v7.n12.2026

REVISTA CIENTÍFICA - RECIMA21 ISSN 2675-6218

AUTOMAÇÃO DE RELATÓRIOS COMERCIAIS COM ARQUITETURA MEDALHÃO E POWER BI EM UMA AGÊNCIA DE ÁLBUNS DE FORMATURA

AUTOMATION OF COMMERCIAL REPORTS WITH MEDALLION ARCHITECTURE AND POWER BI IN A GRADUATION ALBUM AGENCY

AUTOMATIZACIÓN DE INFORMES COMERCIALES CON ARQUITECTURA MEDALLÓN Y POWER BI EN UNA AGENCIA DE ÁLBUMES DE GRADUACIÓN

Gustavo Scandelae de Flores[1], Saulo Sávio Leite Santos[2], Fabiana Florian[3]

https://doi.org/10.47820/recima21.v7i12.9894

PUBLICADO: 12/2026

RESUMO

Este projeto tem como objetivo a construção de um sistema para automatizar o desenvolvimento de indicadores e relatórios comerciais em uma agência de álbuns de formatura. Foi realizado um estudo de caso com planilhas Excel de vendas e estoque da agência. O processamento foi dividido de acordo com a Arquitetura Medalhão, nas camadas: Bronze, responsável pela leitura e armazenamento dos dados; Silver, responsável pela limpeza e padronização dos dados; e Gold, responsável por realizar o cálculo dos indicadores comerciais e pela geração de nove tabelas calculadas no formato Delta Lake. Os resultados são disponibilizados por meio de uma API (*application programming interface*). Os dados disponibilizados permitem a criação de gráficos e tabelas para análise comercial no Power BI. Conclui-se que o sistema permite analisar os dados de vendas e estoque em conjunto, ajudando a agência a acompanhar as vendas e identificar novas oportunidades comerciais.

PALAVRAS-CHAVE: Arquitetura Medalhão. Automação. Delta Lake. Indicadores comerciais. Oportunidades comerciais. Power BI.

ABSTRACT

This project aims to build a system to automate the development of business indicators and reports for a graduation album agency. A case study was conducted using the agency's sales and inventory Excel spreadsheets. Processing was structured according to the Medallion Architecture, comprising three layers: Bronze, responsible for data ingestion and storage; Silver, responsible for data cleaning and standardization; and Gold, responsible for calculating business indicators and generating nine calculated tables in Delta Lake format. The results are made available via an API (application programming interface). The provided data enables the creation of charts and tables for business analysis in Power BI. The study concludes that the system allows for the joint analysis of sales and inventory data, helping the agency track sales and identify new business opportunities.

KEYWORDS: *Automation. Business indicators. Business opportunities. Delta Lake. Medallion Architecture. Power BI.*

RESUMEN

Este proyecto tiene como objetivo la construcción de un sistema para automatizar el desarrollo de indicadores e informes comerciales en una agencia de álbumes de graduación. Se realizó un estudio de caso con hojas de cálculo de Excel de ventas e inventario de la agencia.

[1]Graduando em Engenharia da Computação pela UNIARA, com interesse em análise de dados e automação.
[2]Mestre em Ciência da Computação pela UFSCar, graduado em Arquitetura e Urbanismo pelo UNICEP e docente da UNIARA, com atuação em Tecnologia e Educação.
[3]Doutora em Alimentos e Nutrição pela Universidade Estadual Paulista (UNESP), docente da Universidade de Araraquara (UNIARA), com atuação em Economia e Metodologia Científica.

Este artigo é publicado em acesso aberto (Open Access) sob a licença Creative Commons Atribuição 4.0 Internacional (CC-BY), que permite uso, distribuição e reprodução irrestritos em qualquer meio, desde que o autor original e a fonte sejam creditados.

El procesamiento se dividió de acuerdo con la Arquitectura Medallón en las capas: Bronze, responsable de la lectura y el almacenamiento de los datos; Silver, responsable de la limpieza y la estandarización de los datos; y Gold, responsable de calcular los indicadores comerciales y de generar nueve tablas calculadas en formato Delta Lake. Los resultados se ponen a disposición mediante una API (application programming interface). Los datos disponibles permiten la creación de gráficos y tablas para el análisis comercial en Power BI. Se concluye que el sistema permite analizar conjuntamente los datos de ventas e inventario, ayudando a la agencia a acompañar las ventas e identificar nuevas oportunidades comerciales.

PALABRAS CLAVE: *Arquitectura Medallón. Automatización. Delta Lake. Indicadores comerciales. Oportunidades comerciales. Power BI.*

1. INTRODUÇÃO

Nos últimos anos, a tecnologia tem evoluído significativamente, levando empresas de diversos setores a adaptar seus processos para se manterem competitivas diante daquelas que já adotaram soluções tecnológicas mais avançadas. Essas mudanças permitem simplificar atividades, facilitar a execução de tarefas e reduzir o tempo necessário para realizá-las. Nas agências de álbuns de formatura, cada novo contrato com uma instituição de ensino acrescenta registros de alunos, álbuns, vendas e estoque ao controle comercial. Historicamente, o controle desses dados operacionais, que incluem o mapeamento de potenciais clientes, o rastreamento de oportunidades perdidas (álbuns não vendidos), as taxas de conversão e planilhas de vendas, tem sido realizado de maneira manual, baseando-se muitas vezes na junção e análise de dados espalhados em diferentes fontes, como documentos físicos e tabelas Excel. Esse modo de analisar os dados limita consideravelmente a capacidade analítica da empresa e do setor de vendas.

Com o aumento da demanda e a expansão do atendimento para novas cidades, a manipulação manual dessas planilhas torna-se um gargalo operacional crítico. Na agência estudada, o autor realizava a organização e a união das planilhas de vendas e dos registros de contratos antigos e novos. Com base nessa experiência, estima-se que a atividade exigia cerca de quatro horas semanais de trabalho manual, correspondentes ao turno de uma manhã de trabalho. Durante esse processo, havia a possibilidade de erros no preenchimento e na organização das informações. A falta de organização e padronização dos dados em um só lugar dificulta a obtenção rápida de dados confiáveis, afetando diretamente a tomada de decisões. A partir das dificuldades observadas na agência, este trabalho busca responder à seguinte questão: Como é possível tornar o processo de análise desses dados mais rápido e confiável através da implementação de um *pipeline* de dados organizado pela Arquitetura Medalhão e integrado ao Power BI, que automatize o processo em uma agência de álbuns de formatura? A

Este artigo é publicado em acesso aberto (Open Access) sob a licença Creative Commons Atribuição 4.0 Internacional (CC-BY), que permite uso, distribuição e reprodução irrestritos em qualquer meio, desde que o autor original e a fonte sejam creditados.

hipótese deste trabalho é que a implementação de um *pipeline* de dados baseado na Arquitetura Medalhão possa reduzir o tempo gasto na organização e união das planilhas de vendas e estoque, além de diminuir as inconsistências presentes no tratamento manual. Espera-se que a automatização dessas etapas facilite o processamento dos dados e permita a geração de indicadores comerciais de forma mais rápida e padronizada.

O *Data Lakehouse* reúne características de duas formas de armazenamento: os *Data Warehouses*, voltados a dados estruturados, e os *Data Lakes*, que permitem armazenar dados em diferentes formatos (Armbrust *et al.*, 2021; Schneider *et al.*, 2024). Essa abordagem une a confiabilidade e o gerenciamento de transações ACID (Atomicidade, Consistência, Isolamento e Durabilidade) dos *Warehouses* com a flexibilidade de armazenamento dos *Lakes* (Armbrust *et al.*, 2021). Sendo assim, o presente trabalho desenvolve um *pipeline* (esteira de processamento) automatizado de dados baseado nessa arquitetura, utilizando a organização em camadas conhecida como Arquitetura Medalhão (*Medallion Architecture*). Segundo a Microsoft (2026), essa estrutura divide o processamento em três camadas: a camada Bronze, responsável pela leitura e armazenamento dos dados; Silver (Prata), responsável pela limpeza e padronização dos dados; e Gold (Ouro), responsável por realizar o cálculo dos indicadores comerciais e pela geração de nove tabelas calculadas. Esta pesquisa apresenta uma solução para automatizar o tratamento dos dados comerciais de uma agência de álbuns de formatura. A linguagem Python e a biblioteca Pandas são utilizadas para organizar, tratar e calcular os dados no próprio computador. O PyArrow prepara as tabelas para a gravação, e a biblioteca deltalake realiza a leitura e a escrita no formato Delta Lake. Para o consumo dessas informações, foi desenvolvida uma API REST (*Application Programming Interface* / Interface de Programação de Aplicativos) utilizando o *framework* FastAPI. Essa interface disponibiliza os indicadores e tabelas comerciais através de endereços HTTP, permitindo sua consulta pelo Power BI e por outras ferramentas. Por fim, adota-se a conteinerização de aplicativos por meio da plataforma Docker, facilitando a instalação e a execução do sistema em outros computadores.

O objetivo geral deste trabalho é desenvolver um *pipeline* de dados baseado na Arquitetura Medalhão, utilizando Pandas e Delta Lake para automatizar o tratamento dos dados de vendas e estoque e disponibilizar indicadores comerciais para uma agência de álbuns de formatura. Os objetivos específicos são: a) desenvolver a camada Bronze para a leitura e o armazenamento dos dados brutos das planilhas de vendas e estoque; b) implementar a camada Silver com Pandas para limpar, padronizar e reunir os dados de vendas e estoque; c) estruturar a camada Gold para calcular indicadores comerciais, considerando os registros de álbuns vendidos e não vendidos; d) construir uma API REST para disponibilizar os indicadores da

Este artigo é publicado em acesso aberto (Open Access) sob a licença Creative Commons Atribuição 4.0 Internacional (CC-BY), que permite uso, distribuição e reprodução irrestritos em qualquer meio, desde que o autor original e a fonte sejam creditados.

camada Gold ao Power BI; e) conteinerizar a aplicação com Docker para facilitar sua instalação e execução em outros computadores.

Esta pesquisa classifica-se como aplicada e exploratória, com abordagem quantitativa, reforçada pela observação da rotina de organização das planilhas na agência. Foi realizada uma pesquisa bibliográfica para compreender os conceitos de *Data Lakehouse*, Arquitetura Medalhão e Delta Lake. Também foi desenvolvido um *pipeline* automatizado de dados aplicado ao contexto de uma agência de álbuns de formatura, caracterizando o trabalho como um estudo de caso. Os dados foram obtidos de um arquivo Excel fornecido pela agência, contendo registros de vendas e de álbuns em estoque, e processados em lote com Pandas. O sistema foi desenvolvido em etapas, com a implementação das camadas Bronze, Silver e Gold, seguida pela criação da interface de programação de aplicações (API) e pela conteinerização com Docker.

2. REFERENCIAL TEÓRICO

São abordados em seguida os conceitos de Engenharia de Dados, *Data Warehouse*, *Data Lake* e *Data Lakehouse*, além da organização do processamento pela Arquitetura Medalhão. Também são abordados o uso de Python e das bibliotecas Pandas, PyArrow e deltalake, a disponibilização dos dados por meio de uma API REST e a conteinerização da aplicação com Docker.

2.1. Engenharia de dados e a evolução das arquiteturas

Os *Data Warehouses* são ambientes organizados para armazenar dados estruturados e facilitar consultas e análises. Apesar dessas vantagens, sua estrutura pode dificultar a inclusão de grandes volumes de dados com formatos variados (Azzabi; Alfughi; Ouda, 2024). Nos *Data Lakes*, os dados podem ser armazenados em diferentes formatos, sem exigir uma estrutura única antes da entrada. A estrutura necessária para utilizá-los é definida durante a leitura (Azzabi; Alfughi; Ouda, 2024).

Os *Data Lakes* permitem armazenar diferentes tipos de dados em um mesmo ambiente, porém essa flexibilidade exige uma organização adequada das informações. Quando os dados são armazenados sem critérios de qualidade e padronização, podem surgir dificuldades para localizá-los, interpretá-los e utilizá-los nas análises, comprometendo a confiabilidade das informações (Azzabi; Alfughi; Ouda, 2024).

Este artigo é publicado em acesso aberto (Open Access) sob a licença Creative Commons Atribuição 4.0 Internacional (CC-BY), que permite uso, distribuição e reprodução irrestritos em qualquer meio, desde que o autor original e a fonte sejam creditados.

2.2. O paradigma do Data Lakehouse e a Arquitetura Medalhão

O *Data Lakehouse* reúne a flexibilidade dos *Data Lakes* com recursos encontrados nos *Data Warehouses*, como organização dos dados, controle das operações e suporte a consultas analíticas (Armbrust *et al.*, 2021; Schneider *et al.*, 2024). Dessa forma, dados brutos e dados preparados para análise podem fazer parte de uma mesma arquitetura.

A Arquitetura Medalhão organiza o processamento dos dados em três camadas, chamadas Bronze, Silver e Gold, para melhorar sua qualidade e prepará-los para análise (Microsoft, 2026). Na Bronze, os dados são recebidos e armazenados próximos de seu formato original, podendo incluir informações sobre sua origem e o momento da entrada. Quando as versões anteriores são preservadas, essa camada também permite consultar o histórico das cargas. A Silver utiliza os dados da Bronze para realizar a limpeza, a padronização e a união das bases, incluindo a conversão dos tipos dos campos e o tratamento de valores ausentes e de diferentes formas de registrar a mesma informação. Por fim, a Gold aplica as regras de negócio e gera tabelas destinadas a relatórios e análises, que, no contexto comercial, podem apresentar indicadores como taxa de conversão, *ticket* médio e desempenho de vendas.

2.3. Processamento de dados com Pandas, PyArrow e Delta Lake

O Pandas é uma biblioteca da linguagem Python voltada à organização e à análise de dados. Sua estrutura DataFrame representa os dados em linhas e colunas e permite realizar operações como leitura de arquivos, tratamento de campos, união de bases e agrupamentos (Pandas, [ca. 2026]). Neste projeto, essas operações são executadas na memória do computador para preparar os registros de vendas e estoque e calcular os indicadores comerciais.

O Delta Lake acrescenta um *log* de transações aos arquivos de dados e oferece suporte a transações ACID, formadas pelos princípios de atomicidade, consistência, isolamento e durabilidade. Esses recursos ajudam a manter a consistência das tabelas durante operações de leitura e gravação (Schneider *et al.*, 2024). No projeto, as tabelas das três camadas são armazenadas nesse formato. A biblioteca deltalake permite ler e gravar essas tabelas diretamente em Python (Delta-rs, [ca. 2026]). O PyArrow permite converter um DataFrame do Pandas em uma tabela Arrow, utilizada pelo programa antes da gravação em Delta Lake (Apache Arrow, [ca. 2026]).

A utilização de bibliotecas em Python também exige a organização das dependências do projeto. Para isso, ambientes virtuais permitem manter um conjunto próprio de bibliotecas, separado de outros projetos. A ferramenta uv auxilia esse gerenciamento ao criar o ambiente e

Este artigo é publicado em acesso aberto (Open Access) sob a licença Creative Commons Atribuição 4.0 Internacional (CC-BY), que permite uso, distribuição e reprodução irrestritos em qualquer meio, desde que o autor original e a fonte sejam creditados.

instalar as dependências, utilizando um arquivo que registra as versões dos pacotes e facilita a reprodução do ambiente de execução (Astral, [ca. 2026]).

2.4. Disponibilização de dados por API REST

REST é um estilo de arquitetura que estabelece restrições para a interação entre recursos em aplicações *Web*. APIs orientadas a recursos utilizam HTTP e endereços URI para disponibilizar funcionalidades a outras aplicações (Bogner; Kotstein; Pfaff, 2023).

O FastAPI é um *framework* da linguagem Python voltado à criação de APIs (FastAPI, [ca. 2026]). Neste projeto, ele é utilizado para ler os resultados da camada Gold e disponibilizá-los em formato JSON por meio de endereços HTTP no navegador. Assim, os indicadores podem ser consultados pelo Power BI e por outras aplicações compatíveis.

2.5. Conteinerização com Docker

A conteinerização reúne a aplicação e suas dependências em um ambiente isolado chamado contêiner. Esses ambientes incluem as bibliotecas e os artefatos necessários à execução, facilitando a transferência da aplicação entre ambientes compatíveis com contêineres (Di Modica; Foschini, 2023). Neste trabalho, o Docker é utilizado para preparar o ambiente da API e permitir criar um contêiner para facilitar sua execução em outros computadores.

Em conjunto, a Arquitetura Medalhão organiza as etapas do processamento, enquanto o Delta Lake permite armazenar as tabelas produzidas e registrar as operações realizadas sobre elas. A padronização dos dados ajuda a reunir registros escritos de formas diferentes, e a integração entre a API e o Power BI permite consultar e apresentar os resultados. Neste trabalho, esses recursos foram aplicados à rotina da agência para automatizar a preparação das planilhas, facilitar a análise de vendas e estoque e apoiar o planejamento das futuras ações de venda.

3. DESENVOLVIMENTO

O desenvolvimento partiu dos registros de vendas e estoque de uma agência de álbuns de formatura, organizados em um arquivo Excel. A linguagem Python foi utilizada para preparar esses dados e calcular os indicadores comerciais, com o apoio das bibliotecas Pandas, PyArrow e deltalake. O processamento seguiu a Arquitetura Medalhão, dividindo as tarefas entre as camadas Bronze, Silver e Gold. Para consultar as tabelas produzidas, foi desenvolvida uma API com FastAPI, enquanto o Docker reuniu as dependências necessárias à execução da aplicação.

Este artigo é publicado em acesso aberto (Open Access) sob a licença Creative Commons Atribuição 4.0 Internacional (CC-BY), que permite uso, distribuição e reprodução irrestritos em qualquer meio, desde que o autor original e a fonte sejam creditados.

Por fim, o Power BI foi utilizado para importar os dados disponibilizados pela API e organizar sua apresentação em relatórios comerciais.

A Figura 1 apresenta o fluxo principal do projeto, desde a leitura do arquivo Excel até a apresentação dos indicadores no Power BI.

Figura 1. Fluxo geral da arquitetura do projeto



Fonte: Elaboração própria (2026).

Este artigo é publicado em acesso aberto (Open Access) sob a licença Creative Commons Atribuição 4.0 Internacional (CC-BY), que permite uso, distribuição e reprodução irrestritos em qualquer meio, desde que o autor original e a fonte sejam creditados.

3.1. Organização do projeto e dados utilizados

A fonte principal do projeto foi o arquivo HISTORICO_ALBUNS.xlsx, fornecido pela agência e composto pelas abas VENDAS e ESTOQUE. A primeira reúne os álbuns comercializados, com informações como instituição, contrato, álbum, curso, cidade, vendedor e valor da venda. A segunda contém os álbuns ainda disponíveis, incluindo informações sobre os formandos, a quantidade de fotos de cada álbum e a unidade federativa. Essas duas bases permitem analisar tanto as vendas realizadas quanto os registros que ainda podem representar oportunidades comerciais. O arquivo é colocado na pasta data/landing e permanece como fonte de entrada, sem alterações pelo programa. Para que a leitura seja executada corretamente, os nomes das abas e das colunas devem corresponder à estrutura prevista no código (VENDAS e ESTOQUE).

Além da base comercial, foi utilizada uma relação de municípios como apoio à padronização das cidades. As tabelas da Divisão Territorial Brasileira de 2025, disponibilizadas pelo IBGE, constituem a referência oficial para os nomes dos municípios e suas unidades federativas (IBGE, 2026). No projeto, a relação utilizada está armazenada no arquivo municipios_ibge.csv, na pasta src/referencias, permitindo que o tratamento consulte os nomes localmente. A tabela de referência do IBGE auxilia na identificação dos municípios e na correção de grafias divergentes, contribuindo para a padronização dos registros. Os casos sem correspondência segura são separados para revisão, evitando que uma correção automática atribua o registro ao município errado.

A organização das pastas acompanha as etapas do processamento (Figura 2). A pasta src reúne os programas e os arquivos de referência, enquanto data contém a entrada e as tabelas produzidas. Dentro de data, landing recebe o Excel contendo os dados fornecidos pela empresa, bronze armazena as duas bases lidas, silver contém o histórico consolidado corrigido e padronizado e gold reúne os indicadores comerciais. O arquivo config.py centraliza os caminhos dessas pastas, o nome do Excel e os nomes das tabelas Gold, para que os programas utilizem as mesmas referências. A execução das três camadas é coordenada pelo orquestrador.py, que chama cada etapa na ordem necessária.

Este artigo é publicado em acesso aberto (Open Access) sob a licença Creative Commons Atribuição 4.0 Internacional (CC-BY), que permite uso, distribuição e reprodução irrestritos em qualquer meio, desde que o autor original e a fonte sejam creditados.

Figura 2. Organização das pastas e dos arquivos do projeto



Fonte: Elaboração própria (2026).

3.2. Implementação do processamento de dados

O processamento foi implementado em Python, seguindo a divisão de responsabilidades da Arquitetura Medalhão descrita pela Microsoft (2026). Na Bronze, a biblioteca Pandas realiza a leitura separada das abas VENDAS e ESTOQUE e verifica a presença das colunas obrigatórias. Os identificadores de contrato e álbum são tratados como texto, pois não são utilizados em cálculos, e cada registro recebe informações sobre o arquivo de origem, a aba e o momento da leitura. Em seguida, o PyArrow prepara os dados para a gravação e a biblioteca deltalake armazena cada base em uma tabela Delta. A gravação substitui o conteúdo da versão atual a cada execução, de modo que uma nova carga não acrescente novamente os mesmos registros. Esse armazenamento permite que as etapas seguintes trabalhem com as tabelas geradas, mantendo o Excel como arquivo de origem, sem alterá-lo.

Na Silver, as duas tabelas passam por tratamentos específicos antes da consolidação. Os cursos sofrem ajustes de texto e são padronizados por meio de correções de abreviações, listas de referência e comparação aproximada de textos, sendo também classificados por nível

Este artigo é publicado em acesso aberto (Open Access) sob a licença Creative Commons Atribuição 4.0 Internacional (CC-BY), que permite uso, distribuição e reprodução irrestritos em qualquer meio, desde que o autor original e a fonte sejam creditados.

educacional. Para as cidades, assim como nos cursos, são ajustados espaços, acentos e abreviações, e os nomes são comparados com a referência de municípios. Quando uma UF válida está disponível, a busca é limitada aos municípios do respectivo estado. Na ausência dessa informação, o algoritmo exige maior semelhança entre os nomes para reduzir o risco de associações incorretas. Nomes curtos ou ambíguos são mantidos para conferência, e o arquivo revisao_cidades.csv registra os resultados do tratamento e fica salvo no diretório data/silver. As informações de instituição, formando, unidade federativa e quantidade de fotos também recebem os ajustes e padronizações previstos para cada base, conforme os campos disponíveis.

Os registros provenientes de VENDAS recebem o valor 1 no campo, criado pela camada Silver, que indica a compra do álbum, enquanto os registros de ESTOQUE recebem o valor 0. Os valores de venda são convertidos para números decimais, considerando e padronizando também os casos em que aparecem como texto com símbolo monetário e separadores. Na base de estoque, o valor da venda permanece ausente, pois o álbum ainda não foi comercializado. Depois desses tratamentos, o Pandas reúne as linhas das duas bases em uma estrutura em comum, preenchendo com valores nulos os campos inexistentes em uma delas. Em seguida, assim como na camada Bronze, o PyArrow prepara os dados para a gravação e a biblioteca deltalake armazena a tabela Delta. A tabela de histórico consolidado é gravada na camada Silver, no diretório data/silver, e utilizada como entrada da camada Gold. Essa reunião dos registros permite calcular os indicadores a partir de uma única tabela, preservando a distinção entre vendas e estoque.

A camada Gold utiliza o histórico consolidado para produzir nove tabelas destinadas à análise comercial. A taxa de conversão relaciona a quantidade de registros vendidos ao total de registros mapeados e é calculada por instituição, cidade e nível educacional, além do resultado geral. O *ticket* médio considera os valores dos registros de venda, e o *ranking* de vendedores apresenta a quantidade de vendas, a receita e o valor médio por vendedor. Uma tabela de resumo reúne os principais indicadores gerais. Esses cálculos são realizados com os recursos de agrupamento do Pandas (Pandas, [ca. 2026]).

As outras três tabelas da Gold apresentam oportunidades comerciais por cidade, instituição e curso. Para estimar o valor associado aos álbuns não vendidos, o programa utiliza o *ticket* médio das vendas do respectivo grupo. Quando não há vendas que sirvam de referência naquele grupo, aplica-se o *ticket* médio geral. A soma desses valores apresenta uma estimativa do potencial de vendas do estoque. Como os mesmos álbuns não vendidos são analisados por cidade, instituição e curso, as estimativas desses agrupamentos devem ser consideradas separadamente, evitando a contagem repetida das mesmas oportunidades comerciais. Ao final

Este artigo é publicado em acesso aberto (Open Access) sob a licença Creative Commons Atribuição 4.0 Internacional (CC-BY), que permite uso, distribuição e reprodução irrestritos em qualquer meio, desde que o autor original e a fonte sejam creditados.

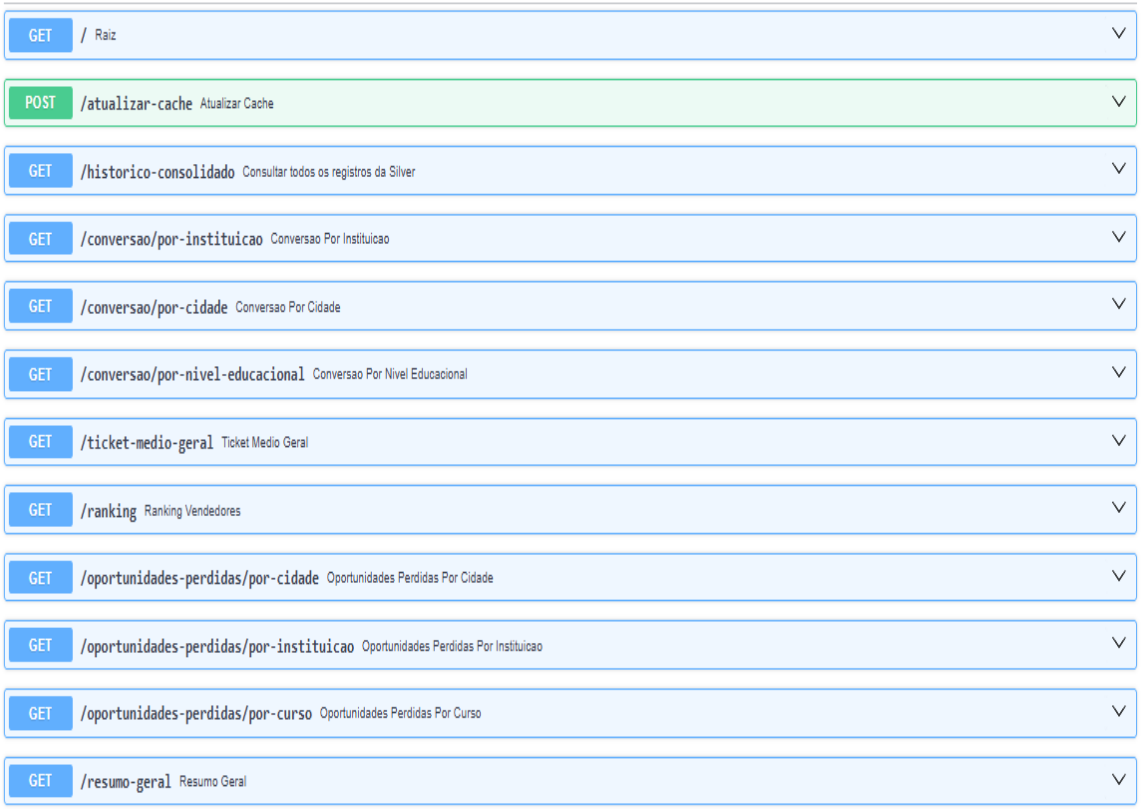
do processamento, todas as tabelas são gravadas em formato Delta Lake, cuja estrutura de arquivos e registro de transações é discutida por Schneider *et al.* (2024). Os resultados ficam disponíveis para consulta pela API e importação no Power BI.

3.3. Disponibilização dos dados, ambiente de execução e conteinerização

A disponibilização dos dados foi implementada com FastAPI, utilizando endereços HTTP que retornam as informações em formato JSON (FastAPI, [ca. 2026]). Ao iniciar, a API lê as nove tabelas da Gold e o histórico consolidado da Silver, mantendo seus registros em memória para responder às consultas. Essa organização permite acessar tanto indicadores já calculados quanto os dados tratados que lhes deram origem. Cada endereço corresponde a uma consulta específica, como resumo geral, *ranking* de vendedores ou conversão por cidade.

Caso uma tabela não possa ser carregada, sua consulta informa a indisponibilidade, enquanto as demais tabelas carregadas permanecem acessíveis. A documentação interativa auxilia a conferência das consultas, como apresentado na Figura 3.

Figura 3. Documentação interativa das consultas da API



Fonte: Elaboração própria (2026).

Este artigo é publicado em acesso aberto (Open Access) sob a licença Creative Commons Atribuição 4.0 Internacional (CC-BY), que permite uso, distribuição e reprodução irrestritos em qualquer meio, desde que o autor original e a fonte sejam creditados.

Para preparar o ambiente de execução, foi utilizada a ferramenta uv, responsável pela criação do ambiente virtual e pela instalação das dependências do projeto. As bibliotecas necessárias são declaradas no arquivo pyproject.toml, enquanto o uv.lock registra as versões utilizadas. Essa organização mantém as dependências separadas das bibliotecas de outros projetos e facilita a preparação do ambiente em outro computador, permitindo instalar as dependências declaradas em uma única operação.

A execução foi preparada também em um ambiente Docker, que reúne a aplicação e suas dependências, seguindo o uso de contêineres discutido por Di Modica e Foschini (2023). O Dockerfile define um ambiente com Python e utiliza o uv para instalar as dependências do projeto conforme as versões registradas no arquivo uv.lock. O Docker Compose, por sua vez, organiza a inicialização da API e sua ligação com a pasta data do computador. Essa ligação permite que o contêiner leia as tabelas produzidas e que os arquivos permaneçam armazenados fora dele. O serviço foi configurado para acesso local pela porta 8000. O mesmo ambiente pode executar o processamento das três camadas antes de iniciar a API.

3.4. Integração com o Power BI

A integração com o Power BI Desktop é realizada por meio das consultas *Web* aos endereços da API. As respostas recebidas em JSON são convertidas em tabelas no Power Query, com a expansão dos registros e a conferência dos tipos das colunas. Quantidades são tratadas como números inteiros, valores monetários e taxas como números decimais, e campos como cidade, instituição e vendedor como texto. Cada consulta importa os dados do endereço informado, permitindo selecionar as tabelas necessárias ao relatório. Os indicadores da Gold oferecem informações já agrupadas e calculadas, enquanto o histórico consolidado da Silver permite trabalhar com os registros de vendas e estoque tratados.

As tabelas importadas fornecem a base para apresentar os totais gerais, comparar a conversão entre grupos, acompanhar o desempenho comercial e verificar a integridade dos dados com a tabela consolidada. O resumo geral pode alimentar cartões de indicadores, enquanto os agrupamentos por cidade e instituição permitem construir gráficos e tabelas de comparação. Os dados de vendedores e de oportunidades comerciais também ficam disponíveis para compor as análises conforme a finalidade do relatório. A atualização dos dados exige uma sequência definida: após a substituição do Excel de entrada na pasta data/landing, o processamento das camadas Bronze, Silver e Gold deve ser executado novamente. Se a API já estiver em funcionamento, seu *cache* deve ser recarregado antes da atualização no Power BI,

Este artigo é publicado em acesso aberto (Open Access) sob a licença Creative Commons Atribuição 4.0 Internacional (CC-BY), que permite uso, distribuição e reprodução irrestritos em qualquer meio, desde que o autor original e a fonte sejam creditados.

por meio de uma requisição POST à rota /atualizar-*cache*, que pode ser executada pela documentação interativa da API.

Durante a atualização, o Power Query consulta os endereços da API previamente configurados no relatório e atualiza as respectivas tabelas importadas. O Power Query não reprocessa as planilhas; esse processamento é realizado pelos programas das camadas Bronze, Silver e Gold.

4. RESULTADOS E DISCUSSÃO

O processamento dos dados da agência resultou em um histórico consolidado com 9.361 registros, formado por 3.166 registros de vendas e 6.195 registros de álbuns em estoque. A reunião das duas bases permitiu consultar essas informações em uma estrutura comum, mantendo a identificação da origem de cada registro e a distinção entre álbuns vendidos e não vendidos. A partir desse histórico, foram produzidas nove tabelas na camada Gold, com indicadores gerais, taxas de conversão, desempenho de vendedores e estimativas de oportunidades comerciais. Dessa forma, informações que antes precisavam ser organizadas manualmente passaram a ser preparadas pelo programa e disponibilizadas para consulta pela API. Os totais apresentados correspondem à contagem dos registros processados.

Na base analisada, a receita registrada nas vendas foi de R$ 8.391.784,00, com *ticket* médio de R$ 2.650,60. A relação entre os registros vendidos e o total mapeado resultou em uma taxa de conversão geral de 33,82%. Esses indicadores permitem observar, em conjunto, o volume comercializado, o valor médio das vendas e a participação dos registros vendidos na base. Por isso, a conversão deve ser interpretada considerando o conjunto de dados fornecido pela agência, sem atribuir aos resultados um período de vendas que não tenha sido delimitado na fonte. As informações gerais estão disponíveis na tabela de resumo e podem compor a visão geral do relatório no Power BI (Figura 4).

A taxa de conversão de 33,82% indica que aproximadamente um terço dos registros analisados corresponde a álbuns vendidos. Os demais 66,18% representam álbuns em estoque, que podem significar novas ações comerciais e oportunidades futuras. Os agrupamentos por cidade e curso ajudam a identificar onde esse estoque está concentrado e a planejar a abordagem aos possíveis compradores.

Este artigo é publicado em acesso aberto (Open Access) sob a licença Creative Commons Atribuição 4.0 Internacional (CC-BY), que permite uso, distribuição e reprodução irrestritos em qualquer meio, desde que o autor original e a fonte sejam creditados.

Figura 4. Página de visão geral do relatório final



Fonte: Elaboração própria (2026).

Os agrupamentos por cidade, instituição e nível educacional permitem comparar a quantidade de registros mapeados com as vendas de cada grupo. A Figura 5 apresenta os indicadores por cidade no Power BI. Essa leitura ajuda a identificar diferenças de conversão e a selecionar os grupos que merecem uma análise comercial mais detalhada. A padronização dos nomes de cidades e cursos contribui para reunir registros que poderiam aparecer separados por abreviações ou diferenças de escrita. Entretanto, os casos sem correspondência segura permanecem sujeitos à conferência, e a interpretação dos indicadores depende da qualidade das informações de origem. O histórico consolidado disponibilizado pela API permite consultar os registros tratados e relacioná-los às análises produzidas, enquanto as tabelas Gold oferecem os agrupamentos já calculados para apresentação no Power BI.

O relatório de revisão das cidades permitiu acompanhar os resultados da padronização. Dos 9.361 registros processados, 9.237 tiveram o nome da cidade reconhecido na lista de referência após os ajustes de texto. Separadamente, outros 42 registros receberam correções automáticas, sendo nove por expansão de abreviações e 33 por comparação aproximada dos nomes. Essas correções envolveram 22 grafias diferentes presentes na planilha. Dos 82 registros restantes, 80 foram sinalizados para conferência, sendo 57 pelo nome da cidade e 23 pela unidade federativa informada, enquanto dois estavam sem informação de cidade.

Este artigo é publicado em acesso aberto (Open Access) sob a licença Creative Commons Atribuição 4.0 Internacional (CC-BY), que permite uso, distribuição e reprodução irrestritos em qualquer meio, desde que o autor original e a fonte sejam creditados.

Figura 5. Página de detalhamento por cidades do relatório final



Fonte: Elaboração própria (2026).

A tabela de *ranking* reúne a quantidade de vendas, a receita e o *ticket* médio por vendedor, permitindo comparar os resultados registrados na base. Esses valores oferecem diferentes perspectivas de desempenho, pois uma maior quantidade de vendas não implica, necessariamente, maior receita ou maior valor médio por venda. Já as tabelas de oportunidades comerciais apresentam estimativas associadas aos álbuns em estoque, agrupadas por cidade, instituição e curso. As estimativas utilizam os valores médios das vendas existentes como referência e podem apoiar a seleção de grupos para acompanhamento comercial. Como os agrupamentos utilizam os mesmos registros de estoque, seus valores devem ser analisados separadamente, evitando que uma mesma oportunidade seja contabilizada mais de uma vez.

O potencial comercial estimado por curso totalizou R$ 17.163.841,39 (Figura 6). Esse valor foi calculado multiplicando a quantidade de álbuns em estoque de cada curso pelo *ticket* médio das vendas do respectivo curso. Quando não há vendas de referência, o programa utiliza o *ticket* médio geral. Como exemplo ilustrativo, um curso com dez álbuns em estoque e *ticket* médio de R$ 2.500,00 teria um potencial estimado de R$ 25.000,00. O resultado indica o valor que esse estoque poderia representar com base nas vendas registradas, sem considerar a probabilidade de compra de cada cliente e sem garantir que os álbuns serão vendidos.

Este artigo é publicado em acesso aberto (Open Access) sob a licença Creative Commons Atribuição 4.0 Internacional (CC-BY), que permite uso, distribuição e reprodução irrestritos em qualquer meio, desde que o autor original e a fonte sejam creditados.

Figura 6. Página de oportunidades comerciais por curso do relatório final



Fonte: Elaboração própria (2026).

A disponibilização das tabelas pela API permite utilizar os dados tratados e os indicadores como fontes de consulta no Power BI, sem reunir novamente as planilhas a cada atualização. Após uma nova execução do processamento, a recarga do *cache* disponibiliza as tabelas atualizadas para as consultas já configuradas.

Com base na experiência do autor na agência, o procedimento manual exigia aproximadamente quatro horas semanais. Para avaliar o tempo do procedimento automatizado, foram cronometradas seis execuções e calculada a média dos tempos registrados, que foi de aproximadamente seis minutos. A medição começou com a colocação da planilha em data/landing e terminou no clique em “Atualizar” no Power BI, incluindo a execução das três camadas e a disponibilização dos dados pela API. Assim, esse intervalo não inclui o tempo necessário para concluir a atualização das consultas no Power BI. A comparação entre os 240 minutos semanais estimados para a organização e a união manual das planilhas e a média de seis minutos cronometrados no procedimento automatizado indica uma redução aproximada de 97,5%. Esse percentual se refere à comparação entre a estimativa do trabalho manual e o intervalo medido até o clique em “Atualizar” no Power BI.

Este artigo é publicado em acesso aberto (Open Access) sob a licença Creative Commons Atribuição 4.0 Internacional (CC-BY), que permite uso, distribuição e reprodução irrestritos em qualquer meio, desde que o autor original e a fonte sejam creditados.

5. CONSIDERAÇÕES FINAIS

O trabalho alcançou o objetivo de desenvolver uma solução para automatizar o tratamento dos dados de vendas e estoque de uma agência de álbuns de formatura e disponibilizar indicadores comerciais para consulta. A organização pela Arquitetura Medalhão permitiu separar a leitura das planilhas, a limpeza e a consolidação dos registros e o cálculo dos indicadores. O uso de Python e das bibliotecas do projeto tornou possível executar essas etapas em sequência, enquanto a API reuniu os resultados em uma forma acessível às consultas do Power BI. A contribuição da solução está na preparação dos dados comerciais a partir de um procedimento definido, que pode ser repetido quando a agência fornece uma nova versão da planilha.

O procedimento automatizado apresentou tempo médio de aproximadamente seis minutos em seis execuções cronometradas pelo autor. Em comparação com a estimativa de quatro horas semanais dedicadas ao processo manual, esse resultado indica uma redução aproximada de 97,5% no tempo considerado. A comparação utiliza uma estimativa do trabalho manual e medições do procedimento automatizado, devendo ser interpretada dentro dessas condições. Além da economia de tempo, o sistema permite repetir o tratamento dos dados e o cálculo dos indicadores com as mesmas regras, reduzindo a necessidade de organizar manualmente as bases a cada atualização.

Os resultados dependem da estrutura e da qualidade das planilhas de entrada, e registros ambíguos ou inconsistentes ainda podem exigir conferência. Além disso, as estimativas de oportunidades comerciais devem ser entendidas como apoio à análise do estoque, sem representar vendas futuras garantidas. Como continuidade, podem ser aprimorados os relatórios no Power BI, as verificações de qualidade dos dados e o agendamento das etapas de processamento e atualização. A solução também pode servir como base para empresas que utilizem planilhas de vendas e estoque e tenham necessidades semelhantes de organização e análise. Essa adaptação exige avaliar os campos disponíveis, as características da atividade e as regras comerciais de cada empresa, antes de aplicar o mesmo modelo de indicadores.

DECLARAÇÃO DE USO DE INTELIGÊNCIA ARTIFICIAL

Durante a preparação deste manuscrito, foi utilizada ferramenta de inteligência artificial generativa (Claude, Anthropic) na tradução do resumo para o espanhol e na adaptação do manuscrito ao modelo de formatação da revista. Todo o conteúdo foi revisado pelos autores, que assumem integral responsabilidade pelo texto.

Este artigo é publicado em acesso aberto (Open Access) sob a licença Creative Commons Atribuição 4.0 Internacional (CC-BY), que permite uso, distribuição e reprodução irrestritos em qualquer meio, desde que o autor original e a fonte sejam creditados.

FINANCIAMENTO

Esta pesquisa não recebeu financiamento de agências de fomento, instituições ou empresas.

CONFLITO DE INTERESSES

Um dos autores atuou na agência em que o estudo de caso foi realizado. Os autores declaram não haver outros conflitos de interesses.

REFERÊNCIAS

APACHE ARROW. **Pandas integration**. [ca. 2026]. Disponível em: https://arrow.apache.org/docs/python/pandas.html. Acesso em: 22 jul. 2026.

ARMBRUST, M. *et al.* Lakehouse: a new generation of open platforms that unify data warehousing and advanced analytics. *In*: CONFERENCE ON INNOVATIVE DATA SYSTEMS RESEARCH (CIDR), 11., 2021, [online]. **Proceedings** [...]. [S. l.]: CIDR, 2021. Disponível em: https://www.cidrdb.org/cidr2021/papers/cidr2021_paper17.pdf. Acesso em: 17 maio 2026.

ASTRAL. **Structure and files**. [ca. 2026]. Disponível em: https://docs.astral.sh/uv/concepts/projects/layout/. Acesso em: 20 jul. 2026.

AZZABI, S.; ALFUGHI, Z.; OUDA, A. Data lakes: a survey of concepts and architectures. **Computers**, v. 13, n. 7, art. 183, 2024. DOI: https://doi.org/10.3390/computers13070183.

BOGNER, J.; KOTSTEIN, S.; PFAFF, T. Do RESTful API design rules have an impact on the understandability of Web APIs? **Empirical Software Engineering**, v. 28, art. 132, 2023. DOI: https://doi.org/10.1007/s10664-023-10367-y.

DELTA-RS. **Writing Delta tables**. [ca. 2026]. Disponível em: https://delta-io.github.io/delta-rs/latest/usage/writing/. Acesso em: 22 jul. 2026.

DI MODICA, G.; FOSCHINI, L. A survey on the use of lightweight virtualization in I4.0 manufacturing environments. **Journal of Network and Systems Management**, v. 31, art. 37, 2023. DOI: https://doi.org/10.1007/s10922-023-09725-4.

FASTAPI. **FastAPI documentation**. [ca. 2026]. Disponível em: https://fastapi.tiangolo.com/. Acesso em: 22 jul. 2026.

IBGE. **Divisão Territorial Brasileira 2025**. Rio de Janeiro: IBGE, 2026. Disponível em: https://www.ibge.gov.br/geociencias/organizacao-do-territorio/estrutura-territorial/23701-divisao-territorial-brasileira.html. Acesso em: 26 set. 2026.

MICROSOFT. **What is the medallion lakehouse architecture?** Microsoft Learn, 11 set. 2026. Disponível em: https://learn.microsoft.com/en-us/azure/databricks/lakehouse/medallion. Acesso em: 3 nov. 2026.

Este artigo é publicado em acesso aberto (Open Access) sob a licença Creative Commons Atribuição 4.0 Internacional (CC-BY), que permite uso, distribuição e reprodução irrestritos em qualquer meio, desde que o autor original e a fonte sejam creditados.

PANDAS. **Package overview**. [ca. 2026]. Disponível em: https://pandas.pydata.org/docs/getting_started/overview.html. Acesso em: 16 maio 2026.

SCHNEIDER, J. *et al.* The lakehouse: state of the art on concepts and technologies. **SN Computer Science**, v. 5, art. 449, 2024. DOI: https://doi.org/10.1007/s42979-024-02737-0.

Este artigo é publicado em acesso aberto (Open Access) sob a licença Creative Commons Atribuição 4.0 Internacional (CC-BY), que permite uso, distribuição e reprodução irrestritos em qualquer meio, desde que o autor original e a fonte sejam creditados.