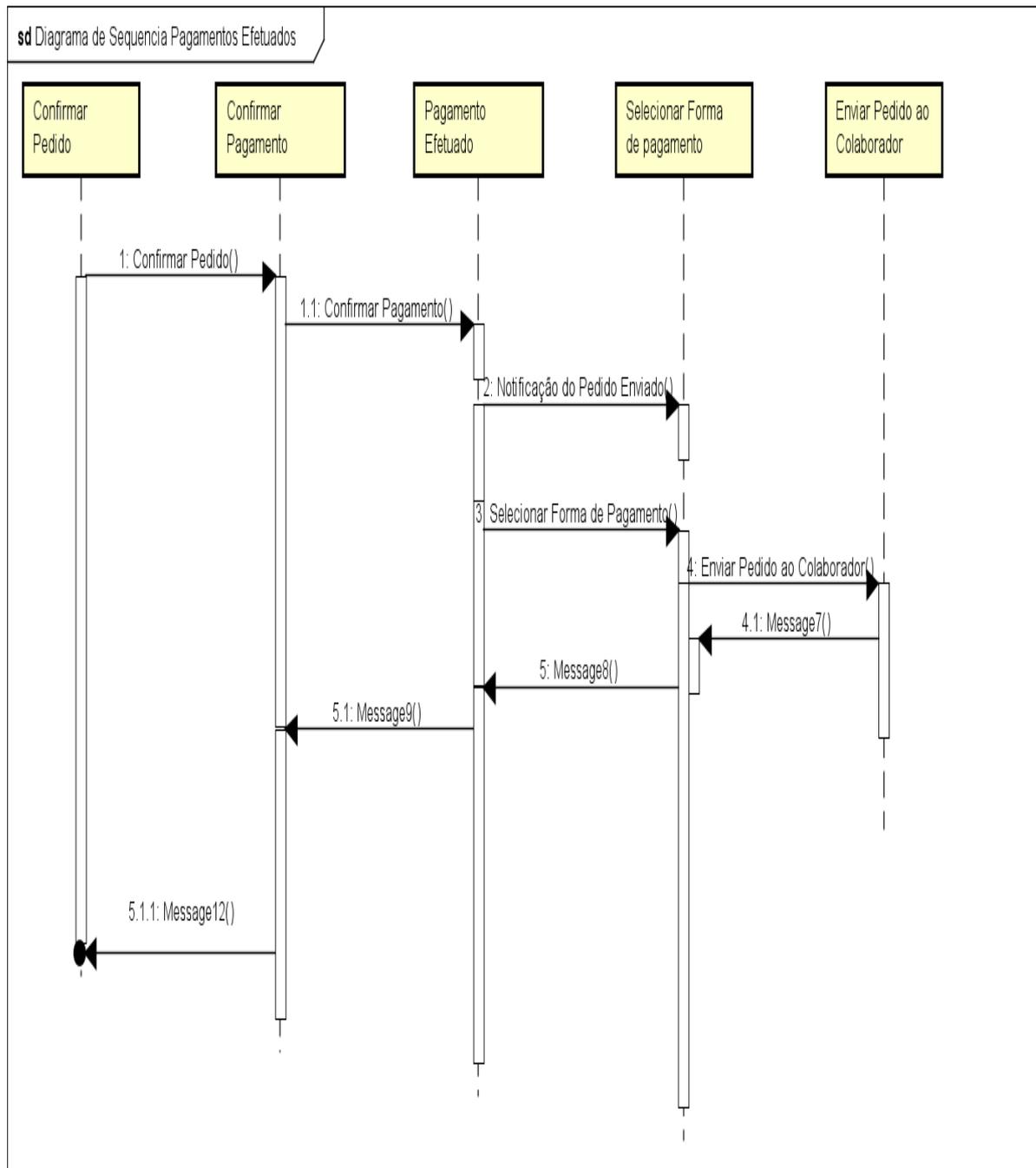
O Diagrama de Sequência é valioso para projetistas e desenvolvedores de software, pois ajuda a visualizar e entender como os componentes interagem durante a execução do sistema, contribuindo para uma implementação eficaz e uma comunicação mais clara entre os membros da equipe de desenvolvimento.

Figura 14 – Diagrama de Sequência Selecionar produtos



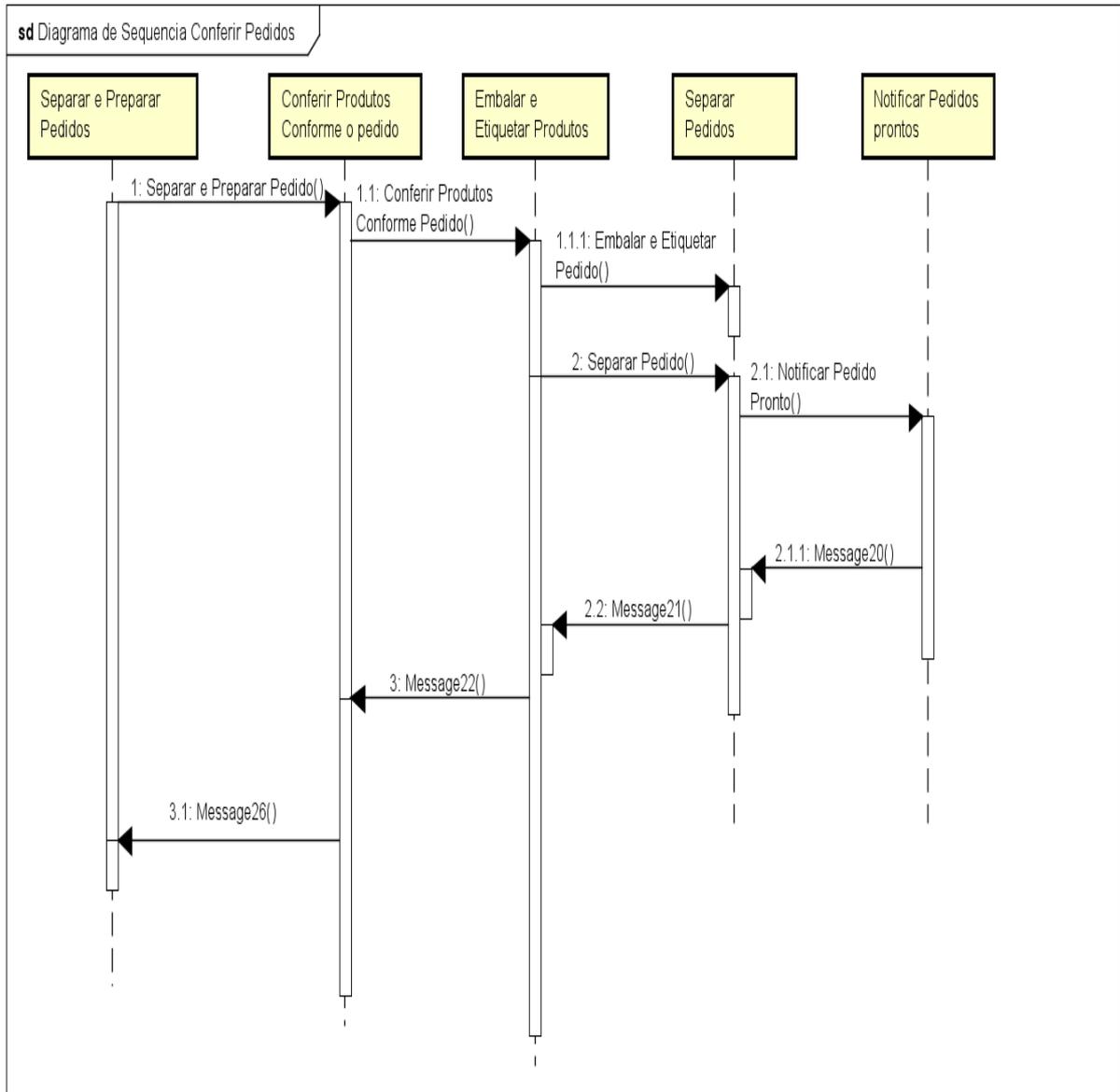
Fonte: Elaborado pelos autores

Figura 15 – Diagrama de Sequência Pagamentos Efetuados



Fonte: Elaborado pelos autores.

Figura 16 – Diagrama de Sequência Pagamentos Efetuados



Fonte: Elaborado pelos autores.

3.10 Matriz de Rastreabilidade

Para auxiliar no acompanhamento eficiente das etapas do desenvolvimento, a utilização da Matriz de Rastreabilidade é fundamental na rotina dos profissionais em engenharia de software. Seu uso visa garantir a consistência, qualidade e adesão aos requisitos do software. Assim sendo ela apoia as equipes no entendimento de como os requerimentos estão sendo atendidos ao longo das fases processuais bem como ajuda na identificação de lacunas potenciais ou erros nas especificações. Informações

cruciais estão presentes na matriz para indicar a proveniência dos requerimentos necessários.

Mais especificamente, consta nela certas particularidades que descrevem não só suas características inerentes, mas também questões nos quais existem ligações com outros componentes do sistema além das respectivas notas explicativas. De igual importância é sua função no gerenciamento de mudanças, pois ajuda a avaliar as consequências das modificações nos requisitos ao longo da construção do software.

Quadro 4 – RF x RN

	RF001	RF002	RF003	RF004	RF005	RF006	RF007
RN001	X	X	X				
RN002				X			
RN003	X				X		
RN004						X	
RN005							X

Fonte: Elaborado pelos autores.

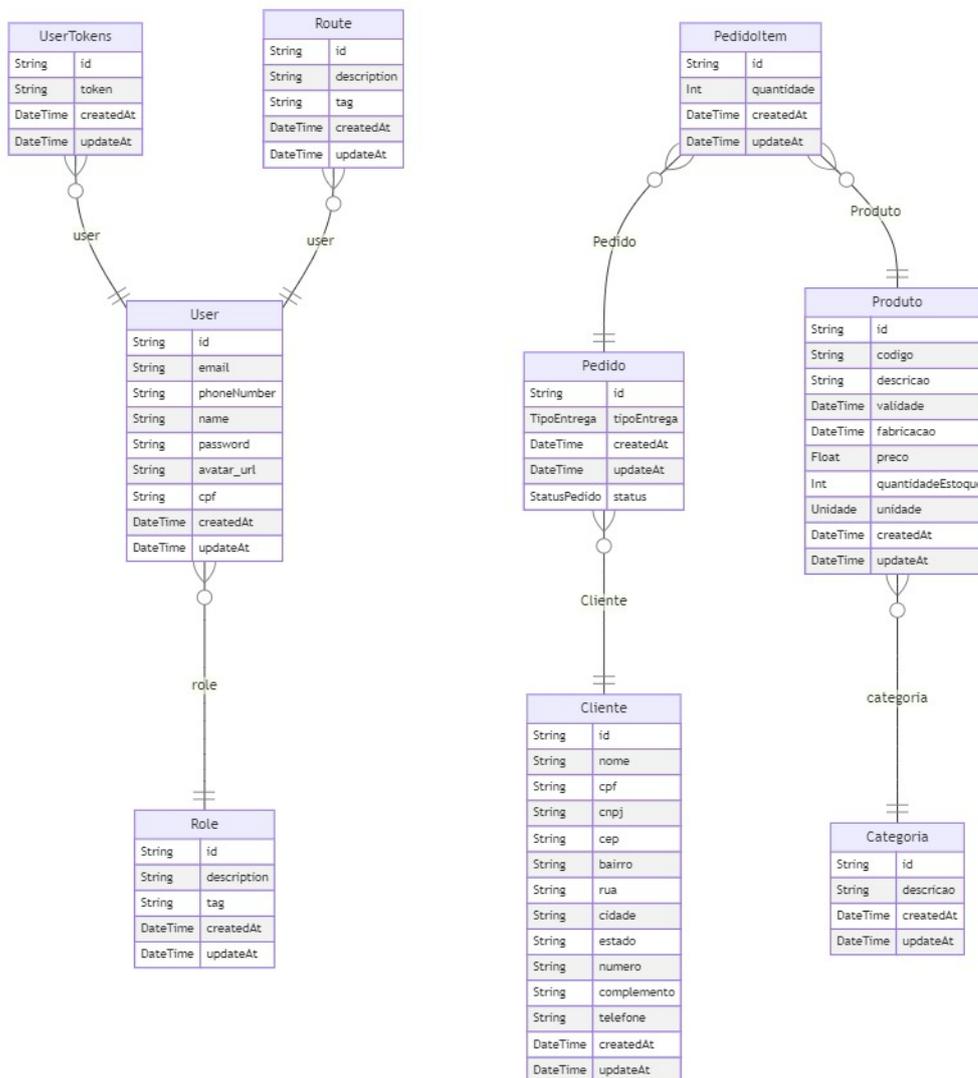
Quadro 5 – RF x UC

	RF001	RF002	RF003	RF004	RF005	RF006	RF007
UC001			X				
UC002	X	X			X	X	
UC003	X	X			X	X	
UC004	X	X			X	X	X
UC005	X	X			X	X	X
UC006	X	X			X	X	X
UC007	X	X			X	X	X
UC008							X

Fonte: Elaborado pelos autores.

3.11 DER

O diagrama de entidade-relacionamento (DER) é um tipo de diagrama de modelagem de dados que representa as entidades e relacionamentos de um sistema de informação. "As entidades são objetos do mundo real que são representados no sistema, como clientes, produtos ou pedidos. Os relacionamentos são as conexões entre as entidades, como a relação entre um cliente e um pedido." (Cockburn, Alistair, pg. 12).



4 Ferramentas e Métodos

Algumas ferramentas que foram utilizadas para desenvolvimento desta solução sistêmica:

- Teams é uma plataforma de colaboração e comunicação da Microsoft. Combina bate-papo, videoconferência, armazenamento de arquivos e integração de aplicativos em uma única plataforma.
- GitHub é uma ferramenta flexível e poderosa que pode ser usada para uma variedade de propósitos. Uma ferramenta de controle de versão, colaboração e desenvolvimento de software que seja fácil de usar, segura e integrada.
- O Figma é uma ferramenta de design gráfico e prototipagem baseada na web, é utilizada para criar uma variedade de designs, incluindo interfaces de usuário, sites, aplicativos móveis e produtos físicos. É uma ferramenta flexível e poderosa para designers que procuram uma ferramenta que seja fácil de usar, colaborativa e integrada.
- Canvas Dependendo da situação em que é usado, o termo Canvas pode se referir a várias ferramentas ou plataformas distintas. Há também uma versão mobile do Canvas disponível para os estudantes utilizarem. Canvas destaca-se como uma ferramenta poderosa no campo do design gráfico digital ao oferecer recursos intuitivos na produção personalizada desse tipo específico da arte visual.
- Bizagi é uma ferramenta de modelagem e automação de processos de negócios (BPM). Ele permite que as empresas visualizem, documentem e automatizem seus processos de negócios oferecendo uma variedade de recursos que o tornam uma ferramenta poderosa para BPM. Esses recursos incluem: Modelagem processos, automação de processos, gerenciamento de processos.
- O Astah é uma ferramenta de modelagem de software que permite aos desenvolvedores criar modelos de sistemas, processos e dados. Ele é baseado na notação UML (*Unified Modeling Language*), uma linguagem padrão para modelagem de software. O Astah oferece uma variedade de recursos que o tornam uma ferramenta poderosa para modelagem de software. É uma boa opção para desenvolvedores que procuram uma ferramenta de modelagem de software que seja fácil de usar, segura e integrada.

4.1 Para desenvolver o Front-end utilizamos essas ferramentas descritas:

- React é uma biblioteca JavaScript de código aberto para criar interfaces de usuário (UI). Ele é usado para criar aplicativos da web, aplicativos móveis e até

mesmo aplicativos de desktop. React é baseado em componentes, que são blocos de construção reutilizáveis que podem ser combinados para criar interfaces complexas. Os componentes são escritos em JavaScript e podem ser renderizados em qualquer navegador da web. React é uma biblioteca poderosa e flexível que pode ser usada para criar interfaces de usuário de alta qualidade. Ele é usado por empresas como Facebook, Instagram, Netflix e Airbnb.

- Node.js é um runtime JavaScript de código aberto, multiplataforma, baseado no interpretador V8 do Google e que permite a execução de códigos JavaScript fora de um navegador web. A principal característica do Node.js é sua arquitetura assíncrona e orientada por eventos. O Node.js é usado para criar uma variedade de aplicações, incluindo:
- TypeScript é uma linguagem de programação de código aberto que adiciona recursos como tipagem estática, interfaces, classes e módulos ao JavaScript. Ele ajuda a tornar o código mais legível, escalável e fácil de manter, e é amplamente utilizado em projetos de grande escala e em empresas renomadas. O TypeScript é compatível com o JavaScript e pode ser compilado em JavaScript, o que significa que pode ser executado em qualquer navegador ou ambiente que suporte JavaScript.

4.2 Para desenvolver o Back-end utilizamos essas ferramentas descritas:

- Node.js é uma plataforma de desenvolvimento do lado do servidor baseada no motor V8 do Google Chrome. Utilizando JavaScript como sua linguagem principal, o Node.js é conhecido por sua eficiência e escalabilidade, permitindo a construção de aplicações rápidas e leves. Sua arquitetura assíncrona e orientada a eventos torna o Node.js ideal para lidar com operações intensivas de I/O, proporcionando um desempenho excepcional. Além disso, o gerenciador de pacotes npm, integrado ao Node.js, facilita o compartilhamento e a instalação de bibliotecas e módulos.

Características principais:

- **Eficiência Assíncrona:** O Node.js é reconhecido por sua arquitetura assíncrona, permitindo a execução simultânea de operações sem bloqueio, o que resulta em aplicações mais rápidas e eficientes.
- **Baseado em JavaScript:** Node.js utiliza JavaScript, proporcionando uma experiência de desenvolvimento consistente tanto no lado do servidor quanto no navegador, facilitando a sincronização de código entre ambas as partes.
Ecosistema npm: O Node.js vem com o gerenciador de pacotes npm, que oferece um vasto repositório de bibliotecas e módulos prontos para uso, agilizando o desenvolvimento e favorecendo a reutilização de código.
- **Single Language Stack:** Com o Node.js, é possível utilizar JavaScript em todo o ecossistema de desenvolvimento, desde o lado do servidor até o cliente, simplificando a criação e manutenção de aplicações.
- **O PostgreSQL** é um poderoso sistema de gerenciamento de banco de dados relacional de código aberto. Reconhecido por sua confiabilidade, escalabilidade e extensibilidade, o PostgreSQL oferece suporte a uma ampla variedade de tipos de dados e recursos avançados, incluindo transações ACID. Sua comunidade ativa contribui para constantes melhorias, tornando-o uma escolha popular para aplicações que demandam robustez e desempenho. Além disso, a capacidade de estender suas funcionalidades por meio de extensões e procedimentos armazenados o torna flexível para atender a diversas necessidades de desenvolvimento.
- **Prisma** é uma poderosa ferramenta de banco de dados para Node.js e TypeScript, oferecendo um moderno *Object-Relational Mapping* (ORM). Com foco na simplicidade e segurança, o Prisma simplifica a interação com bancos de dados, proporcionando uma abstração de alto nível. Seu design intuitivo facilita a modelagem de dados e consultas, enquanto sua integração perfeita com o PostgreSQL garante desempenho e confiabilidade.

5 Desenvolvimento

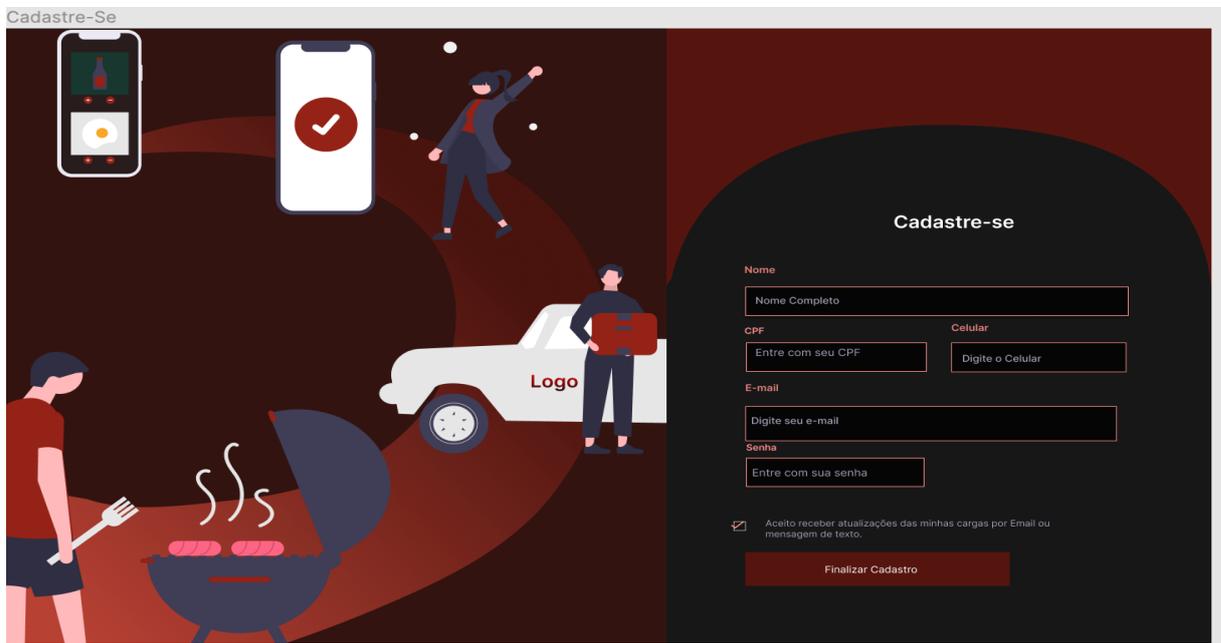
Na construção desta solução sistêmica, optamos por uma estrutura que reflete a modularidade e escalabilidade necessárias para o sucesso a longo prazo.

Em relação a este desenvolvimento, o software de gestão empresarial integrado tornou-se uma ferramenta fundamental. Este estudo examina os resultados que podem ser alcançados através da implementação e utilização eficaz deste software e examina os seus possíveis efeitos positivos nas áreas de registo de clientes, gestão de estoques e gestão de encomendas.

A modelagem de software baseada na experiência do usuário permite uma arquitetura intuitiva para garantir a eficiência operacional. Um sistema modular é explorado para garantir flexibilidade e escalabilidade, permitindo fácil integração de novos recursos. O esquema de cores foi escolhido de forma a refletir a identidade da marca do açougue e ao mesmo tempo proporcionar uma experiência visual agradável aos usuários. Usar cores contrastantes para marcar áreas críticas, como pedidos em atraso e usar cores suaves para elementos informativos pode melhorar a compreensão e facilitar a navegação.

Seguem três códigos relevantes desta solução sistêmica:

1 - Tela Cadastro de Clientes de Login na plataforma



```
import * as S from "../Cadastrar.module";
import churras from "../../assets/imagem_churrasqueira.png";
import logo from "../../assets/imagem_logo_carro.png";
import celularOk from "../../assets/ingem_celular.png";
import { Link } from "react-router-dom";
import { InputMask, InputMaskChangeEvent } from "primereact/inputmask";
```

```

import { useState } from "react";

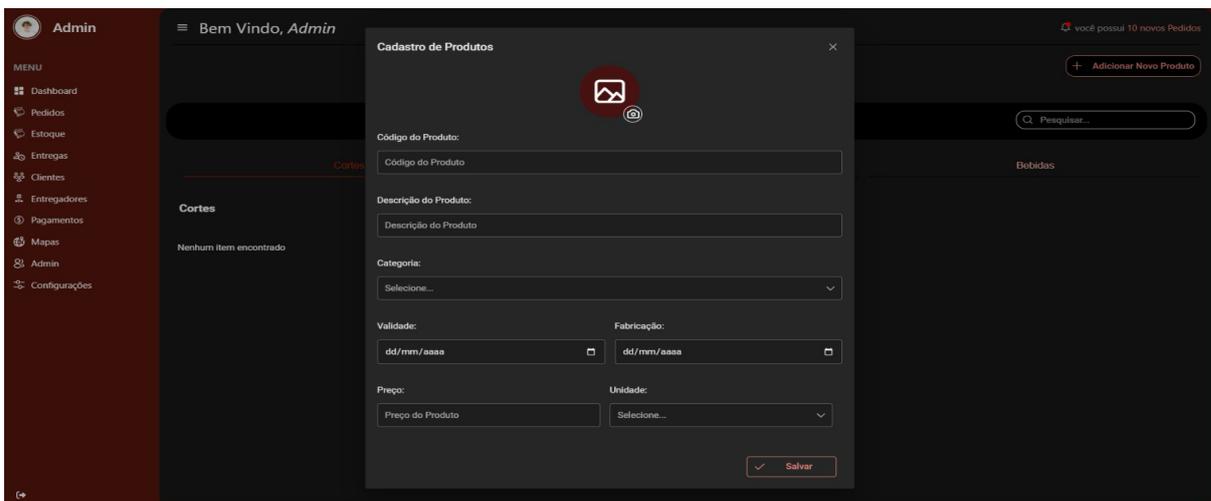
export function Cadastrar() {
  const [formData, setFormData] = useState({
    nome: "",
    cpf: "",
    celular: "",
    email: "",
    password: "",
  });

  const handleInputChange = (e: React.ChangeEvent<HTMLInputElement>)
=> {
    const { name, value } = e.target;
    setFormData((prevFormData) => ({

```

Para acessar o programa todo usuário deverá ter cadastro, se não será redirecionado para a página de cadastro. Na tela é utilizado o módulo de rotas do React chamado React Router Dom utilizado para fazer o link posterior ao cadastro a plataforma redirecionando o usuário.

2 - Tela Cadastro de Produtos no Estoque



```

<<<S.DivAll>
  <S.EstoqueTitle>
    <S.ButtonContainer>
      <S.StyledButtonCreate onClick={() => setVisible(true)} label="Adicionar
Novo Produto" icon="pi pi-plus" />
    </S.ButtonContainer>
    <div className="card">
      <S.Filter>
        <span className="p-input-icon-left" style={{
          height: "35px",
        }}>
          <i className="pi pi-search" />
          <S.CustomInputText
            type="text"

```

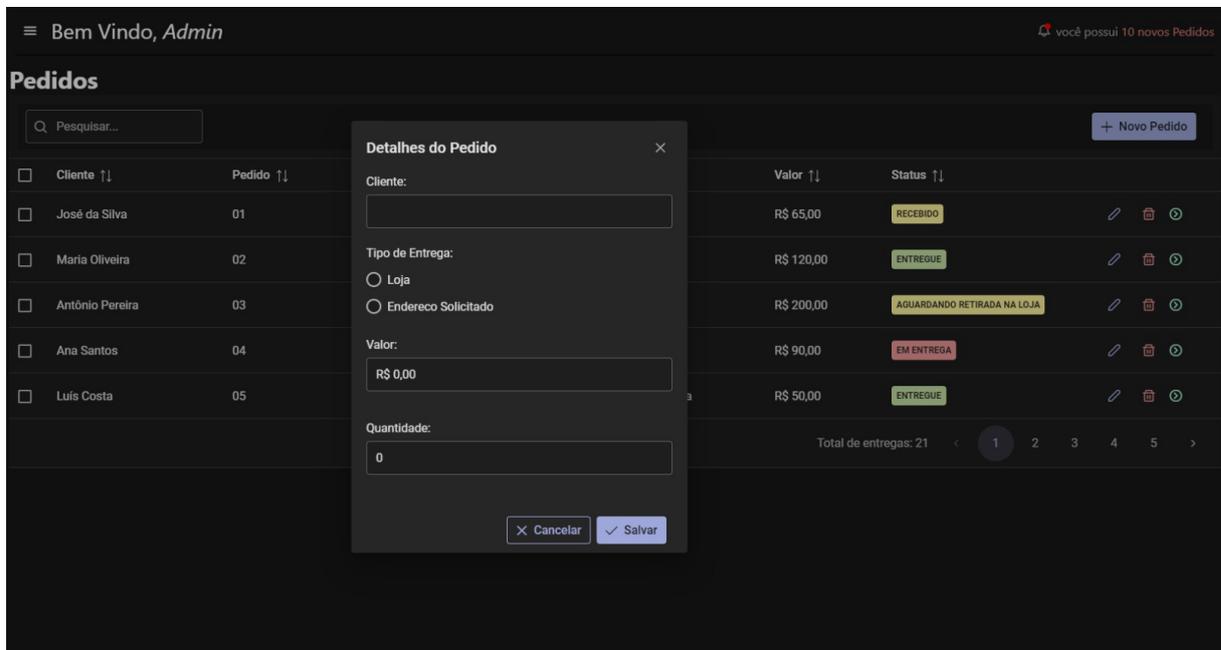
```

placeholder="Pesquisar..."
value={searchTerm}
onChange={handleSearch} />
</span>
</S.Filter>
<S.TabPanel>
  {tabs.map((tab) => (
    <S.TabItem
      key={tab.id}
      active={activeTab === tab.name}
      onClick={() => setActiveTab(tab.name)}
    >
      {tab.name}
    </S.TabItem>
  )}

```

Usando uma biblioteca de componentes chamada Prime React foi realizada a criação da tela de estoque, foi desenvolvida usando a *Library* de estilização *styled components*. A modal em questão construído a partir do componente *Dialog* realiza o inventário e criação de novos produtos na plataforma. O código apresentado segue os conceitos de boas práticas ao reutilizar os componentes e estilizações semelhantes prezando por um projeto semântico e bem estruturado.

3 - Tela de Pedidos



```

<Dialog
  visible={productDialog}
  style={{ width: "28rem" }}
  breakpoints={{ "960px": "75vw", "641px": "90vw" }}

```

```

header="Detalhes do Pedido"
draggable={false}
modal
className="p-fluid"
footer={productDialogFooter}
onHide={hideDialog}
>

<div className="field">
  <label htmlFor="clientName" className="font-bold">
    Cliente:
  </label>
  <InputText
    id="clientName"
    style={{
      marginBottom: "0.5rem",
      marginTop: "0.5rem",
      width: "100%",
    }}
    value={product.clientName}
    onChange={(e) => onInputChange(e, "clientName")}
    required
    autoFocus
    className={classNames({
      "p-invalid": submitted && !product.clientName,
    })}
  />
  {submitted && !product.clientName && (
    <small className="p-error">
      É necessário preencher o nome do cliente.
    </small>
  )}
</div>

<div className="field" style={{
  marginBottom: "1rem",
  marginTop: "1rem",
  width: "100%",
}}>
  <label className="mb-3 font-bold" style={{
    marginBottom: "2rem",
    width: "100%",
  }}>Tipo de Entrega:</label>

  <div className="formgrid grid" style={{
    marginBottom: "0.5rem",
    marginTop: "1rem",
    width: "100%",
  }}>

```

```

    <div className="field-radiobutton col-6" style={{ marginBottom:
"0.5rem", marginTop: "0.5rem" }}>
      <RadioButton
        inputId="entrega1"
        name="entrega"
        value="Loja"
        onChange={onCategoryChange}
        checked={product.category === "Loja"}
      />
      <label htmlFor="entrega1" style={{
        marginLeft: "0.5rem",
        width: "100%",
      }}>Loja</label>
    </div>

```

A tela de pedidos representa o *workflow* principal do sistema, este do qual realiza a intersecção entre produto e cliente. Nela consta uma tabela com a listagem de pedidos bem como o status de cada entrega. Ela foi desenvolvida utilizando os componentes data table, button e column da *Library Prime React*. No Dialog de criação de pedidos é utilizado classes e renderização condicional a fim de permitir uma maior reutilização e aproveitamento das informações dispostas.

6 Resultados e Discussão

A constante evolução da sociedade, o uso da tecnologia tem sido crucial na vida de todo ser humano. As butiques de carnes não estão isentas desse impacto.

Cientes do contexto atual, onde os processos exigem qualidade e agilidade, este projeto desenvolveu uma solução sistêmica que aborda de maneira eficaz o desafio da gestão de uma butique de carnes, proporcionando uma solução transparente e organizada.

Esta solução abrange e permite que a butique tenha eficiência nas atividades de controle dos seus processos, visando auxiliar e organizar as rotinas das compras de carnes feitas online. Melhorando assim, a experiência do cliente, facilitando os processos e o controle da empresa em relação ao seu estoque, vendas e entregas. O software será desenvolvido pensando na experiência do usuário, deixando a interface intuitiva e descomplicada.

O sistema deverá conter tais funcionalidades: Tipos de usuários; Cadastros de usuários e cliente; Cadastro de produtos; Consultas de produtos; Rastreabilidade do produto; ajuda e suporte/dúvida ao cliente.

A implementação do software na boutique de carnes acarretará uma melhoria substancial na organização do gerenciamento de pedidos, substituindo as atividades manuais anteriormente registradas em cadernos e mitigando, assim, o risco de perda de informações. Com a funcionalidade do software para o registro de pedidos, oferecemos opções de status, como "recebidos", "em produção", "aguardando entrega" e "pedido entregue". Além disso, os motoristas terão acesso aos seus pedidos para entrega e poderão, por meio da versão mobile, atualizar o status do pedido para "entregue" à medida que realizam a entrega.

Considerações finais

Ao longo do desenvolvimento desta solução sistêmica, desde agosto de 2022, buscamos atender aos objetivos iniciais estabelecidos para o desenvolvimento do sistema de gestão de uma boutique de carnes.

O principal desafio foi a complexidade dos processos e a organização dos dados sobre o corte das carnes. Embora tenhamos implementado algumas telas, é importante ressaltar que construímos a base sólida da solução.

Visando a liberação do uso desta solução, toda a infraestrutura de desenvolvimento foi implementada e as tecnologias escolhidas provaram ser sólidas e eficazes. Essas tecnologias ofereceram uma série de recursos e facilidades modernas e escaláveis que contribuíram para o sucesso do desenvolvimento de um Sistema de Gestão Integrada para uma boutique de carnes.

Com o conjunto de dados armazenados no banco de dados no futuro pretendemos desenvolver algumas novas funcionalidades, valorizando a fidelização dos clientes.

REFERÊNCIAS

AURÉLIO, Dicionário. 1999

HAMMER E CHAMPY. Reengenharia Revolucionando a Empresa em função da concorrência e das grandes mudanças da gerência. (1994),

KOTLER; KELLER. Administração de Marketing. 2012, p. 49

LUCIDCHART. O que é BPMN? Disponível em: <https://www.lucidchart.com/pages/pt/o-que-e-bpmn>. Acesso em: 10 out. 2023.

LUCIDCHART. O que é um diagrama de máquina de estados? Disponível em: <https://www.lucidchart.com/pages/pt/o-que-e-diagrama-de-maquina-de-estados-uml>. Acesso em: 10 out. 2023.

LUCIDCHART. O que é um diagrama de sequência UML? Disponível em: <LUCIDCHART. Disponível em: Acesso em: 10 out. 2023. >. Acesso em: 10 out. 2023.

MINAS, S. O QUE É BUSINESS MODEL CANVAS E COMO APLICÁ-LO NO SEU NEGÓCIO? Disponível em: <https://inovacaosebraeminas.com.br/o-que-e-business-model-canvas-e-como-aplica-lo-no-seu-negocio/>. Acesso em: 10 out. 2023.

SOMMERVILLE, Ian. Engenharia de Software. (2019)

SQL Guia Prático - 2a edição. [s.l.] Brasport, [s.d.].

REPOSITÓRIO INSTITUCIONAL DA UFMG: Home. Disponível em: <<https://repositorio.ufmg.br/>>. Acesso em: 10 out. 2023.