

**CENTRO FEDERAL DE EDUCAÇÃO TECNOLÓGICA DE MINAS GERAIS
CAMPUS TIMÓTEO**

Matheus Bicalho de Freitas

**MODELAGEM DIMENSIONAL DE UM *DATA WAREHOUSE* PARA
ANÁLISE DE DADOS ACADÊMICOS DO CEFET-MG CAMPUS
TIMÓTEO**

Timóteo

2023

Matheus Bicalho de Freitas

**MODELAGEM DIMENSIONAL DE UM *DATA WAREHOUSE* PARA
ANÁLISE DE DADOS ACADÊMICOS DO CEFET-MG CAMPUS
TIMÓTEO**

Monografia apresentada à Coordenação de Engenharia de Computação do Campus Timóteo do Centro Federal de Educação Tecnológica de Minas Gerais para obtenção do grau de Bacharel em Engenharia de Computação.

Orientador: Odilon Côrrea

Timóteo

2023

Matheus Bicalho de Freitas

**MODELAGEM DIMENSIONAL DE UM DATA WAREHOUSE PARA
ANÁLISE DE DADOS ACADÊMICOS DO CEFET-MG CAMPUS
TIMÓTEO**

Trabalho de Conclusão de Curso
apresentado ao Curso de Engenharia de Computação
do Centro Federal de Educação Tecnológica de
Minas Gerais, campus Timóteo, como requisito
parcial para obtenção do título de Engenheiro de
Computação.

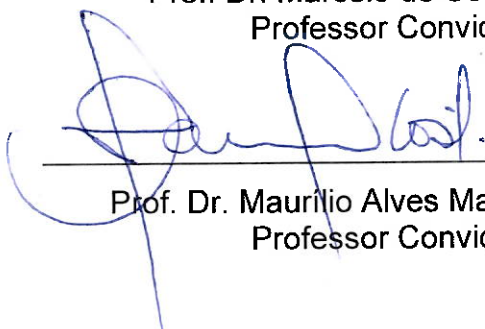
Trabalho aprovado. Timóteo, 13 de dezembro de 2023:



Prof. Me. Odilon Corrêa da Silva
Orientador



Prof. Dr. Marcelo de Sousa Balbino
Professor Convidado



Prof. Dr. Maurílio Alves Martins da Costa
Professor Convidado

Timóteo
2023

.

.

.

Dedico aos meus pais Adenilton e Rosimar
e aos meus irmãos Philipe e Lucas.

Agradecimentos

Gostaria de agradecer a todos aqueles que contribuíram para a realização deste trabalho. Em primeiro lugar, agradeço ao meu orientador, cuja orientação e apoio foram fundamentais para o sucesso deste projeto. Suas valiosas sugestões e conselhos foram essenciais para aprimorar meu conhecimento e habilidades no desenvolvimento deste trabalho. Gostaria também de agradecer aos meus colegas de classe e amigos, que me apoiaram em momentos difíceis e compartilharam suas experiências e conhecimentos comigo. Não posso deixar de mencionar minha família, que sempre me encorajou e me motivou a alcançar meus objetivos acadêmicos. Finalmente, gostaria de agradecer a todos aqueles que, de uma forma ou de outra, contribuíram para a realização deste trabalho e para o meu crescimento pessoal e profissional. Muito obrigado!

*“O sucesso não é final,
o fracasso não é fatal: é a coragem para continuar que conta.”.*
Winston Churchill

Resumo

O presente trabalho visa desenvolver uma modelagem dimensional de um *Data Warehouse* para apoiar a tomada de decisão na coordenação do curso de Engenharia de Computação do CEFET-MG, campus Timóteo. O objetivo é proporcionar análises detalhadas e uma visão abrangente dos dados armazenados no sistema SIGAA. Esta abordagem permite que gestores e tomadores de decisão acessem informações cruciais rapidamente, contribuindo para a melhoria da eficiência e eficácia da gestão acadêmica.

Além de fornecer *insights* sobre o desempenho dos alunos e outras métricas relevantes, o trabalho destaca a importância da visualização de dados como uma ferramenta fundamental. Por meio de interfaces gráficas intuitivas e painéis de controle interativos, o projeto facilita a interpretação dos dados, permitindo aos usuários identificar tendências, padrões e problemas potenciais de maneira mais eficaz. Essas visualizações são projetadas não apenas para simplificar a compreensão dos dados complexos, mas também para habilitar uma análise mais profunda, apoiando decisões informadas para aprimorar o processo educacional e alcançar resultados superiores.

Palavras-chave: *Data Warehouse*, análise, gestão acadêmica, modelagem dimensional, tomada de decisão.

Abstract

The aim of this work is to perform dimensional modeling of a Data Warehouse to assist decision-making in the coordination of the Computer Engineering course at CEFET-MG Timóteo campus, providing analyses and a holistic view of the data stored in SIGAA, enabling managers and decision-makers to access timely information. This can help improve the efficiency and effectiveness of academic management, providing insights into student performance and other relevant metrics. With this information, it is possible to identify trends, patterns, and potential problems, and make decisions to improve the educational process and achieve better results.

Keywords: Data Warehouse, analysis, academic management, dimensional modeling, decision making.

Lista de ilustrações

Figura 1 – Arquitetura e processos de um <i>Data Warehouse</i>	14
Figura 2 – Modelagem dimensional, esquema estrela, tabelas fatos e dimensões	15
Figura 3 – Exemplo de como funciona um ambiente de <i>Business Intelligence</i>	17
Figura 4 – Toda arquitetura utilizada no sistema proposto por (SANTOSO et al., 2017) .	19
Figura 5 – Etapas utilizadas para transformação de um banco de dados operacional em um <i>Data Warehouse</i>	20
Figura 6 – Metodologia	22
Figura 7 – Modelagem Dimensional	24
Figura 8 – Tela do Aluno	29
Figura 9 – Informações Pessoais	29
Figura 10 – Grade próximo semestre	29
Figura 11 – Desempenho do Aluno	30
Figura 12 – Índices Gerais	31

Lista de abreviaturas e siglas

DW	<i>Data Warehouse</i>
BI	<i>Business Intelligence</i>
SI	Sistemas de Informação
ETL	<i>Extract, Transformation e Load</i>
OTLP	<i>Online Transactional Processing</i>
SGBD	Sistema de Gerenciamento de Banco de Dados
KPI	<i>Key Perfomance Indicator</i>
DER	Diagrama de entidade e relacionamento
ML	<i>Machine Learning</i>

Sumário

1	INTRODUÇÃO	10
1.1	Problema	11
1.2	Objetivos	12
1.2.1	Objetivo Geral	12
1.2.2	Objetivos Específicos	12
2	REFERÊNCIAL TEÓRICO	13
2.1	Data Warehouse	13
2.1.1	Características	13
2.1.2	Arquitetura	14
2.1.3	Modelagem Dimensional	15
2.1.4	Extract, Transformation and Load	16
2.2	Ambiente BI	16
3	TRABALHOS RELACIONADOS	18
4	METODOLOGIA	22
5	DESENVOLVIMENTO	23
5.1	Modelagem de Dados	23
5.2	Dicionário de Dados	24
5.3	Inserção de Dados	26
5.4	Validação e Limpeza de Dados	27
5.5	Análise Estratégica	27
5.5.1	Análise de Desempenho Disciplinar:	27
5.5.2	Média de notas por matéria:	27
5.5.3	Procedimento para Planejamento Acadêmico Futuro:	28
5.6	Visualização dos Dados	28
5.6.1	Tela do Aluno	28
5.6.2	Índices Gerais	30
6	CONCLUSÃO	32
6.1	Resultados	32
6.2	Trabalhos futuros	33
7	ANEXO I	34
	REFERÊNCIAS	46

1 Introdução

*“Não existem limites para o que você pode realizar,
exceto os limites que você impõe a si mesmo.”*

Brian Tracy

Nos últimos anos, temos vivenciado um crescimento exponencial no fluxo de dados, marcando nossa entrada na era digital ou tecnológica. Esta era é definida pelo impacto significativo das tecnologias avançadas na sociedade. Notavelmente, ela tem transformado a maneira como as empresas interagem com seus clientes. Diferentemente do passado, onde as empresas conduziam a comunicação, agora são os clientes que dialogam entre si (LEVITT, 1983). Dados são cruciais para o sucesso das empresas, melhorando sua eficiência e ajudando em decisões estratégicas. No entanto, sem tratamento adequado, dados brutos são inúteis. Por isso, é essencial que as empresas analisem e interpretem seus sistemas de informações para identificar oportunidades e se adaptar às mudanças do mercado. Nesse contexto, os sistemas de informações(SI) são essenciais para as empresas, proporcionando a compreensão de seus clientes, concorrentes e do mercado em que atuam.

A análise de dados mudou completamente a maneira como as empresas e corporações traçam seus objetivos e tomam decisões, o que antes era feito muito através do *feeling* e experiência dos gestores, agora podem ser embasadas em análise estruturais de uma grande base de dados coletados. A cultura chamada de *Data Driven* é a estratégia de transformar dados em informações úteis e importantes para o negócio. A alfabetização de dados se tornará uma questão impulsionadora e necessária para agregar valor ao negócio (WHITE; ROLLINGS, 2013).

Apesar dos benefícios proporcionados pelos sistemas de informações, os mesmos apresentam algum tipo de limitação, tais limitações estão ligadas desde a confiabilidade, segurança e qualidade dos dados utilizados, à segurança desses dados. A capacidade do usuário de interpretar os dados pode influenciar na utilização do SI. A implementação e gerenciamento de SI precisam de recursos, estratégias e *expertise* bem planejados para serem bem sucedidos (SCHULTE, 2009). As empresas precisam cada vez mais investir em ferramentas de tomada de decisão, pois essas ferramentas facilitam a identificação de tendências e *insights* importantes, dentre essas ferramentas constam as ferramentas de *Business Intelligence* (BI) além de ferramentas de inteligência artificial e aprendizagem de máquina.

Atualmente, torna-se cada vez mais essencial a utilização de ferramentas de BI. Num sentido mais abstrato, isso implica em um conjunto de técnicas que buscam organizar, filtrar, particionar e delimitar dados, possibilitando a construção de análises estruturadas. O objetivo é gerar uma visualização mais assertiva do negócio em questão. Visualizações de dados podem valorizar informações e aumentar a previsibilidade de resultados usando ferramentas de BI, que analisam históricos e reduzem riscos. Isso, por sua vez, pode aumentar a competitividade

da empresa no mercado atual. BI é uma das principais tendências tecnológicas recentes, mudando a maneira como as empresas gerenciam seus negócios e tomam decisões.

Organizações estão abandonando o trabalho manual e planilhas do Excel, migrando para ferramentas de BI. A visualização de informações para tomada de decisão deve ser simples, o que não é possível com o Excel, causando perda de produtividade e aumentando os riscos da empresa com informações limitadas ou falhas. O Power BI da Microsoft e o Tableau da Salesforce são exemplos de ferramentas que oferecem recursos de visualização de dados amplamente utilizadas em BI. Com o avanço da visualização de dados, o termo democratização de dados se popularizou entre as empresas.

Os *Data Warehouses*(DW) têm desempenhado um papel crucial nas estratégias empresariais modernas, fornecendo um repositório centralizado para dados de diversas fontes. Esta centralização facilita análises complexas e geração de relatórios, permitindo que as empresas obtenham *insights* sobre tendências de mercado, comportamento do consumidor e eficiência operacional. Além disso, os DW podem melhorar a precisão e a velocidade na tomada de decisões, pois oferecem um acesso rápido a dados históricos e atuais. Esta capacidade de integrar e analisar grandes volumes de dados de maneira eficiente é fundamental para que as empresas inovem continuamente em um ambiente de negócios cada vez mais orientado por dados.

No ambiente acadêmico existem alguns trabalhos de construção de DW que podem ser citados, entre eles estão:

1. *Data Warehouse with Big Data Technology for Higher Education* (SANTOSO et al., 2017)
2. *A Data Warehouse Design for a Typical University Information System* (BASSIL, 2012)
3. *A Holistic View of Data Warehousing in Education* (MOSCOSO-ZEA; PAREDES-GUALTOR; LUJÁN-MORA, 2018)

1.1 Problema

O SIGAA(Sistema Integrado de Gestão de Atividades Acadêmicas) é uma ferramenta de ensino utilizado pelo Centro Federal de Educação Tecnológica de Minas Gerais (CEFET-MG) campus Timóteo, utilizado para gerenciar atividades acadêmicas, dentre elas, matrículas, emissão de documentos, visualizações de atividades e notas.

O SIGAA permite os usuários emitirem relatórios e históricos de acordo com suas permissões na ferramenta, porém ela não disponibiliza serviços de BI o que dificulta muito a tomada de decisão.

O problema de pesquisa desta monografia consiste em investigar se é viável desenvolver um DW capaz de lidar com as limitações do SIGAA e o acesso aos dados dos históricos acadêmicos dos discentes. A partir desse DW, almeja-se a utilização de ferramentas de BI para gerar *insights* e análises úteis tanto para os estudantes quanto para os docentes do CEFET-MG.

A pesquisa se justifica pela necessidade de implementação de uma base de dados estruturada que gere *insights* relacionados ao histórico estudantil dos alunos para coordenação do curso de Engenharia de Computação do CEFET-MG, com o intuito de aprimorar o processo educacional, fornecer análises que antes não se era possível, e buscar novas estratégias para melhorar o desempenho dos alunos.

1.2 Objetivos

1.2.1 Objetivo Geral

O presente trabalho tem como objetivo principal realizar a modelagem de dados de um DW para auxiliar a tomada de decisão na coordenação do curso de Engenharia de Computação do CEFET-MG campus Timóteo.

1.2.2 Objetivos Específicos

1. Identificar os desafios enfrentados, pela coordenação do curso, na gestão e planejamento das atividades do curso de Engenharia de Computação do campus de Timóteo;
2. Analisar os dados disponíveis no histórico estudantil e definir os melhores indicadores para auxiliar nas tomadas de decisões da coordenação;
3. Criar um protótipo de visualização de dados para o DW;
4. Avaliar a modelagem do banco de dados através dos indicadores gerados com os dados dos históricos dos alunos;

2 Referencial Teórico

“Se queres prever o futuro, estuda o passado.”.

Confúcio

Neste capítulo, são apresentados os principais conceitos relacionados à modelagem de dados. Inicialmente, são abordados conceitos essenciais para a modelagem de dados, como o *Data Warehouse*, os quais permeiam o ambiente de *Business Intelligence*. Por fim, são discutidos os processos de *Extract, Transformation and Load* (ETL) e os cubos analíticos OLAP (*Online Analytical Processing*).

2.1 Data Warehouse

Nos anos 80 surgiu um conceito conhecido como *Data Warehouse*, esse sistema ganhou força nos anos 90 com o crescimento dos sistemas de informações empresariais, e necessidade de uma análise de dados mais sofisticada, o DW por sua capacidade de sintetizar grandes volumes de dados em qualquer formato e poder transformar-los em dados estruturados fez com que se tornasse uma das principais ferramentas para o mercado de BI. O DW é essencial para a modelagem de um sistema de dados sólido e conciso, onde é possível unificar diferentes tipos de dados, de diferentes sistemas em uma única base e utilizar-se disso para gerar todos os tipos de *insights*, além de deixar o ambiente pronto para atividades como *Data Mining*.

2.1.1 Características

Um DW é focado em um tema específico, por exemplo se uma organização possui dados de diferentes áreas, por exemplo vendas, marketing, financeiro, operacional e entre outros, não é possível criar um DW que filtre todas essas áreas, é necessário um para cada área, afinal o foco é na análise de dado, ou seja, BI, auxiliando sempre na tomada de decisão. Um DW é um repositório de dados extraído de diversas fontes, que armazena as informações por muito tempo, garantindo um histórico de dados (KORTH; SILBERSCHATZ; SUDARSHAN, 2012). Um DW são banco de dados que armazenam dados analíticos diferentemente dos bancos de dados transacionais, com o objetivo de apoiar na tomada de decisão (NAVATHE RAMEZ ELMASRI; SYSTEMS, 2019). Dentro das características de um DW as principais são a integridade de dados apresentados, um DW é uma coleção integrada de dados de várias fontes, é orientado a tempo, ou seja, o DW é uma coleção histórica de dados, onde comumente é ordenado pelo tempo. Segundo Wang (2002) o DW não é volátil, ou seja, os dados que são armazenados não são mutáveis, não sendo atualizados e nem excluídos, presta suporte à análise, portanto um DW é projetado para suportar análises complexas e pesquisas, incluindo agregações, seleções, projeções e junções. O DW é independente, ele funciona sem uma

dependência de ferramenta específica, porém permite que várias ferramentas acesse e utilize os dados (IMHOFF; GALEMMO; GEIGER, 1998).

O DW é uma tecnologia importante responsável por armazenar e gerenciar uma quantidade massiva de dados tanto para análises quanto para relatórios. Ele é a principal fonte de dados para ferramentas de BI e *Big Data*. Atualmente os DW's são cada vez mais utilizados para lidar com a crescente quantidade de dados das empresas, incluindo o *Big Data*, pois ele é essencial na captura, armazenamento e gerenciamento de dados para análise (LINTHICUM, 2013).

2.1.2 Arquitetura

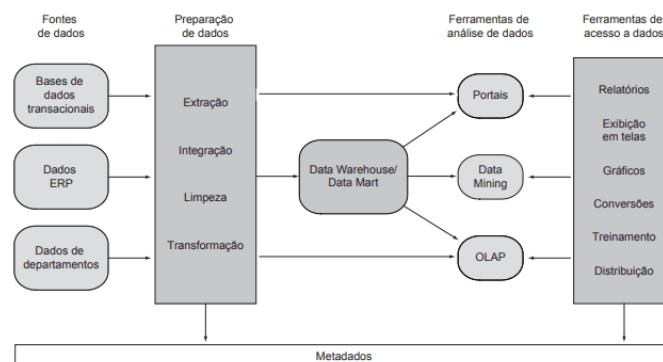
Uma das principais características de um DW é sua arquitetura, que é projetada para garantir que os dados sejam coletados, armazenados e acessíveis de forma eficiente e confiável. A arquitetura de um DW é composta por três camadas principais: a camada de entrada, a camada intermediária e a camada de saída, cada uma dessas camadas tem suas próprias funcionalidades e características.

A camada de entrada inclui ferramentas de ETL e é responsável por capturar dados de fontes externas, limpar e transformar esses dados para serem armazenados no DW (INMON, 2005).

A camada intermediária é responsável por armazenar os dados transformados e gerenciar o acesso a esses dados. Essa camada inclui o banco de dados do DW, que é projetado para suportar análises complexas e consultas, além de garantir a integridade dos dados (INMON, 2005).

A camada de saída é responsável por fornecer acesso aos dados armazenados no DW para usuários finais e aplicativos. Esta camada inclui ferramentas de análise e relatórios, bem como dashboards e outras ferramentas de visualização de dados que permitem que usuários finais acessem e analisem os dados armazenados no DW (KIMBALL; ROSS, 2013).

Figura 1 – Arquitetura e processos de um *Data Warehouse*.

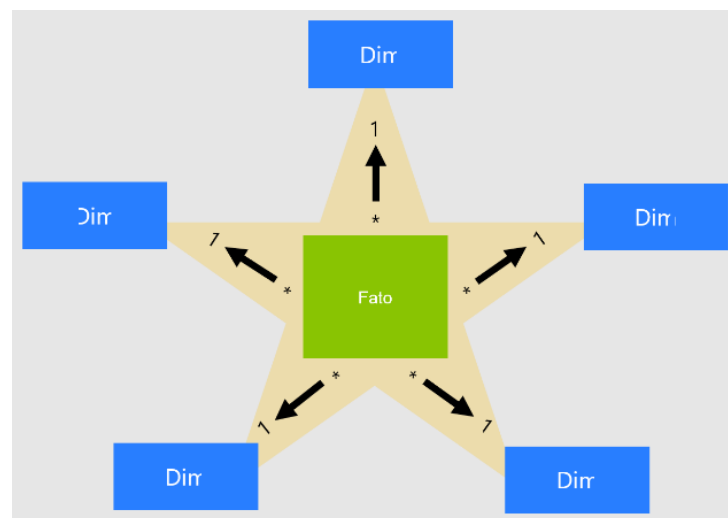


Fonte: (CARVALHO; FREITAS, 2020)

2.1.3 Modelagem Dimensional

A modelagem dimensional é uma técnica utilizada na construção de DW para representar os dados de forma fácil de entender e analisar. Ela é baseada em uma estrutura de tabelas chamada esquema estrela que está representado pela Figura 2, que consiste em uma tabela central chamada fato, que armazena os dados medidos, e várias tabelas de dimensão, que descrevem os dados.

Figura 2 – Demonstração do funcionamento de um esquema estrela com as tabelas dimensões e fato.



Fonte: (MICROSOFT, acessado em 05 de março de 2023)

A tabela de fato contém os dados medidos, como vendas, quantidades vendidas e preços, e é relacionada com várias tabelas de dimensão, como data, produto, cliente e local. Cada tabela de dimensão contém informações descritivas sobre um aspecto dos dados, como a data de uma venda, o nome do produto vendido e o local da venda. Essas tabelas de dimensão são relacionadas com a tabela de fato através de chaves estrangeiras.

A modelagem dimensional permite que os usuários façam perguntas sobre os dados de forma simples e intuitiva, como "Quantas unidades de produto A foram vendidas em janeiro de 2020?" ou "Qual foi o total de vendas em todas as lojas em abril?". Isso é possível porque os dados estão organizados de forma hierárquica e agregáveis.

A modelagem dimensional é uma técnica poderosa que permite representar os dados de forma fácil de entender e analisar, tornando possível que os usuários realizem perguntas sobre os dados de forma simples e intuitiva (KIMBALL; ROSS, 2013). A modelagem dimensional também permite que os dados sejam agrupados em diferentes níveis de granularidade, o que é importante para análises em diferentes níveis de detalhamento.

Os sistemas de gerenciamento de banco de dados relacionais são amplamente utilizados para armazenar e gerenciar os dados em um DW. Esses sistemas permitem que os dados sejam organizados em tabelas relacionais e suportam consultas SQL para manipular e analisar os dados. O gerenciamento de banco de dados relacional é a base fundamental para

o armazenamento e recuperação de dados em um DW (INMON, 2005).

OTLP (*Online Transactional Processing*) é um tipo de sistema de banco de dados projetado para suportar transações em tempo real, como inclusão, alteração e exclusão de dados. Ele é amplamente utilizado para dar suporte a aplicações críticas, como sistemas de comércio eletrônico e sistemas de gestão de recursos humanos. OTLP é uma tecnologia de banco de dados que fornece suporte a transações em tempo real, garantindo a consistência e a integridade dos dados, além de assegurar a disponibilidade para usuários autorizados (GRAY; REUTER, 1993).

2.1.4 Extract, Transformation and Load

O processo ETL é um processo crítico para a construção e manutenção de um DW. Ele consiste em três etapas principais:

Durante a etapa de extração, os dados são coletados de diversas fontes, incluindo sistemas transacionais, arquivos, planilhas eletrônicas, entre outros. A etapa de extração é a primeira etapa no processo ETL e é responsável por coletar e normalizar os dados para que se adequem ao modelo de dados do DW (KIMBALL, 1996).

Durante a etapa de transformação, os dados coletados na etapa anterior são convertidos e processados para que possam ser integrados ao modelo de dados do DW. Isso envolve tarefas como limpeza de dados, conversão de tipos de dados, agregação de informações, entre outras. A etapa de transformação é responsável por preparar os dados para serem carregados no DW, garantindo a qualidade e a consistência dos dados, o que é fundamental para a confiabilidade das análises realizadas a partir desses dados (KIMBALL, 1996).

A etapa de carregamento é a última etapa do processo ETL, na qual os dados são inseridos no banco de dados do DW. Isso envolve tarefas como o carregamento dos dados nas tabelas relacionais, bem como a criação de índices para melhorar o desempenho das consultas realizadas posteriormente. A etapa de carregamento é responsável por garantir a integridade e a disponibilidade dos dados no DW, tornando-os acessíveis para os usuários finais (KIMBALL, 1996). Essa é uma etapa crítica para o sucesso do projeto de DW, pois quaisquer problemas nesse processo podem afetar a qualidade e a confiabilidade das análises realizadas a partir desses dados.

2.2 Ambiente BI

O ambiente de BI é uma estratégia e conjunto de ferramentas utilizadas para coletar, integrar, analisar e fornecer acesso a informações relevantes aos usuários empresariais. Ele é projetado para ajudar as empresas a tomar decisões melhores e mais informadas, permitindo que elas aproveitem melhor seus dados e recursos.

O BI é uma estratégia e conjunto de processos, arquiteturas, tecnologias e ferramentas que transformam dados brutos em informações valiosas e acessíveis para a tomada de decisão (CHAUDHURI; DAYAL, 1997). Ele inclui várias ramificações, incluindo mineração de dados, dashboards e relatórios.

A mineração de dados é a técnica de descobrir padrões e conhecimento a partir de grandes conjuntos de dados. Ele é usado para identificar tendências, relações e outras informações valiosas que podem ser usadas para melhorar as decisões empresariais. a mineração de dados é a descoberta automatizada de conhecimento a partir de grandes conjuntos de dados (HAN; KAMBER, 2006).

Dashboards e relatórios são ferramentas visuais usadas para exibir informações em um formato fácil de entender. Eles permitem que os usuários visualizem informações importantes, como métricas-chave e tendências, em um único local. *Dashboards* e relatórios são ferramentas visuais que ajudam os usuários a entender e interpretar os dados, permitindo que eles tomem decisões informadas (DUBÉ; PARÉ; SICOTTE, 2007).

Em resumo, o ambiente BI é uma estratégia e conjunto de ferramentas utilizadas para coletar, integrar, analisar e fornecer acesso a informações relevantes aos usuários empresariais, todos esses processos estão demonstrados na Figura 3.

Figura 3 – O funcionamento de um ambiente de *Business Intelligence* (BI).



Fonte: (RABELLO, acessado em 05 de março de 2023)

3 Trabalhos Relacionados

*“O conhecimento é poder, mas a informação
é a chave para desbloqueá-lo.”.*
Lewis Platt

Santoso et al. (2017) exploram a aplicação de tecnologias de *Big Data* em conjunto com a técnica de *Data Warehouse* no contexto da educação superior.

O objetivo principal do estudo é demonstrar como a combinação de técnicas de DW e *Big Data* pode ajudar as instituições de ensino superior a gerenciar e analisar grandes volumes de dados, fornecendo *insights* valiosos para a tomada de decisões estratégicas. Para atingir esse objetivo, os autores propõem uma arquitetura de DW com tecnologias de *Big Data* que pode ser implementada em ambientes educacionais.

Para a realização desse trabalho, foi utilizada uma abordagem de desenvolvimento em fases, que consiste em: planejamento, análise, design, implementação e teste. Na fase de análise, foram coletados e analisados os dados de várias fontes, incluindo as transações dos sistemas de gerenciamento acadêmico, pesquisas e outras fontes relevantes. Na fase de design, foi criado um modelo dimensional para o DW, que consiste em várias dimensões, como tempo, curso, estudante e professor.

Para realizar a análise de métricas do DW, foram utilizados *Key Performance Indicator* (KPIs) para monitorar o desempenho dos estudantes, professores e cursos. Essas métricas incluem taxas de conclusão do curso, taxas de desistência, taxa de aproveitamento de créditos, média de notas dos estudantes e outras métricas relevantes.

Além disso, foram utilizadas ferramentas de análise de dados para visualizar e explorar os dados armazenados no DW, incluindo tabelas dinâmicas, gráficos e *dashboards* interativos. Essas ferramentas permitem que os usuários realizem análises mais detalhadas e tomem decisões informadas com base nos dados armazenados no DW, o processo completo desde a etapa de ETL até a parte de análise através de ferramentas de inteligência de negócios é demonstrado pela Figura 4.

O estudo também aborda alguns dos desafios que as instituições de ensino superior enfrentam ao lidar com grandes volumes de dados, tais como a falta de padronização e qualidade dos dados, e apresenta algumas soluções para superar esses desafios.

Figura 4 – Foi apresentada a arquitetura do sistema proposto, todos os estágios até a visualização dos dados através de *dashboards*.

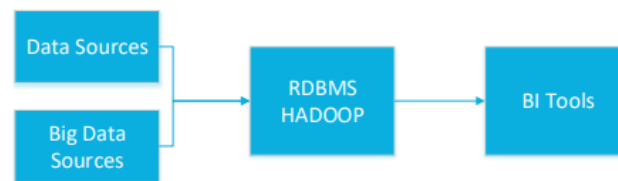


Fig. 1. The proposed system.

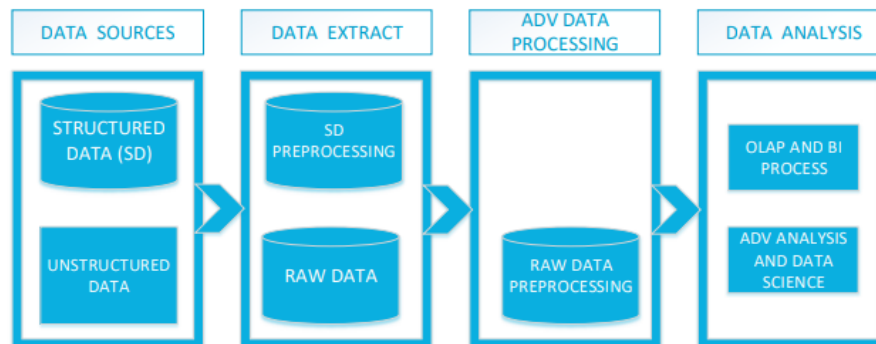


Fig. 2. The architecture of system.

Fonte: (SANTOSO et al., 2017)

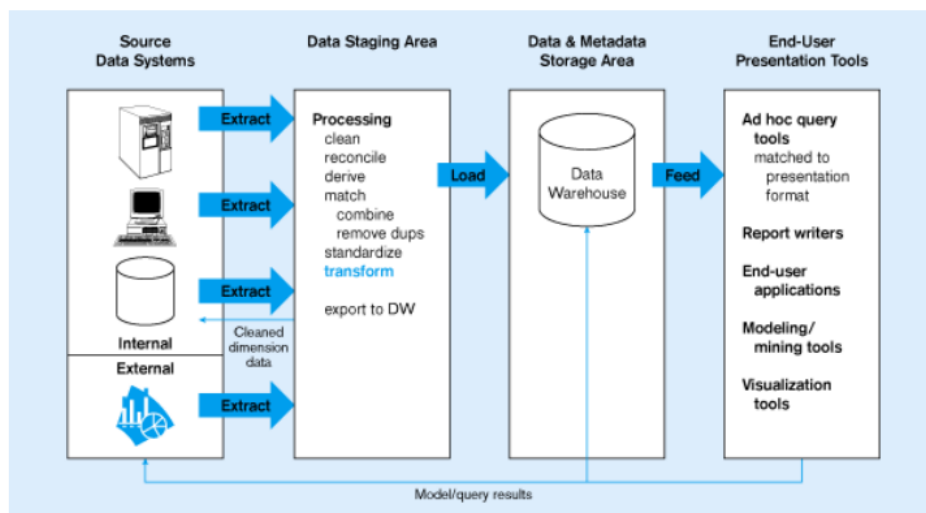
Bassil (2012) apresenta uma proposta de *design* de DW para um sistema de informação típico de uma universidade.

O projeto foi desenvolvido seguindo a metodologia de Kimball, que inclui as etapas de definição de requisitos, modelagem dimensional, projeto físico e implementação. A primeira etapa envolveu a identificação das necessidades de informação dos usuários e a definição das dimensões e fatos relevantes para o modelo dimensional.

A etapa de modelagem dimensional envolveu a criação de um modelo estrela que refletisse a estrutura dos dados do sistema de informação da universidade. Foram criadas dimensões para os diferentes aspectos da gestão acadêmica e administrativa, como aluno, professor, curso, disciplina, matrícula, entre outros. Os fatos foram definidos como medidas quantificáveis que seriam usadas para análises posteriores, como notas, créditos, faltas, entre outros.

Na etapa de projeto físico, foi definido como os dados seriam armazenados e organizados no banco de dados do DW. Foram criadas tabelas para as dimensões e fatos, e os relacionamentos entre elas foram estabelecidos.

Figura 5 – A figura retrata as diferentes etapas necessárias para transformar o banco de dados operacional em um *Data Warehouse*.



Fonte: (BASSIL, 2012)

Por fim, a implementação envolveu a carga de dados no DW, a criação de índices para melhorar o desempenho das consultas e a criação de relatórios e análises que permitiram que os usuários acessassem as informações de forma fácil e intuitiva, todo o processo para chegar nessa etapa de suporte a análise é exemplificado pela Figura 5.

As métricas de análise foram definidas a partir das necessidades dos usuários e incluíram análises de desempenho acadêmico dos alunos, taxas de matrícula, desempenho dos professores, entre outros. As métricas foram definidas em conjunto com os usuários e foram usadas para orientar o desenvolvimento do projeto e garantir que as informações fossem úteis e relevantes para a tomada de decisão.

Moscoso-Zea, Paredes-Gualtor e Luján-Mora (2018) apresentam uma visão holística sobre a aplicação de *Data Warehousing* na área de educação.

A estrutura conceitual proposta pelos autores inclui a identificação das fontes de dados, a transformação e limpeza de dados, a modelagem de dados e a implementação do DW. Além disso, a estrutura conceitual inclui a análise de dados, a visualização de dados e a tomada de decisão baseada em dados.

Os autores também apresentam um estudo de caso em que aplicam a estrutura conceitual proposta em uma universidade, com o objetivo de monitorar o desempenho acadêmico dos alunos. O estudo de caso inclui a descrição da fonte de dados, o processo de transformação e limpeza de dados, a modelagem de dados e a implementação do DW. Além disso, o estudo de caso inclui exemplos de análises de dados e visualizações que foram geradas a partir do DW.

As métricas de análise utilizadas no estudo foram a frequência de publicações em cada ano, o número de citações em cada publicação e a análise de conteúdo das publicações selecionadas. A frequência de publicações e o número de citações foram utilizados para determinar

a evolução do uso de *Data Warehousing* na educação e identificar as áreas de maior interesse dos pesquisadores. Já a análise de conteúdo foi usada para identificar as principais características dos DW aplicados na educação, bem como as principais contribuições e desafios encontrados pelos pesquisadores.

As análises dos trabalhos relacionados indicam que a aplicação de *Data Warehousing* na educação tem sido cada vez mais explorada, com um aumento significativo no número de publicações e citações nos últimos anos. As principais áreas de aplicação foram a administração de escolas e o ensino e aprendizagem, e os principais benefícios relatados foram a melhoria do gerenciamento de dados educacionais, a tomada de decisões mais informadas e a personalização do ensino. No entanto, foram identificados desafios como a integração de dados de fontes diversas, a manutenção de um alto nível de segurança e privacidade dos dados, e a necessidade de desenvolver habilidades em *Data Warehousing* e análise de dados entre os profissionais da educação.

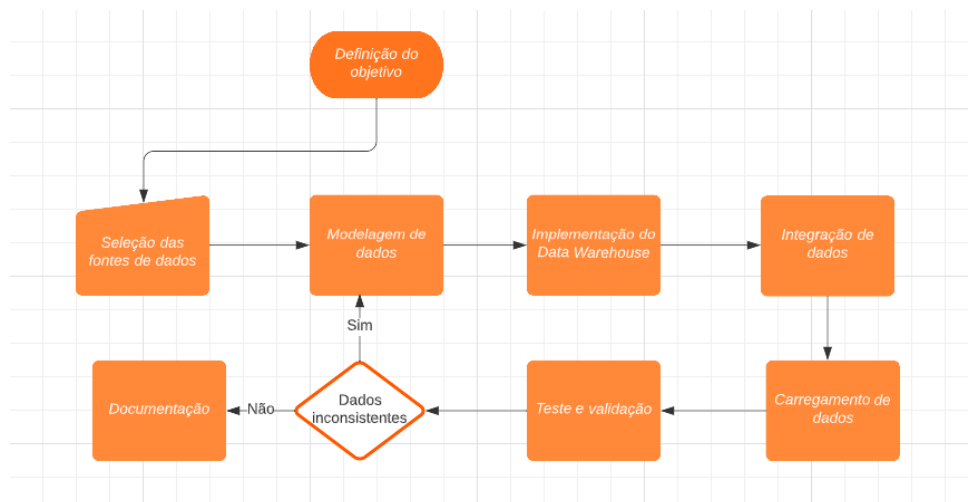
4 Metodologia

“Um bom começo é a metade”.
Aristóteles

A presente pesquisa é descritiva e exploratória do ponto de vista dos objetivos; aplicada do ponto de vista de sua natureza; qualitativa e quantitativa quanto à abordagem ao problema; e pode ser classificada como pesquisa experimental na perspectiva dos procedimentos técnicos, embora tenha mobilizado diferentes procedimentos técnicos em diferentes partes, que merecem classificação diferenciada.

Os procedimentos metodológicos são organizados nas etapas conforme a Figura 6.

Figura 6 – Metodologia.



Fonte: O próprio autor.

1. **Definição do objetivo:** Definido o que será realizado no projeto;
2. **Seleção das fontes de dados:** Definido os dados as serem analisados;
3. **Modelagem de dados:** Estruturação do esquema estrela do DW
4. **Implementação do Data Warehouse:** Estruturação das interações entre os diferentes tipos de dados dentro do DW;
5. **Integração de dados:** Implementação do algoritmo para inserir os dados;
6. **Carregamento de dados:** Inserção dos dados dentro do DW e criação da parte de visualização dos dados;
7. **Teste e validação:** Validação e correção de inconsistências dos dados;
8. **Documentação:** Documentação do projeto;

5 Desenvolvimento

Para desenvolvimento desse trabalho e para a criação e modelagem da nossa base de dados, primeiro foi necessário escolher um SGBD (sistema de gerenciamento de banco de dados relacional), o escolhido foi o SQL Server desenvolvido pela Microsoft.

Existem várias boas razões para utilizar o SQL Server, em primeiro lugar, o SQL Server oferece um ambiente robusto e seguro para o gerenciamento de dados. Ele apresenta uma grande variedade de ferramentas de gerenciamento de dados e capacidades de segurança integradas, incluindo autenticação baseada em função, criptografia de dados, auditoria de acesso a dados e mais.

Outro ponto forte é a integração fácil com outros produtos Microsoft. Se você está usando outras ferramentas ou tecnologias da Microsoft em seu projeto, como o Power BI, você descobrirá que o SQL Server se integra perfeitamente a esses produtos.

Finalmente, a comunidade de desenvolvedores do SQL Server é vasta e ativa. Existem muitos recursos disponíveis on-line, incluindo documentação, tutoriais e fóruns de discussão, que podem ajudar a resolver problemas e aprender novas técnicas.

Em resumo, o SQL Server é uma opção versátil e poderosa para a modelagem e criação de uma base de dados, oferecendo robustez, segurança, alto desempenho e uma grande comunidade de suporte.

5.1 Modelagem de Dados

O esquema utilizado para realizar a modelagem dos dados foi o esquema estrela, foi criado algumas tabelas dimensões sendo elas **dimAluno**, **dimTempo**, **dimMateria**, **textDimStatus**, **DimProximoSemestre**, uma tabela fato principal, sendo ela **fatoHistoricoEstudantil**, e algumas tabelas fatos secundárias, que serviram para facilitar o desenvolvimento da parte de visualização de dados, podemos ver o diagrama de entidade e relacionamento (DER) do nosso banco de dados na Figura 7.

ficar exclusivamente cada matéria), **NOME_MATERIA** (o nome da matéria), **TIPO_MATERIA** (o semestre que a matéria é ofertada), e **HORA_AULA** (a quantidade de horas por aula).

3. **Tabela Dimensão - TEMPO (*dimTempo*):** Esta tabela contém detalhes sobre o tempo cronológico no curso. As colunas incluem **ID_TEMPO** (uma chave primária autoincremental para identificar exclusivamente cada tempo), **DATA** (um agregado do ano e o semestre), **ANO** (O ano do curso), e **SEMESTRE** (se é semestre ímpar ou par).
4. **Tabela Fato - Histórico Estudantil (FatoHISTORICO_ESTUDANTIL):** Essa é a tabela Fato central em nosso Star Schema. Ela contém detalhes sobre o histórico acadêmico de cada aluno. As colunas incluem **ID_FATO** (uma chave primária autoincremental para identificar exclusivamente cada registro), **ID_ALUNO**, **ID_MATERIA** e **ID_TEMPO** (que são chaves estrangeiras referenciando respectivamente as tabelas **dimAluno**, **dimMateria** e **dimTempo**), **HORAS_REALIZADAS** (o número total de horas que o aluno completou), **NOTA_MATERIA** (a nota que o aluno recebeu na matéria), e **STATUS_MATERIA** (o status da matéria, por exemplo, concluída, incompleta, etc.).
5. **Tabela Dimensão - Status (textitDimStatus):** Esta tabela contém os possíveis status que uma matéria pode ter. As colunas incluem **ID_STATUS** (uma chave primária autoincremental para identificar exclusivamente cada status) e **STATUS** (o nome do status).
6. **Tabela Dimensão - ProximoSemestre (DimProximoSemestre):** Esta tabela contém informações sobre a grade curricular do semestre seguinte ao atual. As colunas incluem **ID_HORARIO** (uma chave primária autoincremental para identificar exclusivamente cada registro), **ID_MATERIA** (que é uma chave estrangeira referenciando a tabela **dimMateria**), **TIPO_MATERIA** (o semestre que a matéria é ofertada), **HORARIO** (o horário que a matéria é ofertada) e **ANO** (o semestre que essa grade se refere).
7. **Tabela Fato - Materias_Realizadas (Fato_Materias_Realizadas):** Esta tabela contém informações sobre as tentativas de cada aluno em cada matéria. As colunas incluem **ID** (uma chave primária autoincremental para identificar exclusivamente cada registro), **ID_ALUNO** (que é uma chave estrangeira referenciando a tabela **dimAluno**), **NOME_ALUNO** (nome do aluno), **ID_MATERIA** (que é uma chave estrangeira referenciando a tabela **dimMateria**), **NOME_MATERIA** (nome da matéria), **DATA** (um agregado do ano e o semestre), **NOTA_MATERIA** (a nota que o aluno recebeu na matéria), e **STATUS_MATERIA** (o status da matéria, por exemplo, concluída, incompleta, etc.), **QUANTIDADE_TENTATIVAS** (A quantidade de tentativas que o aluno já teve naquela matéria).
8. **Tabela Fato - Reprovacoes (Fato_Reprovacoes):** Esta tabela contém informações sobre a quantidade de reprovações por matéria. As colunas incluem **NOME_MATERIA** (o nome da matéria), **QTD** (quantos alunos já foram reprovados nessa matéria).
9. **Tabela Fato - Novo_Semestre (Fato_Novo_Semestre):** Esta tabela contém informações sobre as matérias que os alunos cursaram no próximo semestre. As colunas incluem **ID** (uma chave primária autoincremental para identificar exclusivamente cada registro), **ID_ALUNO** (que é uma chave estrangeira referenciando a tabela **dimAluno**),

NOME_ALUNO (nome do aluno), **ID_MATERIA** (que é uma chave estrangeira referenciando a tabela **dimMateria**), **NOME_MATERIA** (nome da matéria), **DATA** (um agregado do ano e o semestre), **TIPO_MATERIA** (o semestre que a matéria é ofertada), **HORARIO** (horário que a matéria é ofertada).

10. **Tabela Fato - Semestre_Aluno (*Fato_Semestre_Aluno*)**: Esta tabela contém informações sobre quais semestres o aluno já se matriculou no curso. As colunas incluem **ID** (uma chave primária autoincremental para identificar exclusivamente cada registro) **NOME_ALUNO** (nome do aluno), **ID_ALUNO** (que é uma chave estrangeira referenciando a tabela **dimAluno**), **MATRICULA_ALUNO** (a matrícula do aluno) e **ANO_INICIO** (o ano que o aluno iniciou no curso) e **DATA** (um agregado do ano e o semestre).
11. **Tabela Dimensão - Materias_Feitas (*Fato_Materias_Feitas*)**: Esta tabela contém da quantidade de alunos já aprovaram em determinada matéria. As colunas incluem **NOME_MATERIA** (o nome da matéria), **QTD** (quantos alunos já foram aprovados nessa matéria).

5.3 Inserção de Dados

A fim de popular as tabelas criadas em nosso DW, optamos por implementar um script em Python para gerar e inserir dados fictícios. Embora os dados sejam sintéticos, eles foram cuidadosamente projetados para respeitar as regras de negócios e as restrições de nossa instituição, tornando-os realistas e úteis para fins de demonstração e teste.

Dadas as restrições impostas pelas regras do negócio, a geração dos dados é um processo complexo e sequencial. Para garantir a consistência e a validade dos dados, nosso script Python respeita as seguintes regras:

1. **Geração de Alunos (*dimAluno*)**: Cada aluno tem um ano de entrada atribuído aleatoriamente dentro de um intervalo razoável (por exemplo, de 2009 a 2023). Este dado direcionará as restrições de quais matérias o aluno pode estar cursando em um determinado ano.
2. **Geração de Histórico Estudantil (*FatoHistorico_Estudantil*)**: Ao preencher a tabela de histórico estudantil, nosso script Python considera o ano de entrada do aluno. Um aluno que entrou na instituição em 2017 não pode estar cursando uma matéria do nono período em 2018, respeitando a progressão lógica do curso.

Ao longo deste processo, os dados são inseridos nas tabelas do nosso DW usando consultas SQL padrão. A integração do Python com SQL permite que geremos e insiramos esses dados de maneira eficiente e automatizada.

A abordagem de geração de dados permite que validemos a estrutura e a funcionalidade do nosso DW sem a necessidade de dados reais, que não estão disponíveis para nós. No entanto, acredita-se que os dados gerados sejam suficientemente realistas para permitir testes e demonstrações significativas.

5.4 Validação e Limpeza de Dados

Com o objetivo de manter um repositório de dados íntegro e robusto, que servirá para análises confiáveis e *insights* precisos que a coordenação do curso pode empregar estrategicamente, foi utilizado um processo abrangente de limpeza e validação de dados, projetado para identificar, analisar e erradicar incongruências, duplicidades e anomalias.

Primeiro se fez necessário identificar as desordens no conjunto de dados. Isso incluiu alunos com múltiplas aprovações para uma única matéria, instâncias paradoxais de alunos reprovados após aprovação na mesma disciplina, registros de alunos que supostamente cursaram matérias antes de seu ingresso oficial e casos de alunos em estágios iniciais de seu curso com matérias avançadas já realizadas. Utilizamos consultas SQL e análise de dados para isolar esses casos, fornecendo uma base sólida para as etapas de correções.

Foi identificado que a natureza do problema é o algoritmo de inserção de dados. Posteriormente, adotamos uma abordagem para reconciliar essas discrepâncias, empregando scripts SQL para eliminar duplicidades e anomalias, preservando a integridade da trajetória acadêmica de cada aluno.

Com isso a base de dados foi refinada para servir como um recurso confiável e autoritativo, capacitando a coordenação do curso com informações precisas para decisões pedagógicas fundamentadas e planejamento acadêmico estratégico.

5.5 Análise Estratégica

Através da modelagem de dados cuidadosamente estruturada, este trabalho facilita uma série de consultas analíticas, permitindo à coordenação do curso extrair *insights* e tomar decisões baseadas em dados. Esta seção detalha as consultas implementadas e explora a profundidade dos insights que elas podem proporcionar.

5.5.1 Análise de Desempenho Disciplinar:

Foi dada especial atenção às matérias com altas taxas de aprovação e reprovação. Ao identificar essas disciplinas, a instituição ganha uma perspectiva valiosa sobre possíveis falhas ou sucessos no currículo atual.

Os dados revelam quais matérias podem exigir uma revisão pedagógica, indicando se o desafio reside na metodologia de ensino, na estrutura do curso ou em outros fatores. As matérias com alto desempenho podem ser estudadas como casos de sucesso, buscando entender quais práticas pedagógicas contribuem para resultados positivos e replicá-las em outras disciplinas.

5.5.2 Média de notas por matéria:

A consulta relacionada à média de notas serve como um termômetro para o desempenho acadêmico geral, destacando disciplinas nas quais os alunos estão tendo mais dificuldade ou facilidade.

Uma média de notas consistentemente baixa pode sinalizar a necessidade de intervenções, como tutoria adicional, ajustes no material didático entre outros. Esta análise orienta a instituição sobre onde investir recursos para elevar o desempenho estudantil e a qualidade educacional.

5.5.3 Procedimento para Planejamento Acadêmico Futuro:

Foi desenvolvido um procedimento automatizado que, com base na matrícula do aluno, sugere matérias apropriadas para o próximo semestre. Ele leva em conta a grade curricular, o progresso do aluno e evita conflitos de horário, priorizando disciplinas de períodos mais baixos.

Este procedimento é crucial para um planejamento acadêmico personalizado. Ajuda a evitar a estagnação acadêmica e assegura que os alunos estão em um caminho coerente com seus objetivos e capacidades. A coordenação, armada com essas informações, pode fornecer aconselhamento proativo, ajudando os alunos a navegar por suas opções acadêmicas com confiança.

As consultas e procedimentos detalhados acima representam uma mudança para uma instituição orientada por dados.

5.6 Visualização dos Dados

A etapa de visualização dos dados é um componente essencial para a interpretação efetiva e a tomada de decisão baseada em dados. Foi explorado como o Power BI facilita a conversão de grandes volumes de dados brutos do DW em representações visuais intuitivas e interativas, permitindo uma compreensão mais profunda e acessível das informações. Esta seção ilustra a implementação do Power BI, destacando sua capacidade de fornecer *insights* e facilitar a identificação de tendências, padrões e anomalias.

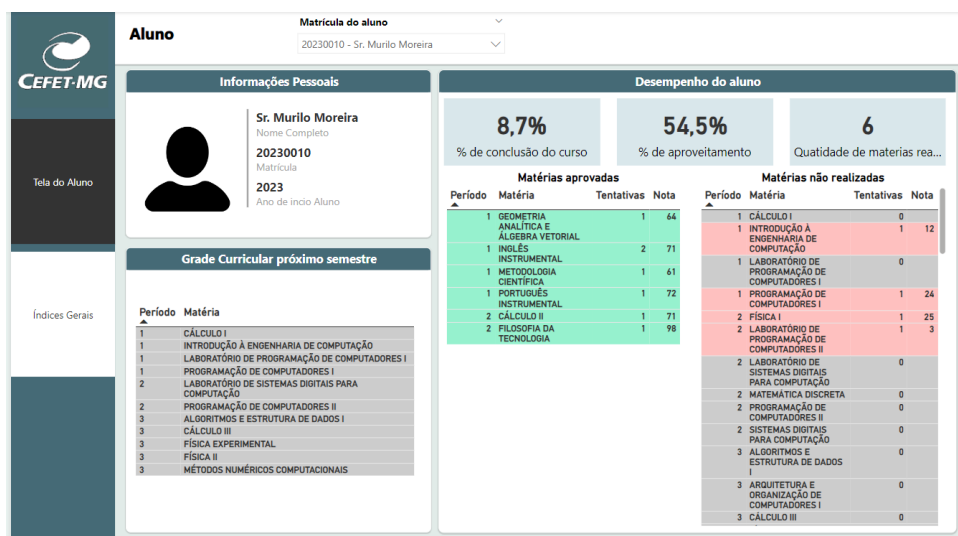
É necessário desenvolver visualizações de dados que não sejam apenas funcionais, mas também esteticamente atraentes, a fim de capturar o interesse do usuário final (CAIRO, 2012).

Foi desenvolvido dois painéis para apresentar os dados obtidos na modelagem, sendo eles a *Tela do Aluno* e *Índices Gerais*.

5.6.1 Tela do Aluno

A *Tela do Aluno* foi desenvolvida para proporcionar uma visualização personalizada dos dados de cada aluno. Inicia-se com um filtro no canto superior, que permite aos usuários selecionarem alunos pelo nome ou matrícula. Uma vez feita a seleção, a tela exibe as informações específicas do aluno escolhido como na Figura 8.

Figura 8 – Tela do Aluno.



Fonte: O próprio autor.

No canto superior esquerdo, a foto do aluno é acompanhada de detalhes como nome completo, matrícula e ano de início no curso como na Figura 9. No canto inferior esquerdo, apresenta-se a grade curricular do próximo semestre, detalhando as matérias que o aluno cursará como na Figura 10. Esta seleção de matérias é baseada no método descrito na seção anterior do projeto.



Figura 9 – Informações pessoais.

Fonte: O próprio autor.

Grade Curricular próximo semestre	
Período	Matéria
4	ALGORITMOS E ESTRUTURA DE DADOS II
4	LABORATÓRIO DE ARQUITETURA E ORGANIZAÇÃO DE COMPUTADORES II
5	PRINCÍPIOS DE COMUNICAÇÃO DE DADOS
6	LABORATÓRIO DE REDES DE COMPUTADORES I
7	CONTROLE DIGITAL DE SISTEMAS DINÂMICOS
7	LABORATÓRIO DE INTELIGÊNCIA ARTIFICIAL
7	PSICOLOGIA APLICADA ÀS ORGANIZAÇÕES
8	OTIMIZAÇÃO II

Figura 10 – Grade próximo semestre.

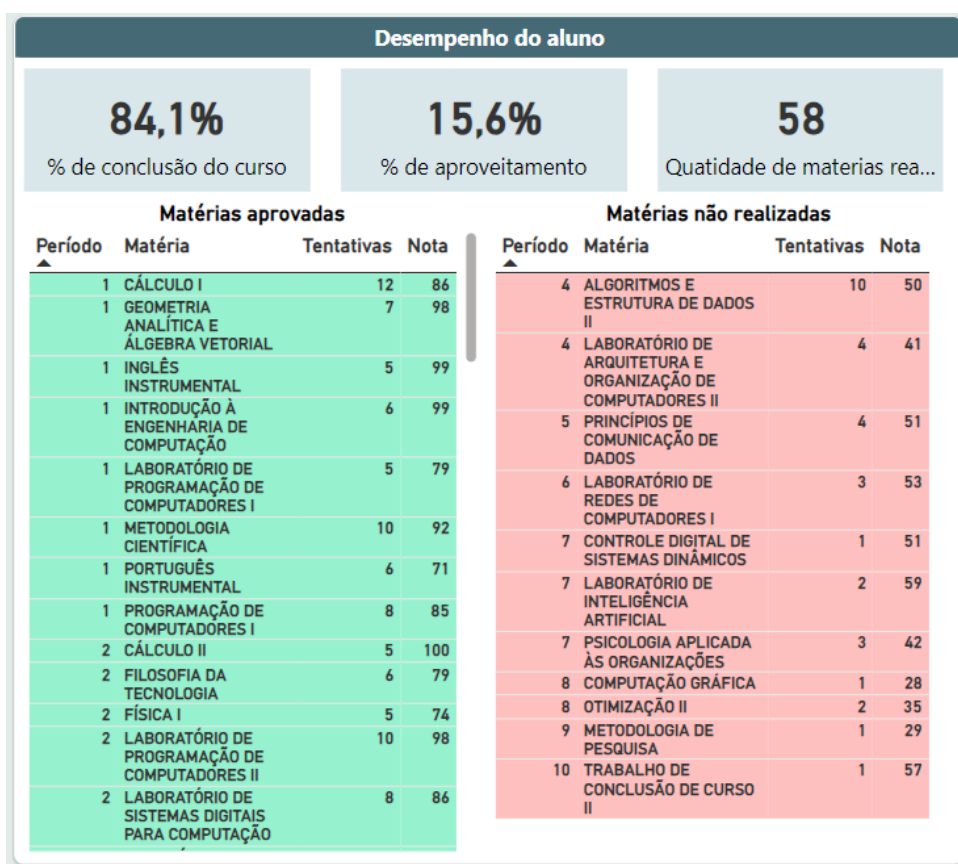
Fonte: O próprio autor.

Na parte direita da tela, encontra-se a seção "Desempenho do Aluno", que inclui três índices principais: o percentual de conclusão do curso, calculado com base na quantidade de matérias aprovadas em relação ao total do curso; o percentual de aproveitamento, derivado da quantidade de matérias aprovadas sobre o total de tentativas do aluno; e um quantitativo de matérias aprovadas.

Abaixo destes índices, duas tabelas fornecem uma visão mais aprofundada do desempenho acadêmico do aluno. A primeira tabela, à esquerda e colorida em verde, lista as

matérias em que o aluno foi aprovado, incluindo a quantidade de tentativas até a aprovação e a nota alcançada. A segunda tabela, à direita, mostra todas as matérias que o aluno ainda não concluiu, com o número de tentativas e a maior nota obtida. As linhas em vermelho indicam as matérias em que o aluno já foi reprovado pelo menos uma vez, enquanto as linhas em cinza representam aquelas que ele ainda não cursou como na Figura 11.

Figura 11 – Desempenho do Aluno.



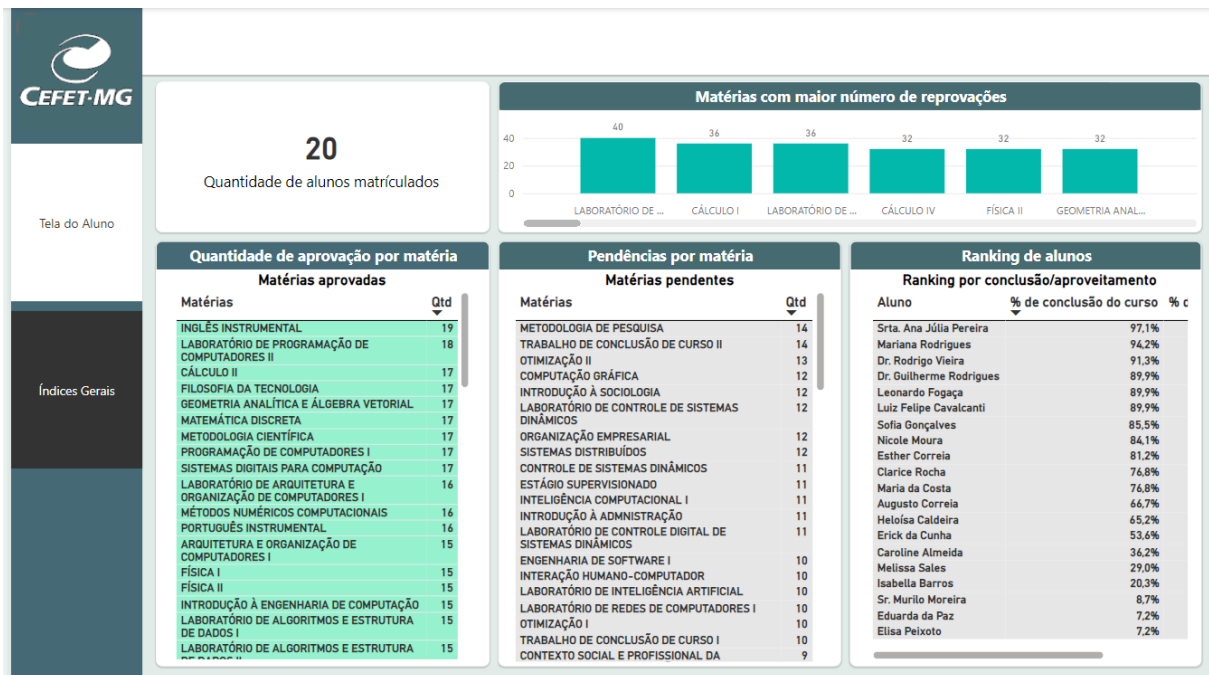
Fonte: O próprio autor.

5.6.2 Índices Gerais

O painel *Índices Gerais* é uma ferramenta analítica crucial do projeto, oferecendo uma visão abrangente do curso Figura 12.

No canto superior esquerdo, este painel exibe o número total de alunos atualmente matriculados no curso. Já no canto superior direito, destaca-se um gráfico de colunas clusterizadas que ilustra as matérias com maior número de reprovações, ordenadas de forma decrescente.

Figura 12 – Índices Gerais.



Fonte: O próprio autor.

Abaixo, o painel se divide em três seções distintas, alinhadas lado a lado. A primeira seção exibe uma tabela que funciona como um ranking das matérias com maior quantidade de alunos aprovados, ordenadas do maior para o menor número de aprovações. A seção central oferece um ranking similar, mas focado nas matérias com o maior número de alunos ainda pendentes de cursá-las. A terceira e última seção, à direita, apresenta um ranking de alunos com base nos índices da *Tela do Aluno*, especificamente os índices de percentual de aproveitamento e de conclusão do curso. Este design do painel proporciona uma análise detalhada e multifacetada do desempenho do curso e dos alunos.

6 Conclusão

*“Não contavam com a minha astúcia!”
Chapolin (Roberto Gómez Bolaños)*

Este trabalho apresentou a modelagem dimensional de um *Data Warehouse* destinado à análise de dados acadêmicos do CEFET-MG campus Timóteo, buscando melhorar a tomada de decisão na coordenação do curso de Engenharia de Computação. Através da implementação do modelo estrela e do uso de ferramentas de *Business Intelligence*, foi possível estruturar um repositório de dados, capaz de gerar *insights* sobre o desempenho dos alunos e a eficácia do processo educacional.

O desenvolvimento do DW envolveu a criação de tabelas de dimensão e fato, além da inserção de dados simulados, respeitando as regras e limitações institucionais. A abordagem adotada para a inserção dos dados mostrou-se eficiente, possibilitando a simulação realista de cenários acadêmicos e proporcionando um ambiente para análises relevantes.

Os resultados deste trabalho demonstram o potencial do uso de DW e BI no contexto acadêmico, não apenas para melhorar a gestão do curso, mas também para enriquecer a experiência educacional dos alunos. As análises geradas podem ajudar a identificar padrões de desempenho, prever tendências acadêmicas e fornecer recomendações estratégicas para melhorias contínuas.

6.1 Resultados

O desenvolvimento do DW proporciona uma compreensão aprofundada do desempenho acadêmico dos alunos, permitindo a identificação de padrões em diversas disciplinas e semestres. Isso inclui a análise de tendências como taxas de reprovação, períodos de maior dificuldade e progressão geral dos estudantes. Essas informações são cruciais para a avaliação da eficácia do currículo atual, ajudando a determinar a adequação das matérias no preparo dos alunos para fases subsequentes e carreiras futuras, com isso podemos dizer que a modelagem dimensional realizada foi bem sucedida, pois foi possível através dela, entregar inteligência na análise dos dados.

Os dados revelam necessidades específicas dos alunos, indicando áreas que demandam reforço ou apoio educacional adicional. Com essas análises, os professores podem desenvolver estratégias de ensino personalizadas, adaptadas às necessidades e estilos de aprendizagem dos alunos, e avaliar a eficácia de diferentes metodologias pedagógicas. O *feedback* contínuo obtido através dos dados permite ajustes nas abordagens de ensino para aprimorar os resultados acadêmicos.

Além disso, o projeto pode influenciar a colaboração interdepartamental na elaboração de currículos e programas de estudo, permitindo avaliar o impacto de mudanças curriculares

no desempenho dos alunos. Com a capacidade de prever tendências acadêmicas e desempenho futuro, o DW oferece à coordenação a possibilidade de tomar decisões baseadas em dados, conduzindo a uma gestão mais eficiente e orientada a resultados. Os *insights* gerados são fundamentais para o desenvolvimento de estratégias educacionais eficazes e para a implementação de melhorias contínuas, visando a excelência acadêmica e o sucesso dos estudantes. Essa abordagem baseada em dados também permite uma alocação mais eficiente de recursos, tanto no apoio aos estudantes quanto na infraestrutura acadêmica.

6.2 Trabalhos futuros

Para trabalhos futuros, a expansão do escopo do DW é um passo natural e vital. Isso pode incluir a inclusão de mais variáveis relacionadas a diferentes aspectos acadêmicos e administrativos, como dados de participação em atividades extracurriculares, *feedback* de avaliações de disciplinas e professores, e dados relacionados ao bem-estar e engajamento dos alunos. Essa expansão permitiria uma análise mais holística do ambiente educacional, contribuindo para um entendimento mais profundo das diversas facetas que influenciam o sucesso acadêmico e o desenvolvimento dos estudantes.

A integração de técnicas de *Machine Learning* (ML) ao DW apresenta uma oportunidade significativa para enriquecer as análises preditivas e descritivas. Modelos de ML podem ser aplicados para identificar padrões complexos e prever tendências acadêmicas, como o risco de desistência de alunos, o sucesso potencial em diferentes áreas de estudo, ou o impacto de certas intervenções pedagógicas. Esses modelos podem ser treinados com dados históricos e continuamente aprimorados à medida que novos dados são coletados, aumentando a precisão e a relevância das previsões.

Um aspecto crucial na aplicação de modelos de ML é a necessidade de interpretar e comunicar as descobertas de maneira compreensível para os tomadores de decisão. Portanto, um foco na interpretabilidade e na explicabilidade dos modelos de ML será fundamental, para garantir que as análises sejam transparentes e possam ser usadas de forma efetiva na tomada de decisões.

Outra área promissora para trabalhos futuros é a integração de *feedback* em tempo real no sistema. Seria possível coletar e analisar dados sobre o engajamento dos alunos em sala de aula, eficiência de métodos de ensino, e outras variáveis dinâmicas. Essa abordagem ofereceria aos educadores a capacidade de fazer ajustes imediatos em suas estratégias pedagógicas, baseando-se em *feedback* quase instantâneo.

Estas direções para trabalhos futuros não apenas aprimorariam a eficácia do sistema atual, mas também contribuiriam significativamente para a literatura sobre análise de dados no ambiente educacional.

7 Anexo I

Listing 7.1 – Criando Tabelas

```
-- Cria o da Dimens o Aluno
CREATE TABLE CEFET_PRODUCAO.DBO.DimALUNO (
    ID_ALUNO INT IDENTITY(1,1) PRIMARY KEY,
    NOME_ALUNO NVARCHAR(255) NOT NULL,
    MATRICULA_ALUNO NVARCHAR(50) NOT NULL,
    ANO_INICIO INT NOT NULL
);

-- Cria o da Dimens o Mat ria
CREATE TABLE CEFET_PRODUCAO.DBO.DimMATERIA (
    ID_MATERIA INT IDENTITY(1,1) PRIMARY KEY,
    NOME_MATERIA NVARCHAR(255) NOT NULL,
    TIPO_MATERIA NVARCHAR(50) NOT NULL,
    HORA_AULA INT NOT NULL
);

-- Cria o da Dimens o Tempo
CREATE TABLE CEFET_PRODUCAO.DBO.DimTEMPO (
    ID_TEMPO INT IDENTITY(1,1) PRIMARY KEY,
    DATA NVARCHAR(6) NOT NULL,
    ANO INT NOT NULL,
    SEMESTRE INT NOT NULL,
);

-- Cria o da Tabela de Fatos Hist rico Estudantil
CREATE TABLE FatoHISTORICO_ESTUDANTIL (
    ID_FATO INT IDENTITY(1,1) PRIMARY KEY,
    ID_ALUNO INT FOREIGN KEY REFERENCES DimAluno(ID_Aluno),
    ID_MATERIA INT FOREIGN KEY REFERENCES DimMateria(ID_Materia),
    ID_TEMPO INT FOREIGN KEY REFERENCES DimTempo(ID_Tempo),
    HORAS_REALIZADAS INT NOT NULL,
    --HORAS_PENDENTES INT NOT NULL,
    NOTA_MATERIA FLOAT,
    STATUS_MATERIA INT NOT NULL
);
```

```
-- Cria o da tabela DimStatus
```

```
CREATE TABLE DimStatus (  
    ID_STATUS INT IDENTITY(0,1) PRIMARY KEY,  
    STATUS NVARCHAR(50) NOT NULL  
);
```

```
-- Cria o da Tabela DimPROXIMOSEMESTRE
```

```
CREATE TABLE DimPROXIMOSEMESTRE (  
    ID_HORARIO INT IDENTITY(1,1) PRIMARY KEY,  
    ID_MATERIA INT FOREIGN KEY REFERENCES DimMateria(ID_Materia),  
    TIPO_MATERIA NVARCHAR(100),  
    HORARIO NVARCHAR(100),  
    ANO NVARCHAR(6)  
);
```

```
-- Cria o da Tabela FATO_MATERIAS_FEITAS
```

```
CREATE TABLE FATO_MATERIAS_FEITAS(  
    NOME_MATERIA NVARCHAR(255) NOT NULL,  
    QTD INT  
);
```

```
-- Cria o da Tabela FATO_MATERIAS_REALIZADAS
```

```
CREATE TABLE FATO_MATERIAS_REALIZADAS (  
    ID INT IDENTITY(1,1) PRIMARY KEY,  
    ID_ALUNO INT FOREIGN KEY REFERENCES DimAluno(ID_Aluno),  
    NOME_ALUNO NVARCHAR(255) NOT NULL,  
    ID_MATERIA INT FOREIGN KEY REFERENCES DimMateria(ID_Materia),  
    NOME_MATERIA NVARCHAR(255) NOT NULL,  
    DATA NVARCHAR(6),  
    NOTA_MATERIA FLOAT,  
    STATUS_MATERIA INT,  
    QUANTIDADE_TENTATIVAS INT  
);
```

```

-- Cria o da Tabela FATO_NOVO_SEMESTRE
CREATE TABLE FATO_NOVO_SEMESTRE (
    ID INT IDENTITY(1,1) PRIMARY KEY,
    ID_ALUNO INT FOREIGN KEY REFERENCES DimAluno(ID_Aluno),
    NOME_ALUNO NVARCHAR(255) NOT NULL,
    ID_MATERIA INT FOREIGN KEY REFERENCES DimMateria(ID_Materia),
    NOME_MATERIA NVARCHAR(255) NOT NULL,
    DATA NVARCHAR(6),
    TIPO_MATERIA INT NOT NULL,
    HORARIO INT NOT NULL
);

-- Cria o da Tabela FATO_REPROVACOES
CREATE TABLE FATO_REPROVACOES(
    NOME_MATERIA NVARCHAR(255) NOT NULL,
    QTD INT
);

-- Cria o da Tabela FATO_SEMESTRE_ALUNO
CREATE TABLE FATO_SEMESTRE_ALUNO (
    ID INT IDENTITY(1,1) PRIMARY KEY,
    NOME_ALUNO NVARCHAR(255) NOT NULL,
    ID_ALUNO INT FOREIGN KEY REFERENCES DimAluno(ID_Aluno),
    MATRICULA_ALUNO NVARCHAR(50) NOT NULL,
    ANO_INICIO INT NOT NULL,
    DATA NVARCHAR(6)
);

```

Listing 7.2 – Gerador.py

```

import random
import pandas as pd
import pyodbc
import datetime
import sqlalchemy

df_return = pd.DataFrame()

def connectDatabase():
    server = 'SeuBancodeDados'
    database = 'CEFET_PRODUCAO'
    username = 'SeuUsuario'
    password = 'SuaSenha'
    driver= '{ODBC_Driver_17_for_SQL_Server}'

```

```

connection_string = f'DRIVER={driver};SERVER={server};DATABASE={
    database};UID={username};PWD={password}'

conn = pyodbc.connect(connection_string)

return conn

def insereFato(cursor):

    lista = [30, 60, 90]
    status_reprovado = [0]
    status_aprovado = [1]
    for i in range(1,5000):
        materia_id = random.randint(1,69)
        aluno_id = random.randint(1,20)
        tempo_id = random.randint(1,30)
        horas_realizadas = random.choice(lista) # gera horas de estudo
            aleatorias
        nota_materia = random.randint(0, 100) # gera nota aleatoria
        if nota_materia >= 60:
            status_materia = 1
        elif nota_materia < 60:
            status_materia = 0

        sql = '''INSERT INTO CEFET_PRODUCAO.DBO.FatoHISTORICO_ESTUDANTIL
            (ID_ALUNO, ID_MATERIA, ID_TEMPO, HORAS_REALIZADAS,
            NOTA_MATERIA, STATUS_MATERIA) VALUES (?, ?, ?, ?, ?, ?)'''

        cursor.execute(sql,aluno_id,materia_id,tempo_id,horas_realizadas
            ,nota_materia, status_materia)
        cursor.commit()

from faker import Faker
fake = Faker('pt_BR') # para nomes em portuges do Brasil

def insereAluno(N, cursor):
    lista = [2009,2010,2011,2012,2013,2014,2015,
    2016,2017,2018,2019,2020,2021,2022,2023]
    for i in range(1, N+1): # substitua N pelo n mero de alunos que
        voce quer gerar
        nome = fake.name()
        ano_inicio = random.choice(lista)
        matricula = '{}00'.format(ano_inicio) + str(i)

```

```

        #print(f"INSERT INTO CEFET_PRODUCAO.DBO.DimALUNO (NOME_ALUNO,
            MATRICULA_ALUNO, ANO_INICIO) VALUES ('{nome}', '{matricula}',
            {ano_inicio});")

    sql = '''INSERT INTO CEFET_PRODUCAO.DBO.DimALUNO (NOME_ALUNO,
        MATRICULA_ALUNO, ANO_INICIO) VALUES (?, ?, ?)'''
    cursor.execute(sql,nome,matricula,ano_inicio)
    cursor.commit()

if __name__ == "__main__":
    conn = connectDatabase()
    cursor = conn.cursor()
    insereAluno(20, cursor)
    insereFato(cursor)

    conn.close()

```

Listing 7.3 – TratamentoDados.sql

```

DROP TABLE #TEMP

SELECT A.ID_FATO, D.NOME_ALUNO, D.ANO_INICIO, B.DATA, C.
    NOME_MATERIA, A.NOTA_MATERIA, A.STATUS_MATERIA, E.STATUS, C.
    HORA_AULA, C.TIPO_MATERIA,
CASE
WHEN C.TIPO_MATERIA = 1 THEN CONVERT(NVARCHAR, D.ANO_INICIO) + '
    .1'
WHEN C.TIPO_MATERIA = 2 AND SUBSTRING(B.DATA, 1, 4) - D.
    ANO_INICIO = 0 THEN SUBSTRING(B.DATA, 1, 4) + '.2'
WHEN C.TIPO_MATERIA = 2 AND SUBSTRING(B.DATA, 1, 4) - D.
    ANO_INICIO > 0 THEN B.DATA
WHEN C.TIPO_MATERIA = 2 AND SUBSTRING(B.DATA, 1, 4) - D.
    ANO_INICIO < 0 THEN CONVERT(NVARCHAR, D.ANO_INICIO) + '.2'
WHEN C.TIPO_MATERIA = 3 AND SUBSTRING(B.DATA, 1, 4) - D.
    ANO_INICIO >= 1 THEN B.DATA
WHEN C.TIPO_MATERIA = 3 AND SUBSTRING(B.DATA, 1, 4) - D.
    ANO_INICIO < 1 THEN CONVERT(NVARCHAR, D.ANO_INICIO + 1) + '.1'
WHEN C.TIPO_MATERIA = 4 AND SUBSTRING(B.DATA, 1, 4) - D.
    ANO_INICIO = 1 THEN SUBSTRING(B.DATA, 1, 4) + '.2'
WHEN C.TIPO_MATERIA = 4 AND SUBSTRING(B.DATA, 1, 4) - D.
    ANO_INICIO > 1 THEN B.DATA
WHEN C.TIPO_MATERIA = 4 AND SUBSTRING(B.DATA, 1, 4) - D.

```



```

        ANO_INICIO < 1 THEN CONVERT(NVARCHAR, D.ANO_INICIO + 1) + '.2'
WHEN C.TIPO_MATERIA = 5 AND SUBSTRING(B.DATA, 1, 4) - D.
        ANO_INICIO >= 2 THEN B.DATA
WHEN C.TIPO_MATERIA = 5 AND SUBSTRING(B.DATA, 1, 4) - D.
        ANO_INICIO < 2 THEN CONVERT(NVARCHAR, D.ANO_INICIO + 2) + '.1'
WHEN C.TIPO_MATERIA = 6 AND SUBSTRING(B.DATA, 1, 4) - D.
        ANO_INICIO = 2 THEN SUBSTRING(B.DATA, 1, 4) + '.2'
WHEN C.TIPO_MATERIA = 6 AND SUBSTRING(B.DATA, 1, 4) - D.
        ANO_INICIO > 2 THEN B.DATA
WHEN C.TIPO_MATERIA = 6 AND SUBSTRING(B.DATA, 1, 4) - D.
        ANO_INICIO < 2 THEN CONVERT(NVARCHAR, D.ANO_INICIO + 2) + '.2'
WHEN C.TIPO_MATERIA = 7 AND SUBSTRING(B.DATA, 1, 4) - D.
        ANO_INICIO >= 3 THEN B.DATA
WHEN C.TIPO_MATERIA = 7 AND SUBSTRING(B.DATA, 1, 4) - D.
        ANO_INICIO < 3 THEN CONVERT(NVARCHAR, D.ANO_INICIO + 3) + '.1'
WHEN C.TIPO_MATERIA = 8 AND SUBSTRING(B.DATA, 1, 4) - D.
        ANO_INICIO = 3 THEN SUBSTRING(B.DATA, 1, 4) + '.2'
WHEN C.TIPO_MATERIA = 8 AND SUBSTRING(B.DATA, 1, 4) - D.
        ANO_INICIO > 3 THEN B.DATA
WHEN C.TIPO_MATERIA = 8 AND SUBSTRING(B.DATA, 1, 4) - D.
        ANO_INICIO < 3 THEN CONVERT(NVARCHAR, D.ANO_INICIO + 3) + '.2'
WHEN C.TIPO_MATERIA = 9 AND SUBSTRING(B.DATA, 1, 4) - D.
        ANO_INICIO >= 4 THEN B.DATA
WHEN C.TIPO_MATERIA = 9 AND SUBSTRING(B.DATA, 1, 4) - D.
        ANO_INICIO < 4 THEN CONVERT(NVARCHAR, D.ANO_INICIO + 4) + '.1'
WHEN C.TIPO_MATERIA = 10 AND SUBSTRING(B.DATA, 1, 4) - D.
        ANO_INICIO = 4 THEN SUBSTRING(B.DATA, 1, 4) + '.2'
WHEN C.TIPO_MATERIA = 10 AND SUBSTRING(B.DATA, 1, 4) - D.
        ANO_INICIO > 4 THEN B.DATA
WHEN C.TIPO_MATERIA = 10 AND SUBSTRING(B.DATA, 1, 4) - D.
        ANO_INICIO < 4 THEN CONVERT(NVARCHAR, D.ANO_INICIO + 4) + '.2'
END AS CALCULO_ANO
INTO #TEMP
FROM CEFET_PRODUCAO.DBO.FatoHISTORICO_ESTUDANTIL (NOLOCK) A
JOIN CEFET_PRODUCAO.DBO.DimTEMPO (NOLOCK) B ON A.ID_TEMPO = B.
        ID_TEMPO
JOIN CEFET_PRODUCAO.DBO.DimMateria (NOLOCK) C ON C.ID_MATERIA = A.
        ID_MATERIA
JOIN CEFET_PRODUCAO.DBO.DimAluno (NOLOCK) D ON D.ID_ALUNO = A.
        ID_ALUNO
JOIN CEFET_PRODUCAO.DBO.DimStatus (NOLOCK) E ON E.ID_STATUS = A.
        STATUS_MATERIA
WHERE D.ID_ALUNO = 3

```

```

===== DATA DE CURSO DA
MATERIA MENOR Q A DATA DE INGRESSAO
=====

```

```

DELETE A FROM
CEFET_PRODUCAO.DBO.FatoHISTORICO_ESTUDANTIL (NOLOCK) A
JOIN CEFET_PRODUCAO.DBO.DimTEMPO (NOLOCK) B ON A.ID_TEMPO = B.
    ID_TEMPO
JOIN CEFET_PRODUCAO.DBO.DimMateria (NOLOCK) C ON C.ID_MATERIA = A
    .ID_MATERIA
JOIN CEFET_PRODUCAO.DBO.DimAluno (NOLOCK) D ON D.ID_ALUNO = A.
    ID_ALUNO
JOIN CEFET_PRODUCAO.DBO.DimStatus (NOLOCK) E ON E.ID_STATUS = A.
    STATUS_MATERIA
WHERE SUBSTRING(B.DATA, 1, 4) - D.ANO_INICIO < 0

```

```

=====
DATA DE CURSO NULL
=====

```

```

DELETE C
--SELECT *
FROM #TEMP A
LEFT JOIN DimTEMPO B ON A.CALCULO_ANO = B.DATA
JOIN FatoHISTORICO_ESTUDANTIL C ON A.ID_FATO = C.ID_FATO
WHERE B.ID_TEMPO IS NULL

```

```

===== DATA MENOR QUE A
DATA MINIMA PARA REALIZAR A MATERIA
=====

```

```

DELETE B
--SELECT A.*
FROM #TEMP A
JOIN FatoHISTORICO_ESTUDANTIL B ON A.ID_FATO = B.ID_FATO
WHERE A.DATA < CALCULO_ANO

```

```

===== FEZ A MESMA MATERIA
DUAS VEZES NA MESMA DATA
=====

```

```

--DROP TABLE #TEMP
SELECT
A.ID_FATO ,
D.ID_ALUNO ,
C.ID_MATERIA ,
D.NOME_ALUNO ,
B.DATA ,
C.NOME_MATERIA ,
A.NOTA_MATERIA ,
A.STATUS_MATERIA ,
E.STATUS ,
DUPRANK = ROW_NUMBER() OVER (PARTITION BY D.ID_ALUNO , C.
    ID_MATERIA , B.DATA ORDER BY D.ID_ALUNO , C.ID_MATERIA , B.DATA)
INTO #TEMP
FROM CEFET_PRODUCAO.DBO.FatoHISTORICO_ESTUDANTIL (NOLOCK) A
JOIN CEFET_PRODUCAO.DBO.DimTEMPO (NOLOCK) B ON A.ID_TEMPO = B.
    ID_TEMPO
JOIN CEFET_PRODUCAO.DBO.DimMateria (NOLOCK) C ON C.ID_MATERIA = A.
    ID_MATERIA
JOIN CEFET_PRODUCAO.DBO.DimAluno (NOLOCK) D ON D.ID_ALUNO = A.
    ID_ALUNO
JOIN CEFET_PRODUCAO.DBO.DimStatus (NOLOCK) E ON E.ID_STATUS = A.
    STATUS_MATERIA
ORDER BY NOME_ALUNO , C.ID_MATERIA , B.DATA

DELETE C
--SELECT *
FROM #TEMP A
JOIN FatoHISTORICO_ESTUDANTIL C ON A.ID_FATO = C.ID_FATO
WHERE A.DUPRANK > 1

===== TEM MAIS DE UMA
    APROVA O NA MESMA MAT RIA
=====

--DROP TABLE #TEMP
SELECT
A.ID_FATO ,
D.ID_ALUNO ,
C.ID_MATERIA ,
D.NOME_ALUNO ,
B.DATA ,
C.NOME_MATERIA ,

```

```

A.NOTA_MATERIA ,
A.STATUS_MATERIA ,
E.STATUS ,
DUPRANK = ROW_NUMBER() OVER (PARTITION BY D.ID_ALUNO , C.
    ID_MATERIA ORDER BY B.DATA)
INTO #TEMP
FROM CEFET_PRODUCAO.DBO.FatoHISTORICO_ESTUDANTIL (NOLOCK) A
JOIN CEFET_PRODUCAO.DBO.DimTEMPO (NOLOCK) B ON A.ID_TEMPO = B.
    ID_TEMPO
JOIN CEFET_PRODUCAO.DBO.DimMateria (NOLOCK) C ON C.ID_MATERIA = A.
    ID_MATERIA
JOIN CEFET_PRODUCAO.DBO.DimAluno (NOLOCK) D ON D.ID_ALUNO = A.
    ID_ALUNO
JOIN CEFET_PRODUCAO.DBO.DimStatus (NOLOCK) E ON E.ID_STATUS = A.
    STATUS_MATERIA
ORDER BY NOME_ALUNO , C.ID_MATERIA , B.DATA

--DROP TABLE #TEMP2
SELECT ID_ALUNO , ID_MATERIA , NOME_ALUNO , STATUS , MIN(DATA) AS
    DATA_APROVACAO -- SUPONDO QUE VOC TENHA UMA COLUNA
    DATA_REGISTRO
INTO #TEMP2
FROM #TEMP
WHERE STATUS_MATERIA = 1 -- APROVADO
GROUP BY ID_ALUNO , ID_MATERIA , NOME_ALUNO , STATUS

DELETE A
--SELECT *
FROM FatoHISTORICO_ESTUDANTIL A
JOIN DIMTEMPO C ON A.ID_TEMPO = C.ID_TEMPO
WHERE EXISTS (SELECT 1
    FROM #TEMP2 B
    WHERE A.id_aluno = B.id_aluno
    AND A.id_materia = B.id_materia
    AND C.DATA > B.data_aprovacao
    )

DELETE C
--SELECT A.*
FROM #TEMP A
JOIN FatoHISTORICO_ESTUDANTIL C ON A.ID_FATO = C.ID_FATO
LEFT JOIN #TEMP2 B ON B.ID_FATO = C.ID_FATO
WHERE A.DUPRANK > B.DUPRANK

```

Listing 7.4 – Procedure Proximo Semestre

```

USE [CEFET_PRODUCAO]
GO
/***** Object:  StoredProcedure [dbo].[PROXIMO_SEMESTRE]
    Script Date: 19/11/2023 11:40:36 *****/
SET ANSI_NULLS ON
GO
SET QUOTED_IDENTIFIER ON
GO

ALTER PROCEDURE [dbo].[PROXIMO_SEMESTRE](

@MATRICULA NVARCHAR(100)

) AS

IF OBJECT_ID('#TEMP') IS NOT NULL
    DROP TABLE #TEMP;

SELECT A.*, C.NOME_MATERIA, D.DATA, C.ID_MATERIA
INTO #TEMP
FROM DimALUNO (NOLOCK) A
JOIN FatoHISTORICO_ESTUDANTIL (NOLOCK) B ON A.ID_ALUNO = B.
    ID_ALUNO
JOIN DimMATERIA (NOLOCK) C ON B.ID_MATERIA = C.ID_MATERIA
JOIN DimTEMPO (NOLOCK) D ON D.ID_TEMPO = B.ID_TEMPO
WHERE A.MATRICULA_ALUNO = @MATRICULA
AND EXISTS (SELECT *
                                FROM DIMPROXIMOSEMESTRE (
                                    NOLOCK) X WHERE X.
                                    ID_MATERIA = C.
                                    ID_MATERIA)

AND B.STATUS_MATERIA = 1

IF OBJECT_ID('#TIPO_SEMESTRE') IS NOT NULL
    DROP TABLE #TIPO_SEMESTRE;

SELECT DISTINCT A.*, C.NOME_MATERIA, E.*,
CASE
WHEN SUBSTRING(A.ANO, 1, 4) - E.ANO_INICIO = 0 THEN 2

```

```

WHEN SUBSTRING(A.ANO, 1, 4) - E.ANO_INICIO = 1 AND SUBSTRING(A.
    ANO, 6, 2) = 1 THEN 3
WHEN SUBSTRING(A.ANO, 1, 4) - E.ANO_INICIO = 1 AND SUBSTRING(A.
    ANO, 6, 2) = 2 THEN 4
WHEN SUBSTRING(A.ANO, 1, 4) - E.ANO_INICIO = 2 AND SUBSTRING(A.
    ANO, 6, 2) = 1 THEN 5
WHEN SUBSTRING(A.ANO, 1, 4) - E.ANO_INICIO = 2 AND SUBSTRING(A.
    ANO, 6, 2) = 2 THEN 6
WHEN SUBSTRING(A.ANO, 1, 4) - E.ANO_INICIO = 3 AND SUBSTRING(A.
    ANO, 6, 2) = 1 THEN 7
WHEN SUBSTRING(A.ANO, 1, 4) - E.ANO_INICIO = 3 AND SUBSTRING(A.
    ANO, 6, 2) = 2 THEN 8
WHEN SUBSTRING(A.ANO, 1, 4) - E.ANO_INICIO = 4 AND SUBSTRING(A.
    ANO, 6, 2) = 1 THEN 9
WHEN SUBSTRING(A.ANO, 1, 4) - E.ANO_INICIO = 4 AND SUBSTRING(A.
    ANO, 6, 2) = 2 THEN 10
ELSE 10
END AS VALIDANDO
INTO #TIPO_SEMESTRE
FROM DimProximoSemestre (NOLOCK) A
JOIN DimMATERIA (NOLOCK) C ON A.ID_MATERIA = C.ID_MATERIA
JOIN FatoHISTORICO_ESTUDANTIL (NOLOCK) D ON C.ID_MATERIA = D.
    ID_MATERIA
JOIN DimALUNO (NOLOCK) E ON D.ID_ALUNO = E.ID_ALUNO
JOIN #TEMP F ON E.ID_ALUNO = F.ID_ALUNO
WHERE EXISTS (SELECT *
                                FROM #TEMP B WHERE A.
                                    ID_MATERIA = B.
                                    ID_MATERIA
                                )

AND D.STATUS_MATERIA = 1

IF OBJECT_ID('#MATERIAS_POSSIVEIS') IS NOT NULL
    DROP TABLE #MATERIAS_POSSIVEIS;

SELECT A.*
INTO #MATERIAS_POSSIVEIS
FROM DimProximoSemestre (NOLOCK) A
LEFT JOIN #TIPO_SEMESTRE (NOLOCK) C ON A.ID_MATERIA = C.
    ID_MATERIA
WHERE NOT EXISTS (SELECT * FROM
                                #TEMP B WHERE A.

```

```

ID_MATERIA = B.
ID_MATERIA)
AND A.TIPO_MATERIA <= ISNULL(C.VALIDANDO, (SELECT MAX(VALIDANDO)
FROM #TIPO_SEMESTRE))

INSERT INTO [dbo].[FATO_NOVO_SEMESTRE]
(
NOME_ALUNO,
ID_ALUNO,
ID_MATERIA,
NOME_MATERIA,
DATA,
TIPO_MATERIA,
HORARIO
)
SELECT DISTINCT B.NOME_ALUNO, B.ID_ALUNO, A.ID_MATERIA, A.
NOME_MATERIA, X.ANO, X.TIPO_MATERIA, X.HORARIO
FROM (
SELECT *, ROW_NUMBER() OVER (PARTITION BY HORARIO ORDER
BY ID_MATERIA) N
FROM #MATERIAS_POSSIVEIS
)X
JOIN DimMATERIA (NOLOCK) A ON X.ID_MATERIA = A.ID_MATERIA
CROSS JOIN #TIPO_SEMESTRE B
WHERE X.N = 1
AND NOT EXISTS (SELECT * FROM DBO.FATO_NOVO_SEMESTRE (NOLOCK) Z
WHERE B.ID_ALUNO = Z.ID_ALUNO AND X.ID_MATERIA = Z.ID_MATERIA)

DROP TABLE #TEMP
DROP TABLE #TIPO_SEMESTRE
DROP TABLE #MATERIAS_POSSIVEIS
```

Referências

- BASSIL, Y. A data warehouse design for a typical university information system. *arXiv preprint arXiv:1212.2071*, 2012. Citado nas páginas 11 e 20.
- CAIRO, A. *The Functional Art: An introduction to information graphics and visualization*. Thousand Oaks, CA: New Riders Publishing, 2012. Citado na página 28.
- CARVALHO, G. P.; FREITAS, F. L. Qualidade de dados no data warehouse de uma empresa de e-commerce. *Núcleo do Conhecimento*, v. 9, n. junho, p. 187–201, 2020. ISSN 2448-0959. Citado na página 14.
- CHAUDHURI, S.; DAYAL, U. An overview of data warehousing and olap technology. *ACM SIGMOD Record*, ACM, v. 26, n. 1, p. 65–74, 1997. Citado na página 16.
- DUBÉ, L.; PARÉ, G.; SICOTTE, C. The effects of strategic and operational information technology alignment on hospital performance. *Journal of medical systems*, Springer, v. 31, n. 1, p. 29–36, 2007. Citado na página 17.
- GRAY, J.; REUTER, A. *Transaction Processing: Concepts and Techniques*. [S.l.]: Morgan Kaufmann, 1993. Citado na página 16.
- HAN, J.; KAMBER, M. *Data Mining: Concepts and Techniques*. 2. ed. [S.l.]: Morgan Kaufmann Publishers, 2006. Citado na página 17.
- IMHOFF, C.; GALEMMO, J.; GEIGER, J. G. *Building the Customer-Centric Enterprise*. 1st. ed. [S.l.]: John Wiley & Sons, 1998. Citado na página 14.
- INMON, W. H. *Building the Data Warehouse*. 4th. ed. [S.l.]: John Wiley & Sons, 2005. Citado nas páginas 14 e 16.
- KIMBALL, R. *The Data Warehouse Toolkit: Practical Techniques for Building Dimensional Data Warehouses*. 1st. ed. [S.l.]: John Wiley & Sons, 1996. Citado na página 16.
- KIMBALL, R.; ROSS, M. *The Data Warehouse Toolkit: The Definitive Guide to Dimensional Modeling*. 3rd. ed. [S.l.]: John Wiley & Sons, 2013. Citado nas páginas 14 e 15.
- KORTH, H.; SILBERSCHATZ, A.; SUDARSHAN, S. *Database System Concepts*. 6th. ed. [S.l.]: McGraw-Hill, 2012. Citado na página 13.
- LEVITT, T. The globalization of markets. *Harvard Business Review*, Harvard University Press, v. 61, n. 3, p. 92–102, 1983. Citado na página 10.
- LINTHICUM, D. *Cloud Computing and SOA Convergence in Your Enterprise*. 1st. ed. [S.l.]: Pearson Education, 2013. Citado na página 14.
- MICROSOFT. *Entenda o esquema em estrela e a importância para o Power BI*. acessado em 05 de março de 2023. <<https://learn.microsoft.com/pt-br/power-bi/guidance/star-schema>>. Citado na página 15.
- MOSCOSO-ZEA, O.; PAREDES-GUALTOR, J.; LUJÁN-MORA, S. A holistic view of data warehousing in education. *IEEE access*, IEEE, v. 6, p. 64659–64673, 2018. Citado na página 11.

NAVATHE RAMEZ ELMASRI, S. B.; SYSTEMS, F. of D. *Fundamentals of Database Systems*. 7th. ed. [S.l.]: Pearson, 2019. Citado na página 13.

RABELLO, G. *Ambiente BI*. acessado em 05 de março de 2023.
<https://www.siteware.com.br/gestao-estrategica/o-que-e-bi-business-intelligence/>. Citado na página 17.

SANTOSO, L. W. et al. Data warehouse with big data technology for higher education. *Procedia Computer Science*, Elsevier, v. 124, p. 93–99, 2017. Citado nas páginas 7, 11 e 19.

SCHULTE, W. R. Planning and implementing information management systems. *Journal of Information Systems Management*, v. 26, n. 2, p. 102–112, 2009. Citado na página 10.

WHITE, K.; ROLLINGS, A. Big data: A revolution that will transform how we live, work, and think. *Harvard Business Review*, Harvard Business Publishing, v. 91, n. 12, p. 24–25, 2013. Citado na página 10.