



UNIVERSIDADE FEDERAL DO CEARÁ
CENTRO DE CIÊNCIAS AGRÁRIAS
DEPARTAMENTO DE ENGENHARIA DE PESCA

LORENA MARTINS SANTIAGO

USO E POTENCIALIDADES DE FERRAMENTAS DE *BUSINESS INTELLIGENCE*
NA GESTÃO DE MICRO E PEQUENAS EMPRESAS DO SETOR DE PESCADOS

FORTALEZA

2024

LORENA MARTINS SANTIAGO

USO E POTENCIALIDADES DE FERRAMENTAS DE *BUSINESS INTELLIGENCE* NA
GESTÃO DE MICRO E PEQUENAS EMPRESAS DO SETOR DE PESCADOS

Trabalho de conclusão de curso apresentado ao Curso de Graduação em Engenharia de Pesca, da Universidade Federal do Ceará, como requisito parcial à obtenção do grau de Bacharel em Engenharia de Pesca.

Orientador: Prof. Dr. Bartolomeu Warlene Silva de Souza.

FORTALEZA

2024

Dados Internacionais de Catalogação na Publicação
Universidade Federal do Ceará
Sistema de Bibliotecas

Gerada automaticamente pelo módulo Catalog, mediante os dados fornecidos pelo(a) autor(a)

S226u Santiago, Lorena Martins.
Uso e potencialidades de ferramentas de business intelligence na gestão de micro e pequenas empresas do setor de pescados / Lorena Martins Santiago. – 2024.
47 f. : il. color.

Trabalho de Conclusão de Curso (graduação) – Universidade Federal do Ceará, Centro de Ciências Agrárias, Curso de Engenharia de Pesca, Fortaleza, 2024.
Orientação: Prof. Dr. Bartolomeu Warlene Silva de Souza.

1. Informação para negócios. 2. Mensuração de desempenho. 3. Gestão. I. Título.

CDD 639.2

LORENA MARTINS SANTIAGO

USO E POTENCIALIDADES DE FERRAMENTAS DE *BUSINESS INTELLIGENCE*
NA GESTÃO DE MICRO E PEQUENAS EMPRESAS DO SETOR DE PESCADOS

Trabalho de conclusão de curso apresentado ao Curso de Graduação em Engenharia de Pesca, da Universidade Federal do Ceará, como requisito parcial à obtenção do grau de Bacharel em Engenharia de Pesca.

Aprovada em: 23/09/2024.

BANCA EXAMINADORA

Prof. Dr. Bartolomeu Warlene Silva de Souza (Orientador)
Universidade Federal do Ceará (UFC)

Prof. Dr. Raimundo Nonato de Lima Conceição
Universidade Federal do Ceará (UFC)

Prof. Dra. Francisca Gleire Rodrigues de Menezes
Universidade Federal do Ceará (UFC)

RESUMO

A abertura de pequenos negócios vem crescendo nos últimos anos e para se obter sucesso no mercado atual, as empresas necessitam aumentar o uso de informações externas e também informações internas, tecnologia e otimização de processos para conseguirem se manter competitivas. A gestão de uma empresa do setor de pescados envolve o manejo de diversas informações complexas e dinâmicas. Diante desse cenário, ferramentas de gestão orientadas pela tecnologia, conhecidas como *Business Intelligence* (BI), foram desenvolvidas com o intuito de fornecer uma visão sistêmica do negócio, auxiliando na coleta, análise e distribuição de dados, transformando grandes volumes de informações em *insights* valiosos e de qualidade para a tomada de decisões estratégicas. O uso de BI permite que empresas do setor de pescados otimizem suas operações, melhorem a eficiência e se adaptem rapidamente às mudanças do mercado, garantindo uma gestão mais eficaz e competitiva. O presente trabalho propõe um modelo para a gestão de dados de micro e pequenas empresas do setor de pescados e analisa os dados coletados de microempresas do setor de pescados aplicando técnicas de *Business Intelligence*, a fim de extrair conhecimento capaz de auxiliar no gerenciamento das organizações. Esse trabalho iniciou com a coleta de dados referentes à venda e estoque de duas empresas de comércio de pescado do Ceará. O método utilizado para a realização do processo de *Extract, Transform, and Load* (ETL) foi desenvolvido em linguagem *python*, utilizando as bibliotecas *pandas* e *numpy*, os dados foram armazenados no banco de dados *BigQuery* e com base nos *Key Performance Indicator* (KPIs) desenvolvidos, os *dashboards* foram construídos na ferramenta de visualização de dados *Looker Studio*, assim foi possível a exploração e apresentação dos dados. Os resultados obtidos demonstram que a metodologia adotada foi eficaz no suporte à extração de informações úteis, na forma de painéis e gráficos, assim como na descoberta de conhecimento capaz de apoiar no processo decisório, contribuindo para a gestão das organizações.

Palavras-chave: Gestão; Informação para negócios; Mensuração de desempenho.

ABSTRACT

The opening of small businesses has been growing in recent years and to be successful in the current market, companies need to increase the use of external information and also internal information, technology and process optimization to remain competitive. Managing a company in the fish sector involves handling a variety of complex and dynamic information. Given this scenario, technology-driven management tools, known as Business Intelligence (BI), were developed with the aim of providing a systemic view of the business, assisting in the collection, analysis and distribution of data, transforming large volumes of information into valuable insights. and quality for strategic decision-making. The use of BI allows companies in the fish sector to optimize their operations, improve efficiency and quickly adapt to market changes, ensuring more effective and competitive management. The present work proposes a model for data management of micro and small companies in the fish sector and analyzes the data collected from micro companies in the fish sector applying Business Intelligence techniques, in order to extract knowledge capable of assisting in the management of organizations. This work began with the collection of data relating to the sale and stock of two fish trading companies in Ceará. The method used to carry out the Extract, Transform, and Load (ETL) process was developed in Python language, using the pandas and numpy libraries, the data was stored in the BigQuery database and based on the Key Performance Indicator (KPIs) developed, the dashboards were built using the Looker Studio data visualization tool, making it possible to explore and present the data. The results obtained demonstrate that the methodology adopted was effective in supporting the extraction of useful information, in the form of panels and graphs, as well as in the discovery of knowledge capable of supporting the decision-making process, contributing to the management of organizations.

Keywords: Management; Information for businesses; Performance measures.

LISTA DE FIGURAS

Figura 1 – Evolução do BI	13
Figura 2 – Processo de ETL.....	21
Figura 3 – Dashboard de vendas da loja Camocim no Looker Studio.....	25
Figura 4 – Visão diária do gráfico de faturamento da loja Camocim no Looker Studio....	26
Figura 5 – Dashboard de estoque da loja Camocim no Looker Studio.....	27
Figura 6 – Dashboard de vendas da loja M.A.C. no Looker Studio.....	28
Figura 7 – Dashboard de estoque da loja M.A.C. no Looker Studio.....	29
Figura 8 – Modelo conceitual para a gestão de dados de micro e pequenas empresas.....	30

LISTA DE TABELAS

Tabela 1 – KPIs da loja Camocim	24
Tabela 2 – KPIs da loja M.A.C.....	25
Tabela 3 – Planilha padrão utilizada para coletar os dados de venda.....	34
Tabela 4 – Planilha padrão utilizada para coletar os dados de estoque.....	34
Tabela 5 – Dados de vendas da loja Camocim no Bigquery.....	45
Tabela 6 – Dados de vendas da loja M.A.C. no Bigquery.....	46
Tabela 7 – Dados de estoque da loja Camocim no Bigquery.....	46
Tabela 8 – Dados de estoque da loja M.A.C. no Bigquery.....	47

SUMÁRIO

1 INTRODUÇÃO	10
1.1 Objetivos	12
2 REVISÃO BIBLIOGRÁFICA	12
2.1 Business Intelligence	12
2.2 Arquitetura de um sistema de Business Intelligence	14
2.3 Banco de dados	14
2.4 Metadados	17
2.5 Data Warehouse	17
2.6 Data marts	19
2.7 Data Mining	19
2.8 Online Analytical Processing (OLAP)	19
2.9 Extract, Transform and Load	20
2.10 Key Performance Indicators	21
2.11 Python	22
2.12 BigQuery	22
2.13 Looker Studio	23
3 MATERIAL E MÉTODOS	23
3.1 Coleta	23
3.2 Criação do Data Warehouse	23
4 RESULTADOS	24
4.1 KPIs	24
4.2 Dashboard	25
4.3 Gestão dos dados	29
5 CONCLUSÃO	31
REFERÊNCIAS	32
APÊNDICE A - PLANILHA BASE PARA COLETA DE DADOS	34
APÊNDICE B - SCRIPT PYTHON	35
APÊNDICE C - QUERIES SQL	43
APÊNDICE D - DADOS ESTRUTURADOS NO BIGQUERY	45
APÊNDICE E - ACESSO AOS DASHBOARDS	48

1 INTRODUÇÃO

As micro e pequenas empresas representam cerca de 93,7% do total das empresas brasileiras (BRASIL, 2023). São responsáveis por 52% dos empregos formais e por 27% do Produto Interno Bruto – PIB do país. As micro e pequenas empresas são as principais geradoras de riqueza no comércio no Brasil, respondem por 53,4% do PIB deste setor. No PIB da Indústria, a participação das micro e pequenas (22,5%) já se aproxima das médias empresas (24,5%). E no setor de Serviços, mais de um terço da produção nacional (36,3%) têm origem nos pequenos negócios (SEBRAE, 2023).

A abertura de pequenos negócios vem batendo recordes nos últimos anos, o levantamento divulgado pelo Serviço Brasileiro de Apoio às Micro e Pequenas Empresas – SEBRAE, mostra que em 2021, mais de 3,9 milhões de empreendedores formalizaram micro e pequenas empresas (MPEs) ou se registraram como microempreendedores individuais (MEIs). O número representa crescimento de 19,8% em relação a 2020, quando foram abertos 3,3 milhões de negócios. Em relação a 2018, a expansão chega a 53,9%. Naquele ano, foram criados 2,5 milhões de cadastros nacionais de pessoas jurídicas (SEBRAE, 2022).

Diante desse cenário para-se obter sucesso no mercado atual, as empresas necessitam aumentar o uso de informações externas (de mercado, concorrentes, clientes, fornecedores, etc.) e também informações internas (operações, transações, etc.), assim como tecnologia e processos para conseguirem se manter competitivas. No âmbito estratégico-empresarial, a informação terá mais ou menos valor não só dependendo da sua aplicação, mas, principalmente, da maneira que for interpretada, gerando novos conhecimentos que irão agregar valor ao negócio. Partindo dessas definições, será a habilidade com que a empresa coleta, organiza, analisa e implementa mudanças a partir de informações, integrando-as ao processo de melhoria contínua de suas atividades, que irá determinar a sua excelência (REZENDE, 2002).

A informação é um dos recursos mais importantes, cuja gestão e aproveitamento mais influencia o sucesso das organizações, a informação também é considerada e utilizada em muitas organizações como um fator estruturante e um instrumento de gestão da organização e como uma arma estratégica indispensável para a obtenção de vantagens competitivas (AMARAL, 1994).

As empresas com maior poder de sobrevivência estão atentas às mudanças ocorridas em seu entorno, adquirindo flexibilidade e capacidade de mudança frente a um ambiente em constante modificação e constroem uma identidade própria, reconhecida por meio de suas ações perante o mercado e a sociedade, estabelecendo

relacionamentos construtivos. Apresentam independência e capacidade no direcionamento do seu crescimento e evolução, não ficando dependentes somente do governo e de suas políticas públicas. Toda empresa é capaz de produzir conhecimento, porém ela só será eficiente se for capaz de transformar esse conhecimento em ação. Todo o esforço de desenvolver e implantar uma grande base de informações não trará resultados, caso esta base não seja constantemente atualizada, visto que seu valor está inteiramente ligado ao seu grau de atualização, assim como à gestão dos conteúdos proporcionando contextualização e propósito aos dados. O conhecimento gerado internamente à organização, quando é objeto de gestão, subsidia as ações e os processos estratégicos que auxiliam a organização na sua atuação no mercado (MOLINA; VALENTIM, 2015).

O recurso da informação deve ser considerado como sendo de grande potencial de retorno às empresas, logo é justificável o investimento em ferramentas de tecnologia da informação (TI), visto que terão custo-benefício no longo prazo, a medida que as decisões forem sendo tomadas, acarretando em economia, redução de desperdícios, aumento de margem e conseqüentemente lucro (MCGEE; PRUSAK, 1994).

Para conseguir um ambiente eficaz de gestão, referente à mensuração do desempenho, faz-se necessário incorporar um sistema de medidas que assegure o alinhamento das atividades com o objetivo maior da organização (PACE; BASSO; SILVA, 2003).

O setor de pescados representa uma oportunidade para as micro e pequenas empresas. A carne de pescado apresenta um grande potencial de mercado por ser um alimento rico em proteínas, de sabor agradável e fácil preparo. A demanda mundial por comércio de pescado vem aumentando nas últimas décadas, principalmente em função do crescimento da população e pela procura dos consumidores por alimentos mais saudáveis. Dados recentes publicados pela Organização das Nações Unidas para a Alimentação e a Agricultura – FAO apontam que a produção de pescados alcançou, em 2020, o patamar de 177,8 milhões de toneladas. Dos diversos tipos de uso que se tem para produtos da pesca e da aquicultura, a FAO estima que 157,4 milhões de toneladas tenham sido destinadas ao consumo direto humano, o que corresponde a 20,2 kg de pescados per capita (FAO, 2022).

A gestão de uma empresa do setor de pescados envolve o manejo de diversos dados. As ferramentas de gestão orientadas pela tecnologia denominadas de *Business Intelligence* – BI, foram criadas com o intuito de fornecer uma visão sistêmica do negócio, ajudando na distribuição dos dados, etendo como seu principal objetivo transformar grandes quantidades de dados em informações de qualidade para a tomada de decisão. Pode-se, utilizando essas ferramentas, atuar no cruzamento de dados,

visualizando a informação de diversas formas para analisar os principais indicadores de desempenho da organização (BATISTA, 2004).

1.1 Objetivos

Diante desse contexto, o presente trabalho tem como objetivo fazer um estudo de caso de microempresas do setor de pescada de Fortaleza, Ceará. Os dados foram coletados no período de maio a outubro de 2022 e fornecidos pelas próprias empresas.

Este trabalho tem como objetivo:

- Analisar os dados coletados de microempresas do setor de pescadas aplicando técnicas de *Business Intelligence*, a fim de extrair conhecimento capaz de auxiliar no gerenciamento da organização.
- Construir um modelo padrão para a gestão de dados de micro e pequenas empresas do setor de pescadas.

2 REVISÃO BIBLIOGRÁFICA

Neste capítulo serão apresentados, com base em uma revisão de literatura, alguns conceitos importantes relacionados ao processo de implementação de uma solução de *Business Intelligence*, bem como definições de seus componentes de extração, tratamento, integração, armazenamento e exploração de dados.

2.1 *Business Intelligence*

Business Intelligence (BI) é um termo “guarda-chuva” que engloba ferramentas, arquitetura, bases de dados, *data warehouse*, gerenciamento de desempenho, metodologias e assim sucessivamente, tudo integrado em uma suíte de *software*. O objetivo do *software* é de possibilitar que os gerentes de negócios e analistas em uma empresa acessem qualquer dado da empresa de maneira fácil e rápida, possivelmente em tempo real, bem como conduzir manipulações e análises apropriadas. Analisando dados históricos e atuais, situações, métricas e desempenhos, os tomadores de decisões obtêm valiosos *insights* que os ajudam a tomar melhores decisões. O processo do BI baseia-se na transformação de dados em informações, depois em decisões e finalmente em ações.

O termo BI foi criado por uma empresa de consultoria chamada *Gartner Group*

em meados da década de 1990. Contudo, o conceito iniciou antes, com suas raízes nos sistemas de geração de relatórios SIG (Sistema de Informações Gerenciais) dos anos 1970. Durante esse período, os sistemas de geração de relatórios eram estáticos, bidimensionais e não possuíam recursos de análise. No início dos anos 1980, surgiu o conceito de sistemas de informações executivas (EIS). Esse conceito expandiu o suporte computadorizado aos gerentes e executivos de nível superior. Alguns dos recursos introduzidos foram sistemas de geração de relatórios dinâmicos multidimensionais, prognósticos e previsões, análise de tendências, detalhamento, acesso a status e fatores críticos de sucesso. Esses recursos aparecem em dezenas de produtos comerciais até o meio da década de 1990. Depois, os mesmos recursos e alguns recursos novos apareceram sob o nome BI. Atualmente, se reconhece que todas as informações de que os executivos necessitam podem estar em um bom sistema de informações empresariais baseado em BI. Em 2005, os sistemas de BI começaram a incluir o recurso de inteligência artificial, bem como outros recursos poderosos de análise (TURBAN et al., 2009).

Figura 1 – Evolução do BI.



Fonte: Adaptado de TURBAN et al. (2009).

Um sistema de BI deve tornar a informação facilmente acessível e compreensível. Os dados devem ser intuitivos e óbvios para o usuário empresarial. As estruturas dos dados e os rótulos devem imitar os processos de pensamento e o vocabulário dos usuários e devem apresentar informações de forma consistente. Os dados devem ser cuidadosamente reunidos a partir de uma variedade de fontes, tratados e de qualidade garantida e liberado somente quando for adequado para consumo do usuário, deve servir como base autorizada e confiável para uma melhor tomada de decisão, deve se

adaptar às mudanças e necessidades do usuário, condições de negócios. O sistema BI deve controlar efetivamente o acesso às informações confidenciais da organização (KIMBALL; ROSS, 2013).

2.2 Arquitetura de um sistema de *Business Intelligence*

A arquitetura mais comum e melhor adaptável que compõe um sistema de BI é formada por quatro componentes:

- Fontes de dados: sistemas de onde se originam os dados da organização, podendo estar em diversos formatos e lugares, havendo a necessidade de integração entre elas.
- Processo de *Extract, Transform and Load* – ETL: consiste na etapa de extração dos dados das fontes, sua integração e preparação. Nesse processo, são eliminadas inconsistências da base, dados incompletos e registros irrelevantes para a análise.
- *Data Warehouse* – DW: criado a partir do processo de ETL, armazenam as informações consolidadas, já modeladas e preparadas para a exploração. O DW é composto por um ou mais subconjuntos lógicos, especializados em determinado assunto, de acordo com seu conteúdo. Esses subconjuntos são chamados de *Data Marts*.
- Área de apresentação: onde é feita a exploração dos dados. São usadas técnicas de análise de dados sobre as bases consolidadas para gerar informação de suporte à tomada de decisões (KIMBALL; ROSS, 2013).

2.3 Banco de dados

Banco de dados é um conjunto de dados integrados que tem por objetivo atender a uma comunidade de usuários. Para manter bancos de dados, são usados Sistemas de Gerência de Banco de Dados – SGBD. Os SGBDs surgiram no início da década de 70 com o objetivo de facilitar a programação de aplicações de banco de dados. Os primeiros sistemas eram caros e difíceis de usar, requerendo especialistas treinados para usar o SGBD específico. Nessa mesma época, houve um investimento considerável de pesquisa na área de banco de dados. Esse investimento resultou em um novo tipo de SGBD, o SGBD relacional. Um banco de dados relacional é composto de tabelas ou relações entre elas (HUESER, 2009).

A *Structured Query Language* – SQL é uma linguagem padrão para acesso e

manipulação de dados em um SGBD relacional. É uma linguagem semelhante ao inglês que consiste em diversas camadas de complexidade e capacidade crescentes. A linguagem SQL é usada para acesso a bancos de dados, operações SGBD, a partir de programas, e funções administrativas em bancos de dados. Também é usada em funções de acesso e manipulação de dados em alguns dos principais produtos de *software* SGBD. O SQL pode ser usado para programas escritos em qualquer linguagem padrão de programação, por isso, facilita a integração de *softwares*. Frequentemente, os sistemas de consulta são combinados com sistemas de relatório (TURBAN et al., 2009).

A partir da década de 80 e devido ao barateamento das plataformas de *hardware* e *software* para executar SGBD relacional, este tipo de SGBD passou a dominar o mercado, tendo se tornado padrão internacional. Além dos SGBD relacionais, existem outros tipos de sistemas no mercado (HUESER, 2009).

As vantagens de usar de usar um SGBD relacional são:

- Controle de redundância: No desenvolvimento de software tradicional utilizando processamento de arquivos, cada grupo de usuários mantém seus próprios arquivos para lidar com seus aplicativos de processamento de dados. Esta redundância no armazenamento dos mesmos dados múltiplas vezes leva a vários problemas. Como a necessidade de realizar uma única atualização lógica várias vezes, uma vez para cada arquivo onde os dados são registrados, isto leva à duplicação de esforços. O espaço de armazenamento é desperdiçado quando os mesmos dados são armazenados repetidamente. Os arquivos que representam os mesmos dados podem tornar-se inconsistentes. Isso pode acontecer porque uma atualização é aplicada a alguns arquivos, mas não a outros.
- Controle de acesso não autorizado: Quando vários usuários compartilham um grande banco de dados, é provável que a maioria dos usuários não estejam autorizados a acessar todas as informações do banco de dados. Por exemplo, dados financeiros como salários e bônus são frequentemente considerados confidenciais e apenas pessoas autorizadas têm permissão para acessar esses dados. Além disso, alguns usuários podem apenas ter permissão para consultar dados, enquanto outros têm permissão para consultar e atualizar. Normalmente, os usuários ou grupos de usuários recebem contas protegidas por senhas, que eles podem usar para obter acesso ao banco de dados. Um SGBD deve fornecer um subsistema de segurança e autorização.

- Fornecendo estruturas de armazenamento e técnicas para processamento eficiente de consultas: Os SGBD devem fornecer recursos para a execução eficiente de consultas e atualizações. Como o banco de dados normalmente é armazenado em disco, o SGBD deve fornecer estruturas de dados especializadas e técnicas de pesquisa para acelerar a pesquisa em disco para os registros desejados. Para processar os registros do banco de dados necessários por uma consulta específica, esses registros devem ser copiados do disco para a memória principal. Portanto, o SGBD geralmente possui um módulo de *buffer* ou *cache* que mantém as partes do banco de dados nos *buffers* da memória principal. Em geral, o sistema operacional é responsável pelo *buffer* de disco para memória. No entanto, como o *buffer* de dados é crucial para melhorar o desempenho do SGBD, a maioria dos SGBDs faz seu próprio *buffer* de dados. O módulo de processamento e otimização de consultas do SGBD é responsável por escolher um plano de execução de consulta eficiente para cada consulta com base no existente nas estruturas de armazenamento.
- Fornecendo *backup* e recuperação: Um SGBD deve fornecer recursos para recuperação de falhas de *hardware* ou *software*. O subsistema de *backup* e recuperação do SGBD é responsável pela recuperação. Se o sistema do computador falhar no meio de uma transação de atualização complexa, o subsistema de recuperação é responsável por garantir que o banco de dados seja restaurado ao estado em que estava antes da transação começar a ser executada. *Backup* de disco também é necessário no caso de uma falha catastrófica.
- Representando relacionamentos complexos entre dados: Um banco de dados pode incluir inúmeras variedades de dados que estão inter-relacionados em muitos aspectos. O SGBD deve ter a capacidade de representar uma variedade de relacionamentos complexos entre os dados, para definir novas relações à medida que surgem, e para recuperar e atualizar dados relacionados com facilidade e eficiência.
- Aplicando restrições: A maioria dos SGBDs têm certas restrições de integridade que devem ser mantidas para os dados. Um SGBD deve fornecer capacidades para definir e fazer cumprir essas restrições. O tipo mais simples de restrição de integridade envolve a especificação de um tipo de dado para cada item. Um tipo mais complexo de restrição que ocorre com frequência envolve a especificação de que um registro em um arquivo deve estar relacionado a registros em outros arquivos. Outro tipo de restrição específica é a exclusividade nos valores dos itens de dados.

- Permitindo ações usando regras e gatilhos: É possível associar gatilhos a tabelas. Um gatilho é uma forma de regra ativado por atualizações na tabela, o que resulta na execução de algumas operações adicionais para algumas outras tabelas, enviando mensagens e assim por diante eles se tornam parte da definição geral do banco de dados e são invocados apropriadamente quando certas condições são atendidas (ELMASRI; NAVATHE, 2016).

2.4 Metadados

Os dados, para produzirem informações e conhecimentos, deverão ter a sua definição completa e unânime. E a definição dos dados é a essência de metadados. Eles servem para você saber o que está consumindo em termos de dados, informações e conhecimento. Os metadados podem ser classificados como sintáticos, semânticos e estruturais. Os sintáticos definem as regras de formação dos nomes dos dados, garantindo padronizações. Os semânticos definem o sentido que se dá àquele elemento informacional, garantindo o seu pleno entendimento nos contextos organizacionais em que é produzido ou consumido. Os estruturais definem como o dado é composto em partes menores e detalha a sua formação estrutural. Os SGBDs usam um catálogo no qual armazena os metadados técnicos necessários ao seu processo de otimização de consultas. Ali residem tabelas contendo informações sobre tabelas relacionais, campos, índices, usuários, *triggers*, *stored procedures* etc. Para cada campo, há metadados como nome, tipo, tamanho, indexação, cardinalidade etc. Esses dados são fundamentais para o SGBD na definição e escolha dos melhores caminhos de acessos existentes dentro de um comando SQL. Nesse contexto de governança e qualidade de dados, um dos pontos que mais se destaca e se discute é a necessidade da devida formação dos metadados. Os metadados também terão papel importante no conceito de *web* semântica, à medida que, nesse contexto, os conteúdos serão cada vez mais acessados e processados por algoritmos inteligentes (BARBIERI, 2011).

2.5 Data Warehouse

Data Warehouse (DW) é um repositório de dados produzido para oferecer suporte à tomada de decisões e também um repositório de dados atuais e históricos. Os dados normalmente são estruturados de modo a estarem disponíveis em um formato pronto para as atividades de processamento analítico. Portanto é uma coleção de dados orientada por assunto, integrada, variável no tempo e não-volátil, que proporciona

suporte ao processo de tomada de decisões. Os tomadores de decisão necessitam de informações concisas e confiáveis sobre operações atuais, tendências e mudanças. Os dados são constantemente fragmentados com o uso de sistemas operacionais diferentes, e assim os gerentes tomam decisões com informações parciais. O *data warehouse* supera esse obstáculo acessando, integrando e organizando os principais dados operacionais de uma forma consistente, confiável, pontual e prontamente disponível onde for necessário.

Características de um DW:

- Orientado por assunto: Os dados são organizados por assunto pormenorizado e contêm apenas as informações relevantes ao suporte à decisão. A orientação por assunto permite que os usuários determinem não só como está o desempenho de sua organização, mas também permite determinar o motivo deste desempenho. Um *data warehouse* difere de um banco de dados operacional no sentido de que estes, em sua maioria, são orientados por produto e ajustados para lidar com transações que atualizam o banco de dados. A orientação por assunto de um DW proporciona uma visão mais abrangente da organização.
- Integrado: A integração está bastante ligada à orientação por assunto. Os DW devem colocar os dados de diferentes fontes em um formato consistente. Para isso, devem enfrentar conflitos de nomenclaturas e discrepâncias entre unidades de medida. Espera-se que um DW seja totalmente integrado.
- Variável no tempo: Um *data warehouse* mantém dados históricos. Os dados não necessariamente mostram o status atual (exceto em sistemas em tempo real). Eles detectam tendências, variações, relações de longo prazo para previsão e comparações. Há uma qualidade temporal para cada DW. O tempo é uma dimensão importante à qual todo DW deve oferecer suporte. Os dados de análise vindos de diversas fontes contêm diversos pontos de tempo (visualizações diárias, semanais, mensais, etc).
- Não-volátil: Após os dados serem inseridos em um DW, os usuários não podem alterar ou atualizá-los. Os dados obsoletos são descartados e as alterações são registradas como dados novos. Isso permite que o *data warehouse* seja ajustado quase exclusivamente para o acesso a dados.

Os *data warehouses* normalmente são criados para apresentar um ambiente informatizado eficiente às aplicações baseadas na *Web*, usa ou uma estrutura relacional ou uma estrutura multidimensional, usa a arquitetura cliente/servidor para proporcionar aos usuários acesso fácil, podem oferecer acesso a dados e recursos de análise em tempo real, ou ativos e podem conter metadados acerca de como os dados estão

organizados e como usá-los de forma eficiente (TURBAN et al., 2009).

2.6 Data marts

Um *data warehouse* une bancos de dados de toda uma empresa; já um *data mart* normalmente é menor e se concentra em um assunto ou departamento específico. É um subconjunto de um DW, que normalmente consiste em uma única área temática, podendo ser dependente ou independente. Um *data mart* dependente é um subconjunto criado a partir do DW. Ele tem a vantagem de usar um modelo de dados consistente e apresentar dados de qualidade. Os *data marts* dependentes suportam o conceito de um único modelo de dados em toda a empresa, porém o *data warehouse* deve ser construído antes, para garantir que o usuário final visualize a mesma versão de dados acessada por todos os outros usuários do DW, entretanto o seu alto custo limita seu uso às grandes empresas. Como alternativa, muitas empresas usam uma versão de *data warehouse* reduzida em custo e escala, denominada *data mart* independente, que é um *warehouse* pequeno, projetado para uma unidade estratégica de negócios ou um departamento (TURBAN et al., 2009).

2.7 Data Mining

Data mining é um processo para explorar dados para encontrar os padrões e relacionamentos que descrevem os dados e prever os valores desconhecidos ou futuros dos dados. O valor chave na mineração de dados é a capacidade de entender por que algumas coisas aconteceram no passado e de prever o que acontecerá no futuro. Quando a mineração de dados é usada para explicar a situação atual ou passada, é chamada de análise descritiva. Quando a mineração de dados é usada para prever o futuro, ela é chamada análise preditiva.

Em *business intelligence*, aplicações populares de mineração de dados são para detecção de fraudes (indústria de cartões de crédito), previsão e orçamento (finanças), desenvolvimento de pacotes celulares/móveis analisando padrões de chamadas (indústria de telecomunicações), análise de cesta de compras (varejo indústria), perfil de risco do cliente (indústria de seguros), monitoramento de uso (energia e serviços públicos), dentre outros (INMON, 2005).

2.8 Online Analytical Processing (OLAP)

O Processamento Analítico Online – OLAP é a atividade de analisar interativamente dados armazenados na dimensão *data warehouse* profissional para tomar decisões de negócios táticas e estratégicas. Funcionalidade típica em OLAP inclui agregação, detalhamento e fatiamento. A funcionalidade OLAP pode ser entregue usando um banco de dados relacional ou usando um banco de dados multidimensional. OLAP que usa um banco de dados relacional é conhecido como processamento analítico online relacional. OLAP que usa um banco de dados multidimensional é conhecido como processamento analítico online multidimensional (INMON, 2005).

Com o OLAP os usuários são capazes de identificar com rapidez e facilidade as tendências de desempenho. Essa identificação é feita com a análise de informações cíclicas e recursos de gráficos de produtos que suportem análises de dados mais sofisticadas e integrem recursos completos de campos calculados aos relatórios. Há muitas ferramentas de *software* que permitem aos usuários criarem relatórios e consultas sob demanda e realizarem análises de dados (TURBAN et al., 2009).

2.9 Extract, Transform and Load

O coração da parte técnica do processo de criação de *data warehouse* é o ETL. As tecnologias de ETL, que já existem há algum tempo, são providenciais para o processo e uso de *data warehouses*. O processo de ETL é um componente integral de qualquer projeto centrado em dados e costumam consumir 70% do tempo em um projeto.

O processo de ETL consiste em extração (leitura dos dados de um ou mais fonte de dados), *transformação* (conversão dos dados extraídos de sua forma anterior na forma em que precisam estar, para que sejam colocados em um DW ou apenas em outro banco de dados) e *carga* (inserção dos dados no DW). A transformação ocorre com o uso de regras ou tabelas de busca ou com a combinação dos dados com outros dados. As ferramentas de ETL também transportam dados entre fontes e alvos, documentam como os elementos de dados (metadados) mudam conforme se movimentam entre fonte e alvo, trocam metadados com outras aplicações conforme necessário, e administram todos os processos e operações de tempo de execução (como programação, gerenciamento de erros, registros de auditorias, estatísticas). O ETL é extremamente importante na integração de dados e também na criação de DW. O objetivo do processo de ETL é carregar dados integrados e limpos no *warehouse*. Os dados usados nestes processos podem ser oriundos de qualquer fonte: um arquivo texto, uma planilha do Excel ou até uma fila de mensagens (TURBAN et al., 2009).

Figura 2 – Processo de ETL.



Fonte: Adaptado de BDASOLUTIONS (2020).

O processo de migração de dados para um DW envolve a extração de dados de todas as fontes relevantes. As fontes de dados consistem em arquivos extraídos de bancos de dados, planilhas ou arquivos externos. Todos os arquivos de entrada são gravados em um conjunto de tabelas temporárias, criadas para facilitar o processo de carga. Um *data warehouse* contém inúmeras regras de negócios que definem questões como a forma de utilização dos dados, regras de resumo, padronização dos atributos codificados e regras de cálculo. Qualquer problema na qualidade dos dados pertencentes aos arquivos-fonte precisa ser corrigido antes que os dados sejam carregados no *data warehouse*. Um dos benefícios de um DW bem projetado é que essas regras podem ser armazenadas em um repositório de metadados e aplicadas de forma central (TURBAN et al., 2009).

2.10 Key Performance Indicators

Os *key performance indicators* comumente chamados de KPI, representam um conjunto de métricas focado em aspectos do desempenho organizacional que são fundamentais para o sucesso da organização. 7 características importantes para os KPIs são:

- São medidos frequentemente (por exemplo, 24 horas por dia, 7 dias por semana, diariamente ou semanalmente)
- A equipe de gerência são os responsáveis pela atuação.
- Indica claramente quais ações são exigidas pela equipe (por exemplo, equipe pode entender as medidas e saber o que corrigir)

- São métricas que vinculam a responsabilidade a uma equipe (por exemplo, O CEO pode chamar um líder de equipe que possa tomar as medidas necessárias)
- Tem um impacto significativo (por exemplo, afetar um ou mais dos os fatores críticos de sucesso)
- Incentivam ações apropriadas (por exemplo, foram testados para garantir que tenham um impacto positivo no desempenho, considerando que medidas mal pensadas podem levar a problemas disfuncionais comportamentais) (PARMENTER, 2010).

2.11 Python

Desde o seu surgimento em 1991, *Python* se tornou uma das linguagens de programação interpretadas mais populares, frequentemente chamado de linguagem de *scripting*, por poder ser usada para escrever rapidamente pequenos programas ou *scripts* para automatizar outras tarefas. *Python* desenvolveu uma comunidade grande e ativa de processamento científico e análise de dados. Nos últimos dez anos, passou a ser considerado uma linguagem de computação científica inovadora, importante em ciência de dados, aprendizado de máquina e desenvolvimento de *softwares* em geral, no ambiente acadêmico e no mercado. Seu suporte melhorado para bibliotecas o transformou em uma opção popular para tarefas de análise de dados, em conjunto com a robustez de *Python* para uma engenharia de *software* de propósito geral, é uma excelente opção como uma linguagem principal para a construção de aplicações de dados (MCKINNEY, 2018).

2.12 BigQuery

O *BigQuery* é um *data warehouse* versátil e escalável dentro do ecossistema de nuvem. Em sua essência, o *BigQuery* foi projetado para abordar as complexidades da análise de dados, fornecendo uma solução de *data warehouse* totalmente gerenciada e sem servidor. Essa natureza sem servidor elimina a necessidade das organizações gerenciarem a infraestrutura, permitindo que elas se concentrem apenas em consultar e analisar seus dados. Oferece suporte a uma variedade de formatos de dados, incluindo dados estruturados e semiestruturados, permitindo que as organizações consumam diversas fontes de dados. O princípio fundamental de sua arquitetura é a escalabilidade, permitindo que organizações manipulem com facilidade conjuntos de dados de tamanhos variados. A plataforma emprega uma arquitetura de processamento massivo paralelo, distribuindo consultas em vários nós para execução paralela. Esse paralelismo garante a execução de consultas de alto desempenho (MIRYALA; GUPTA, 2023).

2.13 Looker Studio

Lançado em 2016 o *Looker Studio* é um programa de visualização de dados projetado como uma ferramenta amigável para representar conjuntos de dados complexos de forma clara, com o propósito de ajudar os usuários a “criar relatórios e *dashboards* dinâmicos e visualmente atraentes” canalizando fontes de dados externas em uma plataforma de fácil navegação, facilitando a criação e o compartilhamento de relatórios baseados em dados, com a capacidade de integrar várias fontes em um único relatório e criar novas métricas para análise dentro da plataforma em vez de exigir modificação dos dados originais. Um dos recursos mais impressionantes da ferramenta permite que os visualizadores interajam com as descobertas em tempo real, permitindo que ajustem os intervalos de datas dos dados exibidos, filtrem relatórios para variáveis específicas e executem outras funções que melhoram a relevância para o visualizador. Outro ponto forte são seus recursos de compartilhamento e colaboração. Os criadores podem compartilhar seus relatórios publicamente ou privadamente e definir permissões para permitir que os colaboradores visualizem e editem seu trabalho. Os relatórios do *Looker* existem independentemente de seus dados subjacentes, portanto, quando um proprietário de relatório concede acesso de edição ou visualização a um terceiro, esse indivíduo não seria capaz de visualizar ou modificar os dados originais sem receber acesso separado (SNIPES, 2018).

3 MATERIAL E MÉTODOS

3.1 Coleta

Foram coletados dados referentes à venda e estoque de duas empresas de comércio de pescado localizadas em Fortaleza, Ceará. Na primeira empresa “M.A.C. Fish”, os dados foram coletados por meio de planilhas impressas confeccionadas pela autora. A segunda empresa “Francisco Camocim: A Casa do Peixe” forneceu os dados de seu próprio sistema de controle.

3.2 Criação do *Data Warehouse*

Após a coleta dos dados, iniciou-se o desenvolvimento da solução de *Business Intelligence*, com o projeto de construção do *Data Warehouse* – DW, criado no serviço

de armazenamento *BigQuery*. Inicialmente foi realizado o processo de ETL (*Extract, Transform, Load*) por meio de um *script* feito na linguagem de programação *Python*, onde os dados das duas empresas foram tratados de forma a eliminar inconsistências da base, tais como: tratar valores nulos, completar dados e eliminar registros irrelevantes para análise, com o objetivo de melhorar a qualidade dos dados, então os dados foram estruturados para formarem uma única tabela de vendas e outra de estoque. Os dados foram organizados e modelados de acordo com as necessidades do projeto.

Após a modelagem, os dados foram armazenados no *BigQuery*. Com o DW finalizado pode-se explorar os dados, consultas em linguagem SQL foram desenvolvidas para trazer os dados necessários para construir os *dashboards* na ferramenta de visualização de dados *Looker Studio*.

4 RESULTADOS

4.1 KPIs

A seleção de KPIs é um processo inicial para o gerenciamento eficaz de uma organização, refletindo o seu desempenho em relação aos seus objetivos estratégicos. A escolha adequada desses indicadores permite uma avaliação precisa do progresso e facilita a tomada de decisões. Os KPIs devem estar diretamente ligados aos objetivos da empresa. Isso garante que as métricas reflitam o desempenho em áreas cruciais para o sucesso do negócio. A organização pode selecionar um conjunto de KPIs que não apenas monitorem o desempenho atual, mas que também impulsionam melhorias contínuas e o alcance dos objetivos estratégicos. Os KPIs escolhidos para esse projeto foram:

Tabela 1 – KPIs da loja Camocim.

<i>Dashboard</i>	KPI	Significado
Venda	Faturamento	Indica o valor total de produtos vendidos no período.
	Maior valor de venda	Indica a venda de maior valor no período.
	Volume de vendas	Número total de vendas no período.
	Ticket médio	Valor médio gasto por cliente no período.
Estoque	Quantidade de fornecedores	Número de fornecedores no período.
	Quantidade de produtos que entraram em estoque	Número de produtos que entraram no estoque no período.
	Quantidade de produtos que saíram do estoque	Número de produtos que saíram no estoque no período.
	Peso dos produtos que entraram em estoque	Peso em Kg dos produtos que entraram no estoque no período.
	Peso dos produtos que saíram do estoque	Peso em Kg dos produtos que saíram no estoque no período.

Fonte: elaborado pela autora (2024).

Tabela 2 – KPIs da loja M.A.C.

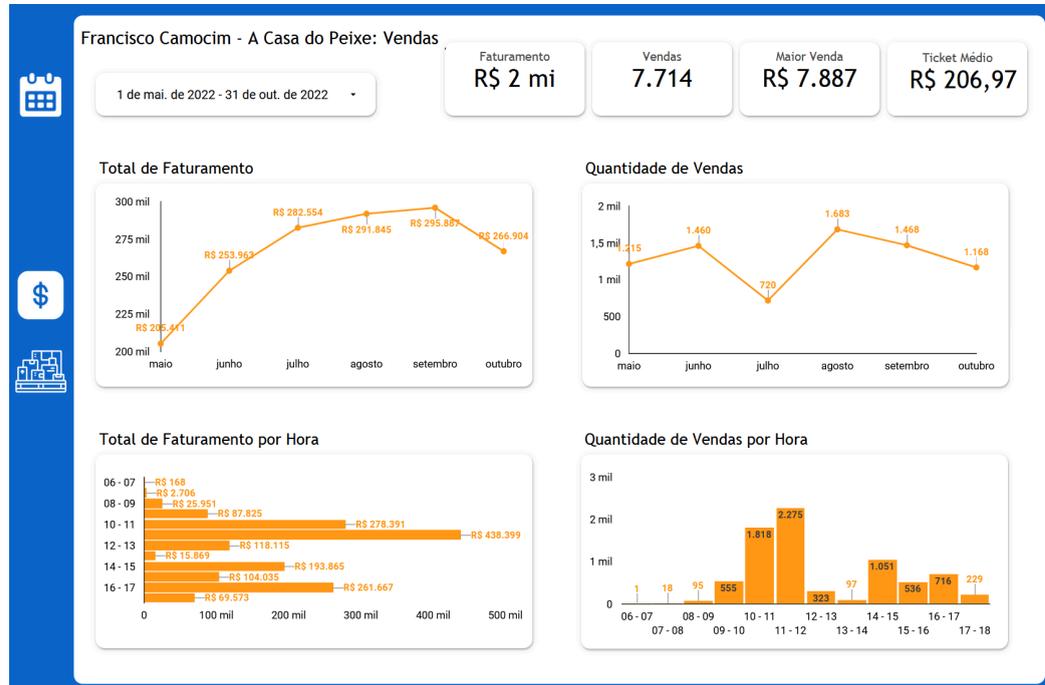
Dashboard	KPI	Significado
Venda	Faturamento	Indica o valor total de produtos vendidos no período.
	Volume de vendas	Número total de vendas no período.
	Ticket médio	Valor médio gasto por cliente no período.
Estoque	Quantidade de fornecedores	Número de fornecedores no período.
	Quantidade de produtos que entraram em estoque	Número de produtos que entraram no estoque no período.
	Valor de compra	Custo do produto
	Peso dos produtos que entraram em estoque	Peso em Kg dos produtos que entraram no estoque no período.

Fonte: elaborado pela autora (2024).

4.2 Dashboard

Um *dashboard* é uma ferramenta que auxilia na organização de informações, no monitoramento de métricas e KPIs, e na análise dos dados do negócio de forma visual. Esse recurso pode ser utilizado de forma abrangente em toda a empresa, em setores específicos ou até mesmo em campanhas e ações pontuais. A solução adotada no *Looker Studio* é visto como o resultado mais concreto deste projeto:

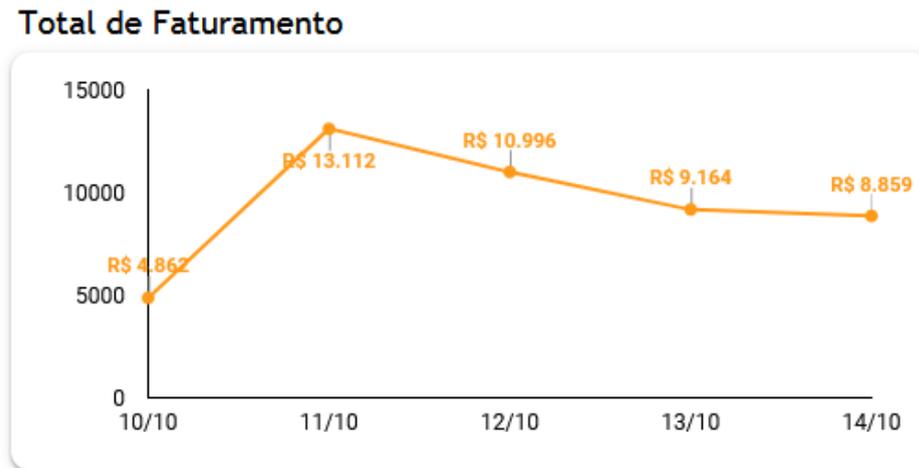
Figura 3 – *Dashboard* de vendas da loja Camocim no Looker Studio.



Fonte: elaborado pela autora (2024).

Os *dashboards* são interativos e possibilitam que métricas sejam mostradas ou ocultadas com um clique:

Figura 4 – Visão diária do gráfico de faturamento da loja Camocim no Looker Studio.



Fonte: elaborado pela autora (2024).

Os números indicam um desempenho robusto em termos de receita e volume de vendas. O faturamento da loja apresentou um crescimento contínuo, atingindo seu pico em setembro. No entanto, nos meses subsequentes, observa-se uma tendência de queda. Além disso, o horário das 11h às 12h representa o pico de vendas, registrando o maior volume de transações e faturamento em todo o período analisado, essa informação é importante para a criação de uma estratégia de otimização de operações e atendimento. Em média, cada cliente gasta R\$206,97 o que sugere que os produtos vendidos possuem um valor considerável, indicando uma estratégia bem-sucedida de venda de produtos *premium* ou uma combinação de produtos de maior valor. Destaca-se ainda que a maior venda realizada por um único cliente ocorreu em 23 de setembro, totalizando R\$7.887,00. Analisar os fatores que contribuíram para essa transação, como o tipo de produto, promoções ou eventos específicos, pode fornecer *insights* valiosos e insumos para a criação de novas estratégias de vendas.

Os gráficos de Faturamento e Vendas mostram variações ao longo dos dias, com alguns picos, indicando dias de maior demanda. Esses gráficos podem ser utilizados para identificar os períodos de maior e menor demanda, permitindo ajustes estratégicos em estoques e funcionários.

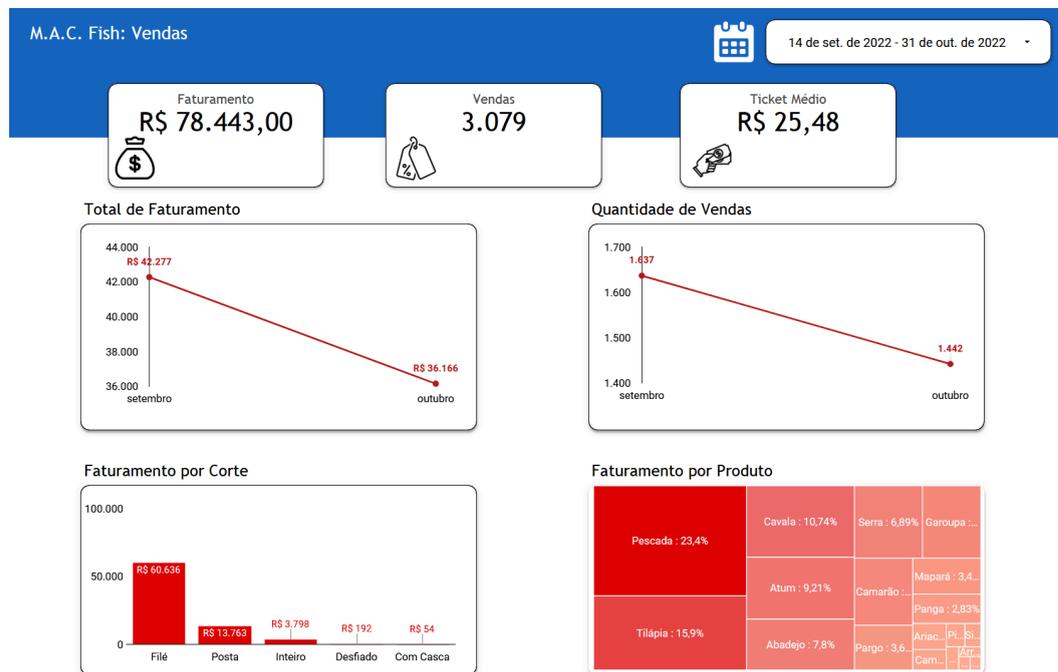
Figura 5 – *Dashboard* de estoque da loja Camocim no Looker Studio.

Fonte: elaborado pela autora (2024).

A variedade de produtos, tanto na entrada quanto na saída, mostra uma oferta diversificada, o que pode atrair uma vasta clientela, os pescados Beijupirá, Dentão e Acará Açú foram os mais movimentados. A preferência por produtos inteiros nas saídas sugere que esse é o formato mais popular entre os clientes, possibilitando futuras estratégias de corte e preparação de produtos. Os gráficos mostram flutuações nas entradas e saídas de produtos ao longo do período, com alguns picos em determinados dias, refletindo a dinâmica de compras, vendas e consumo de estoque. O total de 33 fornecedores indica uma rede diversificada de suprimentos, o que pode ajudar na mitigação de riscos relacionados à dependência de um único fornecedor.

No período analisado a quantidade de produtos que sai é significativamente maior do que a quantidade que entra, indicando uma alta rotatividade de estoque. Isso, somado à análise do *dashboard* de vendas, corrobora que a loja possui um alto fluxo de vendas, mas também pode sugerir a necessidade de monitoramento rigoroso para evitar rupturas de estoque.

Figura 6 – Dashboard de vendas da loja M.A.C. no Looker Studio.

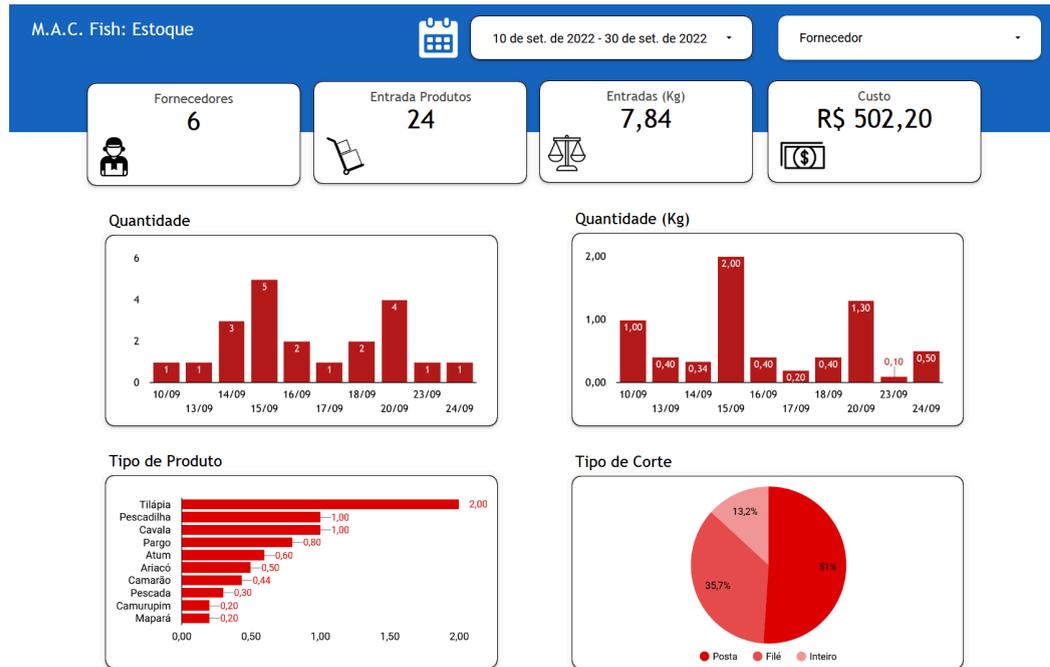


Fonte: elaborado pela autora (2024).

Durante o período analisado, foram realizadas 3.079 vendas e a empresa gerou R\$78.443,00 em receita, o filé se destacou como o tipo de corte com o maior faturamento. Além disso, Tilápia, Pescada e Cavala foram responsáveis por 50% do faturamento total da loja. Em média, cada cliente gastou um total de R\$25,48 por transação.

As vendas mostram flutuações significativas ao longo do período, apesar da forte dependência de alguns produtos a empresa também possui uma diversidade de oferta que pode ser explorada para melhorar o desempenho geral. Identificar os fatores que influenciam esses picos de vendas, como promoções, sazonalidade ou eventos, pode ajudar a aumentar o faturamento no futuro, os dados podem ser utilizados para planejar promoções durante os períodos de menor venda e explorar ao máximo os períodos de alta demanda.

Figura 7 – Dashboard de estoque da loja M.A.C. no Looker Studio.



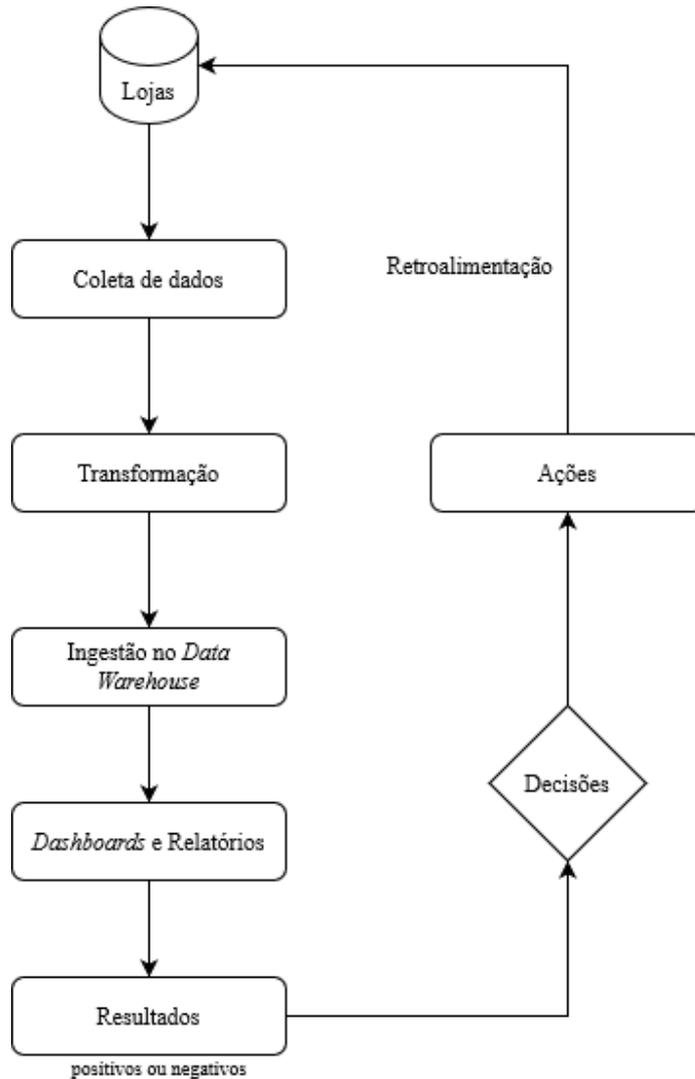
Fonte: elaborado pela autora (2024).

Durante o período analisado, a posta representou a maior parte dos tipos de cortes que entraram no estoque, com a tilápia liderando como o tipo de pescado, porém a dependência de alguns produtos como Tilápia e Pescada, embora positiva em termos de demanda como mostrado no painel de vendas, pode indicar a necessidade de estratégias para equilibrar a oferta e evitar riscos associados à dependência excessiva de poucos itens, também se faz necessário monitorar as preferências dos clientes em relação aos tipos de corte e ajustar a oferta conforme necessário para maximizar a satisfação do cliente e considerar a busca de novos parceiros para aumentar a resiliência da cadeia de suprimentos.

4.3 Gestão dos dados

A manutenção de um projeto de dados para uma micro ou pequena empresa do setor de pescados requer a implementação contínua de várias etapas essenciais. Portanto, foi elaborado um modelo padrão para auxiliar na construção e manutenção de um projeto.

Figura 8 – Modelo conceitual para a gestão de dados de micro e pequenas empresas.



Fonte: elaborado pela autora (2024).

Os dados das lojas são coletados e, em seguida, passam por um processo de transformação, onde são limpos e tratados para o formato necessário. Este processo pode ser realizado por meio de *scripts* em *Python* ou outras ferramentas de ETL. Após o tratamento, os dados são inseridos e armazenados em um *Data Warehouse*, como o *BigQuery* ou outro banco de dados similar. Esses dados armazenados no DW são apresentados em forma de relatórios e *dashboards*, construídos com ferramentas de visualização como o *Looker Studio*. A partir da análise dos resultados, sejam eles positivos ou negativos, são tomadas decisões que geram ações impactando as lojas. Essas ações resultam em novos dados, alimentando continuamente o ciclo de coleta e análise de informações.

5 CONCLUSÃO

Neste projeto, foram analisados os dados de venda e estoque de duas lojas especializadas na venda de pescados, com o objetivo de compreender melhor o comportamento de seus clientes, otimizar suas operações e identificar oportunidades de crescimento.

É possível considerar que a metodologia adotada obteve sucesso, permitindo a criação de gráficos e *dashboards* que apresentam diversas informações relevantes sobre vendas e estoque. Os *insights* obtidos ajudarão as lojas a melhorar suas operações atuais e servirão como base para a tomada de decisões estratégicas futuras. A implementação das recomendações propostas podem resultar em aumento de receita, redução de custos operacionais e uma experiência de compra mais satisfatória para os clientes. Este trabalho também destaca a importância de uma cultura orientada a dados dentro da organização, onde decisões são informadas por *insights* precisos e oportunos, garantindo crescimento sustentável e contínuo.

Por fim, recomenda-se que as lojas implementem um modelo de gestão de dados e continuem investindo em tecnologias de análise de dados e na capacitação de sua equipe, assegurando que as práticas de análise de dados evoluem juntamente com o mercado e as necessidades dos consumidores.

REFERÊNCIAS

- AMARAL, L. **Um referencial para o planejamento de sistemas de informação**. 1994. Tese (Doutorado em Sistema de Informática) – Programa de Pós-Graduação em Sistema de Informática, Universidade do Minho, Braga, 1994.
- BATISTA, E.O. **Sistemas de Informação**. São Paulo: Saraiva, 2004.
- BARBIERI, C. **BI2: Business Intelligence: Modelagem e Qualidade**. Rio de Janeiro: Elsevier, 2011.
- BRASIL. Ministério do Desenvolvimento, Indústria, Comércio e Serviços. **Brasil tem 2,7 milhões de novas empresas em 2023**. Brasília, 2023. Disponível em: <https://www.gov.br/mdic/pt-br/assuntos/noticias/2023/setembro/brasil-tem-2-7-milhoes-de-novas-empresas-em-2023>. Acesso em: 10 mar. 2024.
- CUNHA, F.B.D. **A importância do planejamento estratégico para gestão de pequenas empresas**. Revista Diálogos Interdisciplinares, vol. 3, nº.3, (Edição Especial), 2014.
- ELMASRI, S.; NAVATHE, S.B. **Fundamentals of Database Systems**. 7. ed. [S.l.]: Pearson Education, 2016.
- FAO. **The State of World Fisheries and Aquaculture 2022: Towards Blue Transformation**. Roma: FAO, 2022. Disponível em: <https://www.fao.org/documents/card/en/c/cc0461en>. Acesso em: 13 jan. 2024.
- HEUSER, C. **Projeto de banco de dados : Volume 4 da Série Livros didáticos informática UFRGS**. [S.l.]: Bookman, 2009. (Livros didáticos informática UFRGS).
- INMON, W. H. **Building the Data Warehouse**. 4th ed. ed. [S.l.]: Wiley, 2005.
- KIMBALL, R.; ROSS, M. **The Data Warehouse Toolkit: The Definitive Guide to Dimensional Modeling**. 3. ed. [S.l.]: Wiley, 2013.
- MCGEE, J. V.; PRUSAK, L. **Gerenciamento estratégico da informação**. Rio de Janeiro: Elsevier Brasil, 1994.
- MCKINNEY, W. **Python para análise de dados**. 2. ed. [S.l.]: Novatec, 2018.
- MIRYALA, N. K.; GUPTA, D. **Big Data Analytics in Cloud - Comparative Study**. *International Journal of Computer Trends and Technology*, v. 71, p. 35-43, 2023.
- MOLINA, L. G.; VALENTIM, M. L. P., **Memória Organizacional como Forma de Preservação do Conhecimento**. Perspectivas em Gestão & Conhecimento, João Pessoa, 2015.
- MUSSKOPF, G.W. **Análise das ferramentas de business intelligence utilizadas por empresas brasileiras**. 2017.

PACE, E. S. U.; BASSO, L. F. C; SILVA, M. A. da. **Indicadores de Desempenho como Direcionadores de Valor**. RAC, v. 7, n. 1, Jan./Mar. 2003: 37-65.

PARMENTER, D. **Key performance indicators: developing, implementing, and using winning KPIs**. Second edition. [S.l.]: John Wiley & Sons, 2010.

REZENDE, Y. **Informação para negócios: os novos agentes do conhecimento e a gestão do capital intelectual**. Ci. Inf. Brasília, v.31, n. 2, p. 75-128, 2002.

RODRIGUES, W.D.P.; RIELLA, R. J. **Arquitetura de um ambiente colaborativo de business intelligence para micro, pequenas e médias empresas**.

SEBRAE. **Brasil alcança recorde de novos negócios, com quase 4 milhões de MPE, 2022; Micro e pequenas empresas geram 27% do PIB do Brasil, 2023**. Brasília: Sebrae. Disponível em: <https://www.sebrae.com.br/sites/PortalSebrae/ufs/ma/noticias/brasil-alcanca-recorde-de-novos-negocios-com-quase-4-milhoes-de-mpe,b7e02a013f80f710VgnVCM100000d701210aRCRD> e <https://sebrae.com.br/sites/PortalSebrae/ufs/mt/noticias/micro-e-pequenas-empresas-geram-27-do-pib-do-brasil,ad0fc70646467410VgnVCM2000003c74010aRCRD>. Acesso em: 19 mai. 2024.

SNIPES, G. **Google Data Studio**. *Journal of Librarianship and Scholarly Communication*, v. 6, n. 1, eP2214, 2018. DOI: <https://doi.org/10.7710/2162-3309.2214>.

TURBAN, E. et al. **Business Intelligence: Um enfoque gerencial para a inteligência do negócio**. [S.l.]: Bookman, 2009.

APÊNDICE A – PLANILHA BASE PARA COLETA DE DADOS

Tabela 3 – Planilha padrão utilizada para coletar os dados de venda.

PLANILHA DE VENDAS					
ID CLIENTE	DATA	QUANTIDADE	PESCADO	VALOR	OBSERVAÇÃO

Fonte: elaborado pela autora (2023).

Tabela 4 – Planilha padrão utilizada para coletar os dados de estoque.

PLANILHA DE ESTOQUE					
DATA	QUANTIDADE	PESCADO	VALOR DA COMPRA	FORNECEDOR	OBSERVAÇÃO

Fonte: elaborado pela autora (2023).

APÊNDICE B – *SCRIPT PYTHON*

```

#importando biblioteca
import pandas as pd
import numpy as np
#criando a tabela de vendas vazia
nomes_col_v
=['dt_venda','hora_venda','qtd_clientes','produto','qtd_produtos_unid','valor_vendas','m
aior_venda','tipo_corte','loja']
vendas_dw_mac = pd.DataFrame(columns=nomes_col_v)
vendas_dw_cam = pd.DataFrame(columns=nomes_col_v)
#criando path
path = 'C:\\Users\\LorenaSantiago\\Documents\\Lorena\\Profissional\\Engenharia de
Pesca\\'
#carregando dados
mac_vendas=pd.read_excel(path+'TCC\\Dados\\Planilhas_MAC.xlsx',
sheet_name='VENDAS')
display(mac_vendas)
#verificando e tratando valores
mac_vendas.isna().any()
mac_vendas['PESCADO'] = mac_vendas['PESCADO'].str.title()
mac_vendas['OBSERVAÇÃO'] = mac_vendas['OBSERVAÇÃO'].str.title()
mac_obs_uniq = mac_vendas['OBSERVAÇÃO'].drop_duplicates()
print(mac_obs_uniq)
mac_prod_uniq = mac_vendas['PESCADO'].drop_duplicates()
print(mac_prod_uniq)
mac_vendas.replace('Inteiro (F)','Inteiro', inplace=True)
mac_vendas.replace('Fresco','Inteiro', inplace=True)
mac_vendas.replace('-', 'Inteiro', inplace=True)
mac_vendas.replace('C/Casca','Com Casca', inplace=True)
mac_vendas.replace('Sirigado (M)','Sirigado', inplace=True)
mac_vendas.replace('Camarão (G)','Camarão', inplace=True)
mac_vendas.replace('Camarão (M)','Camarão', inplace=True)
mac_vendas.replace('Camarão (P)','Camarão', inplace=True)

```

```

mac_vendas['VALOR'] = pd.to_numeric(mac_vendas['VALOR'])
mac_vendas['QUANTIDADE'] = pd.to_numeric(mac_vendas['QUANTIDADE'])
mac_vendas['VALOR'] = mac_vendas['VALOR'] * mac_vendas['QUANTIDADE']
display(mac_vendas)
#populando a tabela
vendas_dw_mac['dt_venda'] = mac_vendas['DATA']
vendas_dw_mac['valor_vendas'] = mac_vendas['VALOR']
vendas_dw_mac['qtd_produtos_unid'] = mac_vendas['QUANTIDADE']
vendas_dw_mac['produto'] = mac_vendas['PESCADO']
vendas_dw_mac['tipo_corte'] = mac_vendas['OBSERVAÇÃO']
vendas_dw_mac['loja'] = 'Mac'
display(vendas_dw_mac)
#carregando dados
cam_vendas=pd.read_excel(path+'TCC\\Dados\\Planilhas_Camocim.xlsx',
sheet_name='VENDAS')
display(cam_vendas)
#verificando e tratando valores
cam_vendas.isna().any()
cam_vendas['DT_VENDA']=pd.to_datetime(cam_vendas['DT_VENDA'],
format='%d/%m/%Y')
display(cam_vendas)
#populando a tabela
vendas_dw_cam['dt_venda']=cam_vendas['DT_VENDA']
vendas_dw_cam['hora_venda']=cam_vendas['HORA']
vendas_dw_cam['valor_vendas']=cam_vendas['VALOR_VENDAS']
vendas_dw_cam['maior_venda']=cam_vendas['MAIOR_VENDA']
vendas_dw_cam['qtd_clientes']=cam_vendas['QUANTIDADE_CLIENTES']
vendas_dw_cam['loja']='Camocim'
display(vendas_dw_cam)
#concatenando tabelas
vendas_dw = pd.concat([vendas_dw_mac,vendas_dw_cam])
display(vendas_dw)
#output
vendas_dw.to_csv(path+'TCC\\Dados\\Tratados\\vendas_dw.csv', index=False)

```

```

#criando a tabela de estoque vazia
nomes_col_e=['dt_estoque','id_fornecedor','fornecedor','id_produto','produto','valor_co
mpra','qtd_estoque_kg','qtd_saida_kg','tipo_corte','loja']
estoque_dw_mac = pd.DataFrame(columns=nomes_col_e)
estoque_dw_cam = pd.DataFrame(columns=nomes_col_e)
#carregando dados
mac_estoque=pd.read_excel(path+'TCC\\Dados\\Planilhas_MAC.xlsx',
sheet_name='ESTOQUE')
display(mac_estoque)
#verificando e tratando valores
mac_estoque.isna().any()
mac_estoque.replace('P/POSTA','POSTA', inplace=True)
mac_estoque.replace('CORTE','POSTA', inplace=True)
mac_estoque.replace('FILÉ (CONGELADO)','FILÉ', inplace=True)
mac_estoque.replace('INTEIRA','INTEIRO', inplace=True)
mac_estoque.replace('FRESCO','INTEIRO', inplace=True)
mac_estoque['OBSERVAÇÃO'] = mac_estoque['OBSERVAÇÃO'].str.title()
mac_estoque['PESCADO'] = mac_estoque['PESCADO'].str.title()
mac_estoque['QUANTIDADE']=
pd.to_numeric(mac_estoque['QUANTIDADE']/1000
mac_estoque_duplicado = mac_estoque.iloc[[-1]]
mac_estoque.replace('Ariacó / Pargo','Ariacó', inplace=True)
mac_estoque = pd.concat([mac_estoque,mac_estoque_duplicado])
mac_estoque.replace('Ariacó / Pargo','Pargo', inplace=True)
display(mac_estoque)
#populando a tabela
estoque_dw_mac['id_fornecedor'] = mac_estoque['ID FORNECEDOR']
estoque_dw_mac['produto'] = mac_estoque['PESCADO']
estoque_dw_mac['dt_estoque'] = mac_estoque['DATA']
estoque_dw_mac['valor_compra'] = mac_estoque['VALOR DA COMPRA']
estoque_dw_mac['tipo_corte'] = mac_estoque['OBSERVAÇÃO']
estoque_dw_mac['qtd_estoque_kg'] = mac_estoque['QUANTIDADE']
estoque_dw_mac['loja'] = 'Mac'
display(estoque_dw_mac)

```

```

#carregando dados
cam_estoque=pd.read_excel(path+'TCC\\Dados\\Planilhas_Camocim.xlsx',
sheet_name='ESTOQUE')
display(cam_estoque)
#verificando e tratando valores
cam_estoque.isna().any()
cam_estoque['FORNECEDOR'].fillna('FORNECEDORDESCONHECIDO',
inplace=True)
cam_estoque['ID_FORNECEDOR'].fillna(0, inplace=True)
cam_estoque['DT_ESTOQUE']=pd.to_datetime(cam_estoque['DT_ESTOQUE'],
format='%d/%m/%Y')
cam_estoque['FORNECEDOR'] = cam_estoque['FORNECEDOR'].str.title()
cam_estoque['PRODUTO'] = cam_estoque['PRODUTO'].str.title()
unique_produto = list(cam_estoque['PRODUTO'].drop_duplicates())
print(unique_produto)
cam_estoque['TIPO_CORTE']=
np.where(cam_estoque['PRODUTO'].str.contains('File'),'Filé','Inteiro')
cam_estoque['TIPO_CORTE']=
np.where(cam_estoque['PRODUTO'].str.contains('CarneDe
Caranguejo'),'Filé',cam_estoque['TIPO_CORTE'])
cam_estoque['TIPO_CORTE']=
np.where(cam_estoque['PRODUTO'].str.contains('Bacalhau DesfiadoCong Dessalgado
C'),'Filé',cam_estoque['TIPO_CORTE'])
cam_estoque['TIPO_CORTE']=
np.where(cam_estoque['PRODUTO'].str.contains('Arraia
Desf'),'Filé',cam_estoque['TIPO_CORTE'])
cam_estoque['TIPO_CORTE']=
np.where(cam_estoque['PRODUTO'].str.contains('Posta'),'Posta',cam_estoque['TIPO_
CORTE'])
cam_estoque['TIPO_CORTE']=
np.where(cam_estoque['PRODUTO'].str.contains('Bacalhau Saithe Lombo Riberalves
1'),'Posta',cam_estoque['TIPO_CORTE'])
cam_estoque.replace('Peixe Tilapia','Tilápia', inplace=True)
cam_estoque.replace('Camarao C/Cab','Camarão', inplace=True)

```

cam_estoque.replace('File De Sirigado M','Sirigado', inplace=True)
cam_estoque.replace('File De Sirigado G','Sirigado', inplace=True)
cam_estoque.replace('File De Camarão','Camarão', inplace=True)
cam_estoque.replace('Peixe Atum','Atum', inplace=True)
cam_estoque.replace('File De Atum','Atum', inplace=True)
cam_estoque.replace('File De Pescada Amarela','Pescada Amarela', inplace=True)
cam_estoque.replace('File De Cioba','Cioba', inplace=True)
cam_estoque.replace('File De Bejupira G','Bejupira', inplace=True)
cam_estoque.replace('File De Arabaiana','Arabaiana', inplace=True)
cam_estoque.replace('Posta De Caçao','Cação', inplace=True)
cam_estoque.replace('Posta De Pescada B','Pescada Branca',inplace=True)
cam_estoque.replace('File De Vermelho M/G','Vermelho', inplace=True)
cam_estoque.replace('Posta De Pescada Am','Pescada Amarela', inplace=True)
cam_estoque.replace('File De Galo','Galo', inplace=True)
cam_estoque.replace('Camarao','Camarão', inplace=True)
cam_estoque.replace('File De Dourado','Dourado', inplace=True)
cam_estoque.replace('File De Cavala','Cavala', inplace=True)
cam_estoque.replace('Posta Cavala','Cavala', inplace=True)
cam_estoque.replace('File De Surubin','Surubin', inplace=True)
cam_estoque.replace('Tucunare','Tucunaré', inplace=True)
cam_estoque.replace('Sardinha Int','Sardinha', inplace=True)
cam_estoque.replace('Lula Aneis Cong Gr10 Kg','Lula', inplace=True)
cam_estoque.replace('Sardinha Lage Espalmada Pct 1 Kg Co','Sardinha',
inplace=True)
cam_estoque.replace('Lula Tubo Intefolhada','Lula', inplace=True)
cam_estoque.replace('Palombeta Evisc Cong','Palombeta', inplace=True)
cam_estoque.replace('Cavalinha Imp 300-500 Int Cong','Cavalinha', inplace=True)
cam_estoque.replace('File De Buchexa De Peixe','Buchexa de Peixe', inplace=True)
cam_estoque.replace('File De Abadejo','Abadejo', inplace=True)
cam_estoque.replace('File De Pescadilha','Pescadilha', inplace=True)
cam_estoque.replace('File De Garopa','Garopa', inplace=True)
cam_estoque.replace('File De Pescada B','Pescada', inplace=True)
cam_estoque.replace('File De Serra','Serra', inplace=True)
cam_estoque.replace('File De Cara','Tilápia', inplace=True)

cam_estoque.replace('Cavala Kg','Cavala', inplace=True)
cam_estoque.replace('Pargo Cong','Pargo', inplace=True)
cam_estoque.replace('Arraia Ped','Arraia', inplace=True)
cam_estoque.replace('Bejupira','Beijupirá', inplace=True)
cam_estoque.replace('Arabaina M','Arabaiana', inplace=True)
cam_estoque.replace('Xareu','Xaréu', inplace=True)
cam_estoque.replace('File De Salmao Chile','Salmão', inplace=True)
cam_estoque.replace('File Truta 500','Truta', inplace=True)
cam_estoque.replace('Bolinho De Bacalhau Ribeira Alves','Bolinho de Bacalhau',
inplace=True)
cam_estoque.replace('Bacalhau Saithe Lombo Riberlves 1','Bacalhau', inplace=True)
cam_estoque.replace('Polaca Do Alasca Salg Resf Lascas 1','Polaca do Alasca',
inplace=True)
cam_estoque.replace('Polaca Do Alasca Dess Tipo Bacalhau','Polaca do Alasca',
inplace=True)
cam_estoque.replace('Bolinho De Bacalhau Cong 20X270G','Bolinho de Bacalhau',
inplace=True)
cam_estoque.replace('Bolinha Bacalhau (Ban C/370G)','Bolinho de Bacalhau',
inplace=True)
cam_estoque.replace('Bacalhau Desfiado Cong Dessalgado C','Bacalhau',
inplace=True)
cam_estoque.replace('Patinha De Caranguejo Emp C/12 Un','Patinha de Caranguejo',
inplace=True)
cam_estoque.replace('Bolinha De Peixe (Bandeja C/370G)','Bolinha de Peixe',
inplace=True)
cam_estoque.replace('Bolinha C/Do Sol (Band C/370G)','Bolinha Carne de Sol',
inplace=True)
cam_estoque.replace('Bolinha Camarão (Band C/370G)','Bolinha Camarão',
inplace=True)
cam_estoque.replace('File De Camarão Pct 800Kg','Camarão', inplace=True)
cam_estoque.replace('Arraia Desf','Arraia', inplace=True)
cam_estoque.replace('Manjubimha','Manjubinha', inplace=True)
cam_estoque.replace('Tilapia','Tilápia', inplace=True)
cam_estoque.replace('Cavala Fresca','Cavala', inplace=True)

cam_estoque.replace('File De Camarao','Camarão', inplace=True)
cam_estoque.replace('File De Camarão 800G','Camarão', inplace=True)
cam_estoque.replace('Peixe Cavala','Cavala', inplace=True)
cam_estoque.replace('Kani Pct 200G','Kani', inplace=True)
cam_estoque.replace('Paella Ingredientes Cong 800 G','Paella Ingredientes',
inplace=True)
cam_estoque.replace('Gelo Em Escama','Gelo em Escama', inplace=True)
cam_estoque.replace('Ariaco','Ariacó', inplace=True)
cam_estoque.replace('Peixe Timbiro','Tibiro', inplace=True)
cam_estoque.replace('Peixe Inteiro Fresco Corvina','Corvina', inplace=True)
cam_estoque.replace('Peixe Inteiro Fresco Beijupira','Beijupirá', inplace=True)
cam_estoque.replace('Peixe Inteiro Fresco Cara Acu','Acará Açu', inplace=True)
cam_estoque.replace('Peixe Inteiro Fresco Serra','Serra', inplace=True)
cam_estoque.replace('Peixe Inteiro Fresco Cururuca','Corvina', inplace=True)
cam_estoque.replace('Cara Tilapia','Tilápia', inplace=True)
cam_estoque.replace('File De Tilapia','Tilápia', inplace=True)
cam_estoque.replace('Posta De Serra','Serra', inplace=True)
cam_estoque.replace('Peixe Fresco Xareu','Xaréu', inplace=True)
cam_estoque.replace('Peixe Fresco Dentao','Dentão', inplace=True)
cam_estoque.replace('Peixe Fresco Ariaco','Ariacó', inplace=True)
cam_estoque.replace('Ariaco Kg','Ariacó', inplace=True)
cam_estoque.replace('Peixe Fresco Guaiuba','Guaiuba', inplace=True)
cam_estoque.replace('Pargo Cong F','Pargo', inplace=True)
cam_estoque.replace('Pargo M Cong','Pargo', inplace=True)
cam_estoque.replace('File De Panga','Panga', inplace=True)
cam_estoque.replace('Lula Em Aneis Congelado','Lula', inplace=True)
cam_estoque.replace('File De Salmao C/ P Vacuo Cong','Salmão', inplace=True)
cam_estoque.replace('Macaxeira Pre Cozida','Macaxeira', inplace=True)
cam_estoque.replace('File De Camarão Pp 800G','Camarão', inplace=True)
cam_estoque.replace('Peixe Inteiro Fresco - Bejupira','Beijupirá', inplace=True)
cam_estoque.replace('Peixe Inteiro Fresco - Serra','Serra', inplace=True)
cam_estoque.replace('Tainha 1Kg Up','Tainha', inplace=True)
cam_estoque.replace('Mexilhão Desc 100/200','Mexilhão', inplace=True)
cam_estoque.replace('Charro 3/4 Pcs','Charro', inplace=True)

```

cam_estoque.replace('Pampo Real 800 1200Kg','Pampo Real', inplace=True)
cam_estoque.replace('File De Merluza','Merluza', inplace=True)
cam_estoque.replace('Peixe Bonito','Bonito', inplace=True)
cam_estoque.replace('Garajuba','Guarajuba', inplace=True)
cam_estoque.replace('Almondegas De Peixe Congelada (400G','Almôndegas de Peixe',
inplace=True)
cam_estoque.replace('Carne De Caranguejo','Carne de Caranguejo', inplace=True)
cam_estoque.replace('Ova De Peixe','Ova de Peixe', inplace=True)
cam_estoque.replace('Cabeça De Peixe','Cabeça de Peixe', inplace=True)
cam_estoque.replace('Galo Do Sul','Galo do Sul', inplace=True)
cam_estoque['TOTAL_ENTRADAS'] =
pd.to_numeric(cam_estoque['TOTAL_ENTRADAS'])
cam_estoque['ID_FORNECEDOR'] = cam_estoque['ID_FORNECEDOR'].astype(int)
cam_estoque['ID_PRODUTO'] = cam_estoque['ID_PRODUTO'].astype(int)
display(cam_estoque)
#populando a tabela
estoque_dw_cam['id_fornecedor'] = cam_estoque['ID_FORNECEDOR']
estoque_dw_cam['fornecedor'] = cam_estoque['FORNECEDOR']
estoque_dw_cam['id_produto'] = cam_estoque['ID_PRODUTO']
estoque_dw_cam['produto'] = cam_estoque['PRODUTO']
estoque_dw_cam['dt_estoque'] = cam_estoque['DT_ESTOQUE']
estoque_dw_cam['tipo_corte'] = cam_estoque['TIPO_CORTE']
estoque_dw_cam['qtd_estoque_kg'] = cam_estoque['TOTAL_ENTRADAS']
estoque_dw_cam['qtd_saida_kg'] = cam_estoque['TOTAL_SAIDAS']
estoque_dw_cam['loja'] = 'Camocim'
display(estoque_dw_cam)
#concatenando tabelas
estoque_dw = pd.concat([estoque_dw_mac,estoque_dw_cam])
display(estoque_dw)
#output
estoque_dw.to_csv(path+'TCC\\Dados\\Tratados\\estoque_dw.csv', index=False)

```

APÊNDICE C – *QUERIES* SQL

- *Query* de vendas da loja Camocim:

```
select
  dt_venda,
  hora_venda,
  qtd_clientes,
  valor_vendas,
  maior_venda
from
  `portfolio-408419.monografia_dw.vendas`
where
  loja = 'Camocim'
```

- *Query* de vendas da loja M.A.C.:

```
select
  dt_venda,
  produto,
  qtd_produtos_unid,
  valor_vendas,
  tipo_corte
from
  `portfolio-408419.monografia_dw.vendas`
where
  loja = 'Mac'
```

- *Query* de estoque da loja Camocim:

```
select
  dt_estoque,
  id_fornecedor,
  fornecedor,
  id_produto,
  produto,
  qtd_estoque_kg,
```

```
    qtd_saida_kg,  
    tipo_corte  
from  
    `portfolio-408419.monografia_dw.estoque`  
where  
    loja = 'Camocim'  
    and (qtd_estoque_kg > 0 or qtd_saida_kg > 0)
```

- *Query* de estoque da loja M.A.C.:

```
select  
    dt_estoque,  
    id_fornecedor,  
    produto,  
    valor_compra,  
    qtd_estoque_kg,  
    tipo_corte  
from  
    `portfolio-408419.monografia_dw.estoque`  
where  
    loja = 'Mac'
```

APÊNDICE D – DADOS ESTRUTURADOS NO *BIGQUERY*

Tabela 5 – Dados de vendas da loja Camocim no Bigquery.

dt_venda ▼	hora_venda ▼	qtd_clientes ▼	valor_vendas ▼	maior_venda ▼
2022-05-01	09 - 10	18	1319.6	198.0
2022-05-01	10 - 11	27	2942.1	455.0
2022-05-01	11 - 12	6	926.9	360.0
2022-05-01	12 - 13	26	4559.8	729.0
2022-05-02	14 - 15	23	13039.58	6000.0
2022-05-02	15 - 16	3	312.0	231.5
2022-05-03	08 - 09	1	95.7	95.7
2022-05-03	14 - 15	36	5309.7	1837.0
2022-05-03	15 - 16	1	17.0	17.0
2022-05-03	16 - 17	5	1021.3	482.0
2022-05-04	14 - 15	39	5263.1	1244.0
2022-05-04	16 - 17	6	807.0	262.0
2022-05-05	11 - 12	29	6021.7	2179.0
2022-05-05	14 - 15	8	1884.0	750.0
2022-05-05	16 - 17	12	2517.0	510.0
2022-05-06	09 - 10	2	325.6	166.6
2022-05-06	14 - 15	28	3861.7	708.0

Fonte: elaborado pela autora (2024).

Na coluna denominada de “dt_venda” temos a data da venda do produto; “hora_venda” é o intervalo de tempo em que a venda ocorreu; “qtd_clientes” representa o total de clientes que compraram naquele intervalo de tempo; “valor_vendas” representa o valor das vendas do intervalo de tempo; “maior_venda” indica a venda de maior valor do intervalo temporário.

Tabela 6 – Dados de vendas da loja M.A.C. no Bigquery.

dt_venda	produto	qtd_produtos_unid	valor_vendas	tipo_corte
2022-10-02	Atum	1	15.0	Posta
2022-10-06	Atum	15	225.0	Posta
2022-10-11	Atum	5	75.0	Posta
2022-10-12	Atum	1	15.0	Posta
2022-10-15	Atum	1	15.0	Posta
2022-10-16	Atum	11	220.0	Filé
2022-10-16	Atum	1	15.0	Posta
2022-10-20	Atum	15	225.0	Posta
2022-10-21	Atum	11	165.0	Posta
2022-10-27	Atum	5	75.0	Posta
2022-10-29	Atum	15	225.0	Posta
2022-09-18	Atum	20	400.0	Filé
2022-09-20	Atum	20	400.0	Filé
2022-09-15	Atum	5	75.0	Posta
2022-09-16	Atum	1	15.0	Posta
2022-10-07	Atum	1	20.0	Filé
2022-10-12	Atum	1	20.0	Filé

Fonte: elaborado pela autora (2024).

Na coluna denominada de “dt_venda” temos a data da venda do produto; “produto” é o tipo de pescado que foi vendido; “qtd_produtos_unid” representa a quantidade de produtos vendidos; “valor_vendas” representa o valor das vendas do período; “tipo_corte” indica qual foi o tipo do corte vendido.

Tabela 7 – Dados de estoque da loja Camocim no Bigquery.

dt_estoque	id_fornecedor	fornecedor	id_produto	produto	qtd_estoque_kg	qtd_saida_kg	tipo_corte
2022-05-27	5	Minerva S A	35	Truta	22.44	1.0	Filé
2022-06-10	5	Minerva S A	35	Truta	31.29	0.0	Filé
2022-06-15	5	Minerva S A	35	Truta	47.16	0.0	Filé
2022-06-17	5	Minerva S A	35	Truta	45.31	0.0	Filé
2022-06-22	5	Minerva S A	35	Truta	113.29	46.0	Filé
2022-08-04	5	Minerva S A	35	Truta	114.02	15.0	Filé
2022-08-12	5	Minerva S A	35	Truta	43.72	0.0	Filé
2022-05-23	5	Minerva S A	38	Salmão	100.0	0.0	Filé
2022-06-30	5	Minerva S A	38	Salmão	200.0	0.0	Filé
2022-08-12	5	Minerva S A	38	Salmão	200.0	40.0	Filé
2022-09-02	5	Minerva S A	38	Salmão	200.0	38.1	Filé
2022-09-19	5	Minerva S A	38	Salmão	175.9	1.0	Filé
2022-05-17	5	Minerva S A	83	Bolinho de Bacalhau	86.4	0.0	Inteiro
2022-06-28	5	Minerva S A	83	Bolinho de Bacalhau	43.2	0.0	Inteiro
2022-07-20	5	Minerva S A	83	Bolinho de Bacalhau	129.6	0.0	Inteiro
2022-08-25	5	Minerva S A	83	Bolinho de Bacalhau	12.96	0.0	Inteiro
2022-09-15	5	Minerva S A	83	Bolinho de Bacalhau	86.4	39.0	Inteiro

Fonte: elaborado pela autora (2024).

Na coluna denominada de “dt_estoque” temos a data de compra do produto;

“id_fornecedor” é o número de identificação do fornecedor; “fornecedor” indica o nome do fornecedor; “id_produto” representa o número de identificação do produto; “produto” informa o tipo de pescado comprado; “qtd_estoque_kg” representa a quantidade em quilogramas de pescado que entrou no estoque; “qtd_saida_kg” informa a quantidade em quilogramas de pescado que saiu do estoque; “tipo_corte” indica qual foi o tipo do corte do produto.

Tabela 8 – Dados de estoque da loja M.A.C. no Bigquery.

dt_estoque	id_fornecedor	produto	valor_compra	qtd_estoque_kg	tipo_corte
2022-09-10	1	Tilápia	13.0	1.0	Posta
2022-09-13	2	Atum	10.0	0.4	Posta
2022-09-14	3	Garoupa	27.0	0.1	Posta
2022-09-14	4	Camarão	21.0	0.036	Inteiro
2022-09-14	3	Camarão	40.0	0.2	Filé
2022-09-15	1	Pescadilha	15.2	1.0	Filé
2022-09-15	2	Abadejo	23.0	0.1	Filé
2022-09-15	3	Pescada	34.0	0.2	Filé
2022-09-15	5	Mapará	13.0	0.2	Filé
2022-09-15	6	Cavaia	20.0	0.5	Inteiro
2022-09-16	2	Serra	14.0	0.2	Filé
2022-09-16	2	Atum	10.0	0.2	Posta
2022-09-17	3	Camarão	40.0	0.2	Filé
2022-09-18	2	Pargo	25.0	0.3	Filé
2022-09-18	1	Goivá	15.0	0.1	Posta
2022-09-20	1	Tilápia	13.0	1.0	Posta
2022-09-20	2	Abadejo	23.0	0.1	Filé

Fonte: elaborado pela autora (2024).

Na coluna denominada de “dt_estoque” temos a data de compra do produto; “id_fornecedor” é o número de identificação do fornecedor; “produto” informa o tipo de pescado comprado; “valor_compra” representa o valor total das compras no período; “qtd_estoque_kg” indica a quantidade em quilogramas de pescado comprado; “tipo_corte” indica qual foi o tipo do corte do produto.

APÊNDICE E – ACESSO AOS *DASHBOARDS*

Acesso ao *dashboard* interativo da loja MAC Fish através do link:

<https://lookerstudio.google.com/s/hqeQOcDpM_w>.

Acesso ao *dashboard* interativo da loja Francisco Camocim: A Casa do Peixe através

do link: <<https://lookerstudio.google.com/s/pk5h78UWxGo>>.