

**UNIVERSIDADE DE CAXIAS DO SUL  
ÁREA DO CONHECIMENTO DE CIÊNCIAS EXATAS E  
ENGENHARIAS**

**LUCAS WEBBER MOLIN**

**PROPOSTA DE SOLUÇÃO DE VISUALIZAÇÃO DE DADOS  
ATRAVÉS DE DASHBOARDS PARA EMPRESA DO RAMO DE  
TRANSPORTES**

**CAXIAS DO SUL**

**2021**

**LUCAS WEBBER MOLIN**

**PROPOSTA DE SOLUÇÃO DE VISUALIZAÇÃO DE DADOS  
ATRAVÉS DE DASHBOARDS PARA EMPRESA DO RAMO DE  
TRANSPORTES**

Trabalho de Conclusão de Curso apresentado como requisito parcial à obtenção do título de Bacharel em Ciência da Computação na Área do Conhecimento de Ciências Exatas e Engenharias da Universidade de Caxias do Sul.

Orientador: Prof.<sup>a</sup> Dra. Helena Graziotin Ribeiro

**CAXIAS DO SUL**

**2021**

**LUCAS WEBBER MOLIN**

**PROPOSTA DE SOLUÇÃO DE VISUALIZAÇÃO DE DADOS  
ATRAVÉS DE DASHBOARDS PARA EMPRESA DO RAMO DE  
TRANSPORTES**

Trabalho de Conclusão de Curso apresentado como requisito parcial à obtenção do título de Bacharel em Ciência da Computação na Área do Conhecimento de Ciências Exatas e Engenharias da Universidade de Caxias do Sul.

**Aprovado em 03/12/2021**

**BANCA EXAMINADORA**

---

Prof.<sup>a</sup> Dra. Helena Graziottin Ribeiro  
Universidade de Caxias do Sul - UCS

---

Prof. Dr. Daniel Luis Notari  
Universidade de Caxias do Sul - UCS

---

Prof.<sup>a</sup> Ma. Gabriele Dani  
Universidade de Caxias do Sul - UCS

## **AGRADECIMENTOS**

Agradeço à Prof.<sup>a</sup> Dra. Helena Graziottin Ribeiro pela orientação neste trabalho, e também à empresa envolvida pelo fornecimento dos dados.

*“Deve-se aprender sempre, até mesmo com um inimigo.”*

***Isaac Newton***

## RESUMO

Uma empresa fabricante de acessórios para veículos de transporte de carga coleta um vasto volume de dados de tais veículos e os monitora a partir de um sistema *web* através de gráficos avulsos e relatórios. O presente trabalho teve como objetivo propôr uma solução de visualização de dados centralizando todos os dados em uma só tela através de um *dashboard*. Para isso, foi estudada a bibliografia essencial para esse meio, que envolve conceitos como *business intelligence*, *data warehouse* e visualização de dados. Após, foi realizado o planejamento da estrutura do *data warehouse* e o esboço dos *dashboards*, para então desenvolver um extrator de dados que realiza o processo de extração, transformação e carga (ETL) dos dados a partir de planilhas fornecidas pela empresa. Por fim, foram construídos dois *dashboards* operacionais, utilizando a ferramenta Power BI, que focam na visualização de dados de performance dos veículos.

**Palavras-chave:** *business intelligence, dashboard, data warehouse.*

## LISTA DE FIGURAS

Figura 1 – Ferramentas e técnicas utilizadas em sistemas de BI . . . . .	12
Figura 2 – Arquitetura de BI . . . . .	13
Figura 3 – Esquema em estrela . . . . .	16
Figura 4 – Esquema em floco de neve . . . . .	17
Figura 5 – Gráfico de linhas . . . . .	19
Figura 6 – Gráfico de inclinação . . . . .	20
Figura 7 – Gráficos de barras . . . . .	21
Figura 8 – Gráficos de barras empilhadas . . . . .	21
Figura 9 – Gráfico de área . . . . .	22
Figura 10 – Exemplo de gráfico de pizza e sua comparação com um gráfico de barras . . . . .	22
Figura 11 – Gráfico de dispersão . . . . .	23
Figura 12 – Exemplo de tabela e mapa de calor . . . . .	24
Figura 13 – Gráfico de cascata . . . . .	25
Figura 14 – Interface gráfica do Power BI para criação de <i>dashboards</i> . . . . .	28
Figura 15 – Interface gráfica do Tableau para criação de <i>dashboards</i> . . . . .	29
Figura 16 – Exemplo de código-fonte usando a biblioteca Highcharts para criação de <i>dashboards</i> . . . . .	30
Figura 17 – Exemplo de gráficos do sistema <i>web</i> da empresa fornecedora dos dados . . . . .	32
Figura 18 – Esboço do <i>dashboard</i> geral . . . . .	33
Figura 19 – Esboço da caixa de filtro . . . . .	34
Figura 20 – Esboço do indicador de ativação de veículos . . . . .	34
Figura 21 – Esboço do indicador de operadores em atividade . . . . .	34
Figura 22 – Esboço do indicador de abastecimento geral . . . . .	35
Figura 23 – Esboço do indicador de monitoramento de infrações de velocidade . . . . .	35
Figura 24 – Esboço do indicador de monitoramento de infrações de temperatura . . . . .	36
Figura 25 – Esboço da tabela de dados gerais . . . . .	36
Figura 26 – Esboço do <i>dashboard</i> específico por veículo . . . . .	37
Figura 27 – Esboço da caixa de filtro . . . . .	38
Figura 28 – Esboço do indicador de aproveitamento de tempo . . . . .	38
Figura 29 – Esboço do indicador de distância percorrida . . . . .	39
Figura 30 – Esboço do indicador de abastecimento . . . . .	39
Figura 31 – Esboço do indicador de velocidade . . . . .	40
Figura 32 – Esboço do indicador de temperatura . . . . .	40
Figura 33 – Esboço do indicador de aceleração . . . . .	41
Figura 34 – Esboço do indicador de desaceleração . . . . .	41
Figura 35 – Esboço do indicador de rotação . . . . .	42

Figura 36 – Esboço do indicador de reversão de velocidade . . . . .	42
Figura 37 – Esboço do indicador de localização . . . . .	43
Figura 38 – Esboço do indicador de histórico de operador . . . . .	43
Figura 39 – Esboço do novo <i>dashboard</i> geral de veículos . . . . .	44
Figura 40 – Esboço do indicador de registro de impacto . . . . .	44
Figura 41 – Esboço da nova tabela de dados gerais . . . . .	45
Figura 42 – Esboço do novo <i>dashboard</i> específico de veículos . . . . .	46
Figura 43 – Esboço do indicador de horímetro de motor ligado . . . . .	47
Figura 44 – Etapas para extração e consulta dos dados . . . . .	48
Figura 45 – Esboço da estrutura do <i>data warehouse</i> . . . . .	48
Figura 46 – Pasta contendo os arquivos de <i>log</i> das extrações . . . . .	55
Figura 47 – <i>Screenshot</i> da execução do extrator . . . . .	57
Figura 48 – Tela inicial da ferramenta Power BI . . . . .	58
Figura 49 – Construção de gráficos na ferramenta Power BI . . . . .	63
Figura 50 – Resultado final do <i>dashboard</i> geral . . . . .	63
Figura 51 – Indicador de Veículos em Atividade . . . . .	64
Figura 52 – Indicador de Operadores em Atividade . . . . .	64
Figura 53 – Indicadores de Infrações de Velocidade e Impacto . . . . .	65
Figura 54 – Tabela de produtividade e infrações . . . . .	65
Figura 55 – Resultado final do <i>dashboard</i> específico . . . . .	66
Figura 56 – Indicador de horímetro de motor ligado . . . . .	67
Figura 57 – Indicador de horímetro de carga e deslocamento . . . . .	67
Figura 58 – Indicador de histórico de velocidade . . . . .	67
Figura 59 – Indicador de histórico de temperatura . . . . .	68
Figura 60 – Indicador de histórico de aceleração e desaceleração . . . . .	68
Figura 61 – Indicador de último impacto . . . . .	69
Figura 62 – Diagrama ER referente aos dados dos veículos de transporte . . . . .	72



# SUMÁRIO

<b>1</b>	<b>INTRODUÇÃO</b>	<b>10</b>
1.1	PROBLEMA	10
1.2	OBJETIVO	10
1.3	ESTRUTURA DO TRABALHO	11
<b>2</b>	<b>BUSINESS INTELLIGENCE</b>	<b>12</b>
2.1	BENEFÍCIOS	13
2.2	ARQUITETURA	13
<b>2.2.1</b>	<b><i>Data Warehouse</i></b>	<b>13</b>
2.2.1.1	Elementos de um <i>Data Warehouse</i>	14
2.2.1.2	Tipos de tabelas de fatos	15
2.2.1.3	Tipos de tabelas dimensões	15
2.2.1.4	Estrutura de um <i>Data Warehouse</i>	16
<b>2.2.2</b>	<b>ETL</b>	<b>17</b>
<b>3</b>	<b>VISUALIZAÇÃO DE DADOS</b>	<b>18</b>
3.1	GRÁFICOS E TABELAS	18
<b>3.1.1</b>	<b>Gráfico de linhas</b>	<b>18</b>
<b>3.1.2</b>	<b>Gráfico de inclinação</b>	<b>19</b>
<b>3.1.3</b>	<b>Gráfico de barras</b>	<b>20</b>
<b>3.1.4</b>	<b>Gráfico de área</b>	<b>21</b>
<b>3.1.5</b>	<b>Gráfico de pizza</b>	<b>22</b>
<b>3.1.6</b>	<b>Gráfico de dispersão</b>	<b>23</b>
<b>3.1.7</b>	<b>Tabela</b>	<b>23</b>
<b>3.1.8</b>	<b>Gráfico de cascata</b>	<b>24</b>
3.2	<i>DASHBOARDS</i>	25
<b>3.2.1</b>	<b>Tipos de <i>dashboards</i></b>	<b>25</b>
<b>3.2.2</b>	<b>Construção de <i>dashboards</i></b>	<b>26</b>
<b>4</b>	<b>FERRAMENTAS DE VISUALIZAÇÃO</b>	<b>28</b>
4.1	POWER BI	28
4.2	TABLEAU	29
4.3	HIGHCHARTS	29
<b>5</b>	<b>PROPOSTA DE SOLUÇÃO</b>	<b>31</b>
5.1	DADOS	31
5.2	<i>MOCKUP DOS DASHBOARDS</i>	32

5.2.1	<i>Dashboard geral de veículos</i> . . . . .	33
5.2.2	<i>Dashboard específico por veículo</i> . . . . .	37
5.3	Reestruturação dos <i>dashboards</i> . . . . .	43
5.3.1	<b>Novo <i>dashboard</i> geral de veículos</b> . . . . .	44
5.3.2	<b>Novo <i>dashboard</i> específico por veículo</b> . . . . .	45
5.4	EXTRAÇÃO E TRANSFORMAÇÃO DOS DADOS . . . . .	47
5.4.1	<i>Data Warehouse</i> . . . . .	48
6	<b>DESENVOLVIMENTO DA SOLUÇÃO</b> . . . . .	49
6.1	DADOS . . . . .	49
6.2	EXTRAÇÃO, TRANSFORMAÇÃO E CARGA . . . . .	50
6.3	CONSTRUÇÃO . . . . .	57
6.3.1	<b>Consultas para o <i>dashboard</i> geral de veículos</b> . . . . .	58
6.3.2	<b>Consultas para o <i>dashboard</i> específico por veículo</b> . . . . .	60
6.3.3	<b>Construção dos <i>dashboards</i></b> . . . . .	62
7	<b>CONCLUSÕES</b> . . . . .	70
	<b>REFERÊNCIAS</b> . . . . .	71
	<b>APÊNDICE A – APÊNDICE(S)</b> . . . . .	72

# 1 INTRODUÇÃO

Com o advento dos computadores e, conseqüentemente, dos sistemas de informação empresariais, tornou-se possível a coleta e armazenamento massivo de dados. Assim, tendo-se em mãos um vasto volume de dados e máquinas capazes de processá-los rapidamente, a análise de dados tornou-se mais eficiente, possibilitando às empresas a compreenderem melhor seus negócios. Para facilitar tais análises, empresas de informática como, por exemplo, a Microsoft Corporation, desenvolveram ferramentas de visualização de dados para a construção de *dashboards*, com gráficos, tabelas e mapas. Para o desenvolvimento de tais soluções de visualização de dados, estão envolvidos processos e recursos de engenharia, ciência e análise de dados, dentre os quais destacam-se: ETL (*Extract Transform & Load*), *Data Warehouse* e *Data Marts* (SHARDA; DELEN; TURBAN, 2018).

Uma empresa situada na cidade de Caxias do Sul, que optou por anonimato, desenvolve e fornece acessórios para veículos industriais de movimentação de cargas a fim de aumentar a segurança e eficiência da logística interna de seus clientes.

Com uma vasta linha de produtos, fornecidos pela empresa em questão, que inclui limitadores de velocidade, sinalizadores visuais, faróis, horímetros, aceleradores, potenciômetros, conectores de baterias, entre outros produtos que são conectados aos veículos, é gerado um enorme volume de dados referente ao uso de tais acessórios nos veículos. Os dados gerados são armazenados em um banco de dados MariaDB<sup>1</sup>, que são visualizados através de gráficos em telas separadas de um sistema *web* feito em PHP (Hypertext Preprocessor).

## 1.1 PROBLEMA

Com o uso de acessórios fornecidos pela empresa em questão para veículos de transporte de carga, é gerado um vasto volume de dados, e com a preocupação de seus gestores e funcionários em monitorar o uso a fim de monitorar o desempenho dos veículos e identificar anormalidades, torna-se necessário o desenvolvimento de uma solução que permita visualizar os dados de forma combinada, agrupando características e dados para uma melhor tomada de decisão.

## 1.2 OBJETIVO

O objetivo geral deste trabalho é construir uma solução de visualização de dados através de *dashboards* para uma empresa de transportes de carga. Para atingir tal objetivo, é necessário o cumprimento dos objetivos específicos:

---

<sup>1</sup> <https://mariadb.org/>

1. Realizar um estudo bibliográfico da teoria e prática envolvida no desenvolvimento de soluções de preparação e visualização de dados.
2. Elaborar uma proposta de solução para o problema em questão.
3. Realizar o desenvolvimento da proposta de solução.

### 1.3 ESTRUTURA DO TRABALHO

Este trabalho está organizado da seguinte forma:

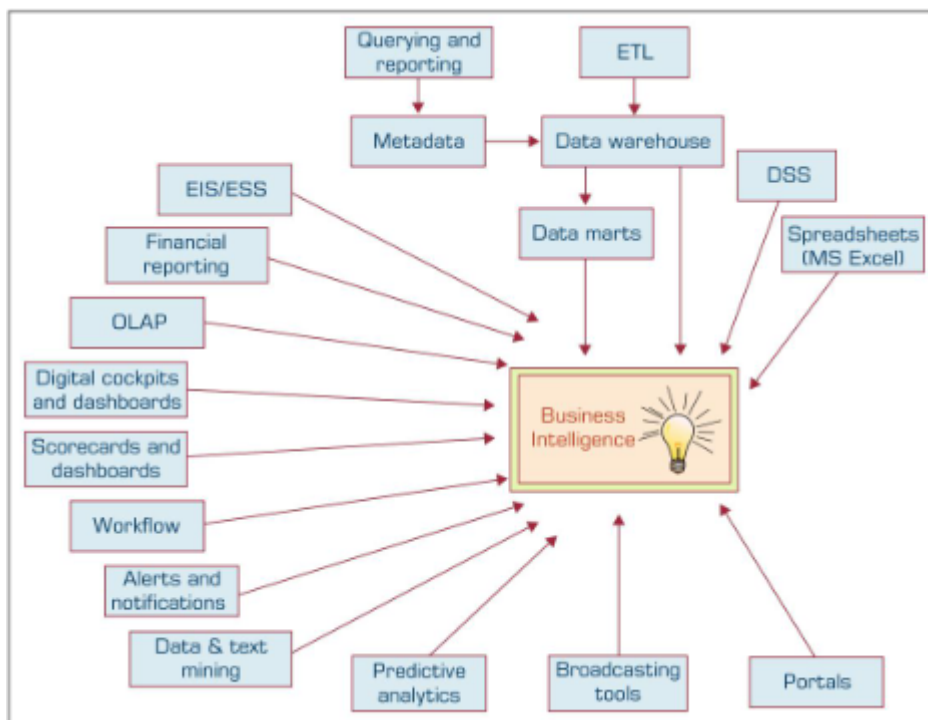
- O Capítulo 2 apresenta os conceitos de *Business Intelligence*, aprofundando-se em *Data Warehouse* e ETL.
- O Capítulo 3 apresenta os principais elementos de visualização de dados como gráficos e tabelas e como utilizá-los para a construção de *dashboards*.
- O Capítulo 4 apresenta as principais ferramentas e bibliotecas para a construção de *dashboards*.
- O Capítulo 5 apresenta a proposta de solução para o problema em questão, aprofundando-se no problema, na ferramenta e nos processos que serão utilizados para a criação de um *data warehouse* e *dashboards*.
- O Capítulo 6 apresenta as etapas para o desenvolvimento da solução proposta.
- O Capítulo 7 apresenta as conclusões deste trabalho.

## 2 BUSINESS INTELLIGENCE

*Business Intelligence* (BI) é um termo cunhado no final da década de 1980 pelo analista Howard Dresden da Gartner Group (DÍAZ, 2010) que engloba arquiteturas, ferramentas, bases de dados, aplicações e metodologias aplicadas à análise de dados. O processo é baseado na transformação de dados em informação para uso na tomada de decisões (SHARDA; DELEN; TURBAN, 2018).

Apesar do termo ter sido cunhado na década de 1980, a ideia era já utilizada pelos povos antigos do Oriente Médio na coleta de dados referentes ao comportamento das marés, períodos de chuva e seca e posição dos astros, a fim de obter informações que seriam usadas para gerir melhor suas aldeias (PRIMAK, 2020). Com o advento dos computadores modernos e, conseqüentemente, dos sistemas de informação empresariais (EIS) na década de 1980, buscou-se o desenvolvimento de soluções para auxiliar no processo de análise de dados no meio empresarial. Assim, no início da década de 1990, funcionalidades como geração de relatórios, análise de previsão e predição, *data drilling*, entre outras, apareceram sob o nome de BI. A partir de 2005, sistemas de BI começaram a incluir inteligência artificial para prover capacidades analíticas mais poderosas (SHARDA; DELEN; TURBAN, 2018). A figura 1 apresenta diversas ferramentas e técnicas utilizadas em sistemas de BI.

Figura 1 – Ferramentas e técnicas utilizadas em sistemas de BI



Fonte: SHARDA, DELEN & TURBAN (2018).

## 2.1 BENEFÍCIOS

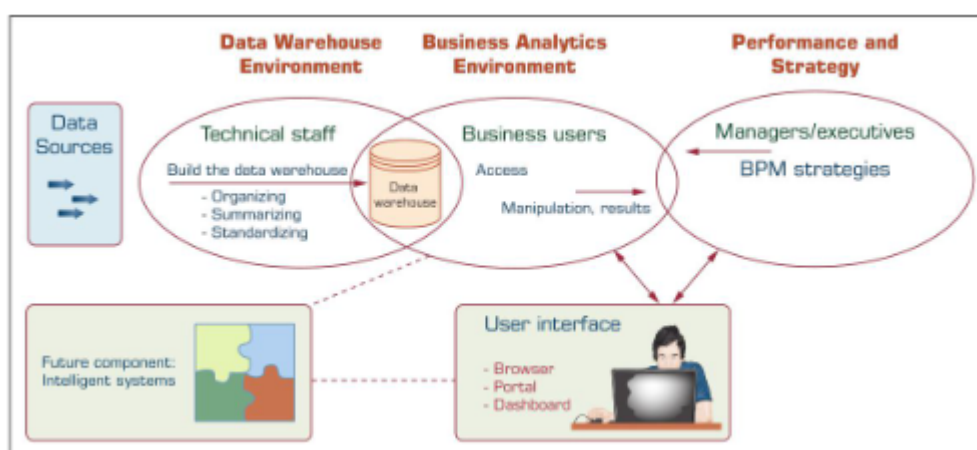
Segundo DÍAZ (2010), a implantação de sistemas de BI proporciona benefícios como: a melhora e agilidade na tomada de decisões devido aos dados transformados em informação, que por sua vez geram conhecimento; a possibilidade de uma visão única, histórica, persistente e de qualidade sobre toda a informação do negócio; obter informação atualizada tanto a nível agrupado quanto a nível detalhado, entre outros benefícios.

Porém, para obter tais benefícios, um sistema de BI deve ser facilmente compreensível, exibindo dados de forma intuitiva e óbvia para um usuário leigo como um gestor empresarial comum, por exemplo. É importante também que os dados apresentados sejam confiáveis e consistentes, dado que são extraídos e processados a partir de diversas fontes (KIMBALL; ROSS, 2009).

## 2.2 ARQUITETURA

Um sistema de BI contém quatro componentes principais: um *Data Warehouse* (DW), estrutura que armazena os dados; *business analytics*, conjunto de ferramentas para manipulação, mineração e análise de dados consultados do DW; *Business Process Management* (BPM), processo para monitoramento e análise de performance, e uma interface gráfica (por exemplo, *dashboard*) (SHARDA; DELEN; TURBAN, 2018). Na figura 2 está ilustrada esse modelo de arquitetura.

Figura 2 – Arquitetura de BI



Fonte: SHARDA, DELEN & TURBAN (2018).

### 2.2.1 *Data Warehouse*

Conforme apresentado na figura 1, um sistema de BI é composto por diversas ferramentas, técnicas e componentes como ETL (Extract Transform Load), mineração de dados, OLAP (*Online Analytical Processing*), *dashboards*, entre outras, das quais destaca-se o *Data*

*Warehouse*, responsável pelo armazenamento dos dados e considerado também um dos principais componentes de uma arquitetura de BI, conforme citado na seção 2.3 e apresentado na figura 2. Segundo INMON (2002), um *Data Warehouse* é o coração de um sistema de BI, funcionando como um depósito de dados que proporciona uma visão global, comum e integrada dos dados de uma organização, com as seguintes propriedades: estável, coerente, confiável, histórico e granulável. Para o armazenamento de tais estruturas de dados, comumente são utilizados bancos de dados relacionais. Como exemplo de bancos de dados relacionais, há: MariaDB, MySQL, Oracle, PostgreSQL, entre outros. Segundo DÍAZ (2010), deve-se considerar também os seguintes elementos no contexto de um *Data Warehouse*:

- *Data Warehousing*: processo de extração e filtragem dos dados comuns de uma organização, provenientes de diferentes sistemas de informação.
- *Data Mart*: é o subconjunto de dados do *Data Warehouse* cujo objetivo é atender a uma determinada análise, função ou necessidade.
- *Data Lake*: é um tipo de repositório que armazena grandes volumes de dados brutos em seus formatos originais.
- *Operational Data Store*: é um tipo de depósito de dados que contém apenas os dados mais recentes e não o histórico completo.
- *Staging Area*: é o sistema que está entre as fontes de dados e o *Data Warehouse* com o objetivo de facilitar a extração de dados, melhorar a qualidade dos dados e ser usado como *cache* para posteriormente ser realizado o processo de *Data Warehousing*.
- ETL (Extract Transform Load): processo composto por três fases: extração, transformação e carregamento dos dados, que podem ser realizadas utilizando-se ferramentas e linguagens variadas. Ao longo dos anos, também surgiram variações como ELT, onde o processo de carga antecede a transformação.
- OLAP (Online Analytical Processing): operações para manipulação e análise de dados sob múltiplas perspectivas.
- Metadados: dados estruturados e codificados que descrevem características de instâncias.

#### 2.2.1.1 Elementos de um *Data Warehouse*

Segundo DÍAZ (2010), em um *Data Warehouse* o objetivo principal é armazenar os dados de forma a otimizar as consultas, dividindo os dados nos seguintes elementos:

- Tabela de fatos: armazena os dados referentes a processos de negócio de uma organização. Por exemplo, uma tabela de fatos pode armazenar dados de compras realizadas numa loja.

- Tabela de dimensão: armazena os dados referentes a entidades específicas de processos de negócio. No exemplo de compras em uma loja, tabelas dimensões são aquelas que armazenam uma entidade específica como cliente, produto e representante, por exemplo.
- Métrica: são os indicadores de um processo de negócio. Por exemplo, o valor de uma compra.

### 2.2.1.2 Tipos de tabelas de fatos

Segundo DÍAZ (2010), existem quatro diferentes tipos de tabelas de fatos:

- *Transaction Fact Table*: representam eventos que aconteceram em um determinado espaço-tempo, caracterizando-se por permitir analisar os dados com o máximo de detalhamento. Por exemplo, uma tabela contendo dados de vendas.
- *Factless Fact Tables/Coverage Table*: são semelhantes ao tipo citado no item anterior, porém não tem detalhamento.
- *Periodic Snapshot Fact Table*: são tabelas utilizadas para armazenar informações de forma periódica em intervalos de tempo regulares. Por exemplo, dados coletados de equipamentos a cada intervalo de tempo.
- *Accumulating Snapshot Fact Table*: representam o ciclo de vida completo de uma atividade ou processo, caracterizando-se por representar múltiplas dimensões relacionadas a eventos.

### 2.2.1.3 Tipos de tabelas dimensões

Segundo DÍAZ (2010), existem diversos tipos de tabelas dimensões, das quais destacam-se os tipos *Slowly Changing Dimension* (SCD):

- SCD Tipo 0: os dados nunca são alterados. Por exemplo, uma tabela de cadastro de planetas do sistema solar.
- SCD Tipo 1: não são armazenados dados históricos; dados novos sobrescrevem dados antigos. Por exemplo, o preço atual de cada produto.
- SCD Tipo 2: são armazenados os dados históricos, havendo a criação de novas linhas com data e *surrogate key*. Por exemplo, o endereço residencial de uma pessoa.
- SCD Tipo 3: são armazenados os dados históricos, havendo a criação de novas colunas com os valores antigos e substituindo os novos valores nas colunas atuais. Por exemplo, data de lançamento da nova versão de um jogo.



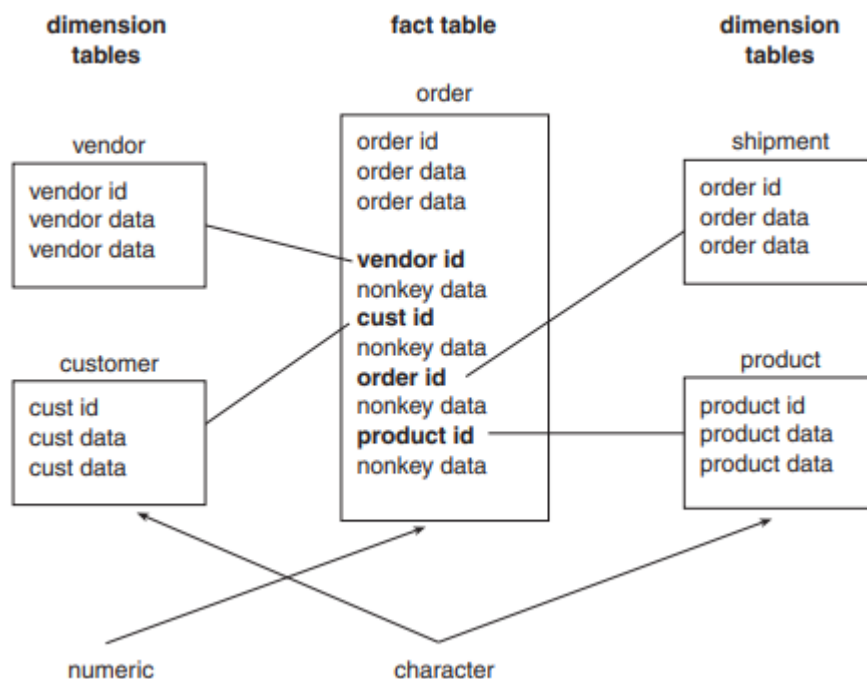
- SCD Tipo 4: são armazenados os dados históricos, dividindo-se em duas tabelas, uma para dados atuais e outra para dados antigos. Por exemplo, uma tabela dos produtos da temporada atual e outra tabela dos produtos das temporadas anteriores.
- SCD Tipo 6/Híbrida: consiste na combinação dos tipos 1, 2 e 3, da qual a soma resulta em 6, por isso esse nome.

#### 2.2.1.4 Estrutura de um *Data Warehouse*

Segundo KIMBALL & ROSS (2009), existem principalmente dois tipos de estruturação de um *Data Warehouse*:

- Esquema em estrela: consiste em uma tabela de fatos no centro e diversas tabelas de dimensão para cada ponto de análise para descrever o fato. A figura 3 ilustra um exemplo do esquema estrela contendo uma tabela de pedidos como fato e tabelas de produto, cliente, fornecedor e envio como dimensão.

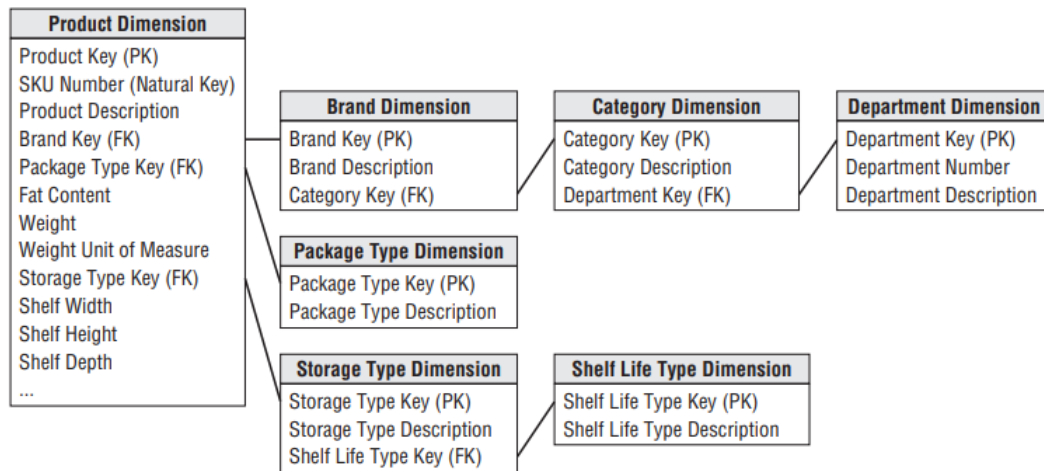
Figura 3 – Esquema em estrela



Fonte: (INMON, 2002)

- Esquema em floco de neve: é um esquema de representação derivado do esquema em estrela, no qual as tabelas de dimensão possuem vínculo com múltiplas tabelas de dimensão. Por exemplo, a tabela produtos do esquema em estrela apresentado na figura 3 poderia ter vínculo com uma tabela de marcas referente aos produtos (e a tabela de marcas, por sua vez, poderia ter vínculo com uma tabela de categoria) como mostra a figura 4.

Figura 4 – Esquema em floco de neve



Fonte: (KIMBALL; ROSS, 2009)

## 2.2.2 ETL

Segundo AMARAL (2016 apud BASSO, 2020), a construção de um *Data Warehouse* é composta pelas etapas de extração, transformação e carregamento dos dados. O conjunto de tais etapas denomina-se ETL (*Extract Transform Load*, traduzindo-se do inglês para o português como extração, transformação e carga).

- Extração: os dados são extraídos dos sistemas de informação através de tecnologias como SQL (*Structured Query Language*), Pentaho, Python, entre outras.
- Transformação: após a extração, os dados são enviados para um ambiente temporário no qual são realizados ajustes nos dados para melhorar sua qualidade.
- Carga: após a transformação dos dados, é realizado o carregamento dos dados para um ambiente através do qual a camada de visualização irá consultar os dados para exibir ao usuário final.

## 3 VISUALIZAÇÃO DE DADOS

A visualização de dados é o resultado da utilização de tecnologias computacionais para transformar dados abstratos em modelos visuais, resultando em informações mais facilmente compreensíveis (RIBEIRO, 2015). A visualização de dados em ferramentas de BI é feita através de uma interface gráfica composta, principalmente, por *dashboards*, nos quais são utilizados gráficos para exibir o resultado obtido após a extração e processamento dos dados (SHARDA; DELEN; TURBAN, 2018). Segundo RIBEIRO (2015), os gráficos revelam, através de suas formas geométricas e cores, variações e padrões de repetição que facilitam a compreensão das informações apresentadas.

### 3.1 GRÁFICOS E TABELAS

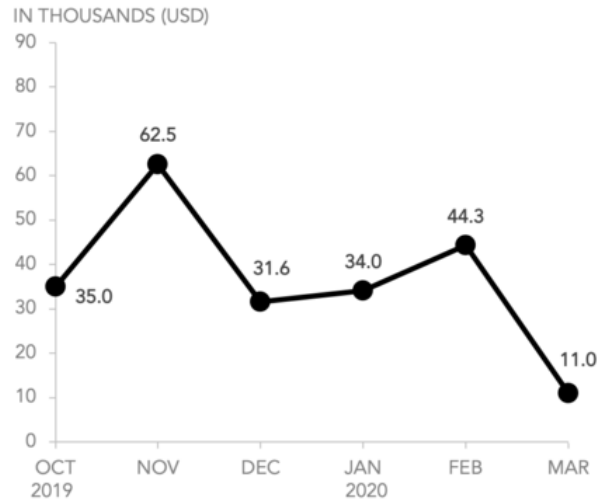
Segundo KNAFLIC (2019), os principais tipos de gráficos (e outros elementos) utilizados para visualização de dados são gráficos de linhas, de barras, de inclinação, de área, entre outros, apresentados nesta seção:

#### 3.1.1 Gráfico de linhas

Os gráficos de linhas são comumente utilizados para visualizar dados contínuos. Por haver uma conexão entre cada ponto, torna-se estranho utilizar para dados categóricos. Como exemplo de valores utilizados há: dias, meses, anos. É possível também utilizar mais de uma linha (série). A figura 5 ilustra um gráfico de linhas.

Figura 5 – Gráfico de linhas

### 6-Month sales report and forecast



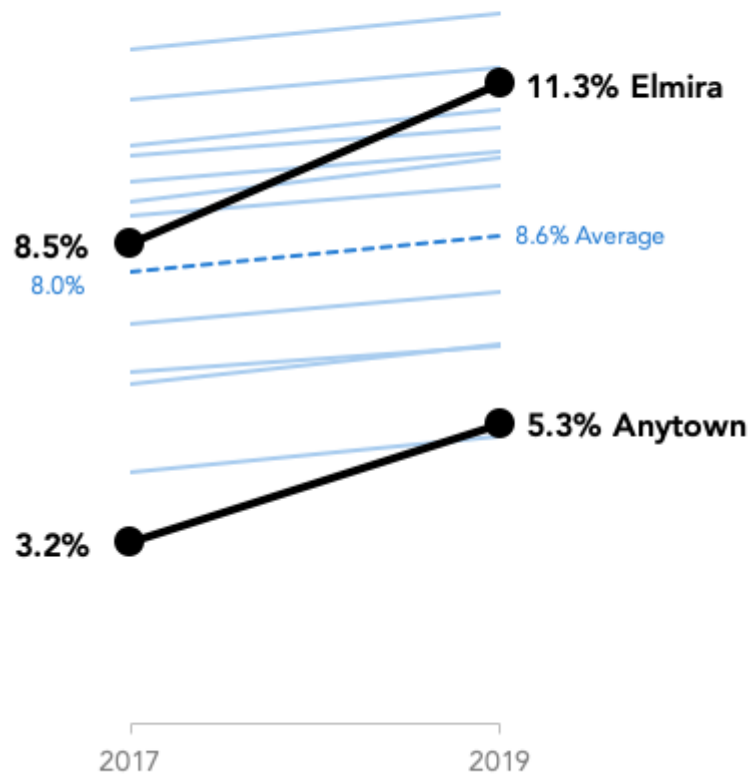
Fonte: <https://www.storytellingwithdata.com/chart-guide>.  
Acesso em: 28 abr. 2021

### 3.1.2 Gráfico de inclinação

Os gráficos de inclinação são uma variação do gráfico de linhas. Normalmente são utilizados quando há apenas dois períodos de tempo e quando deseja-se exibir prontamente os aumentos ou diminuições relativas dentre dois pontos. A figura 6 exemplifica um gráfico de linhas utilizado para exibir as vendas ao longo de alguns meses.

Figura 6 – Gráfico de inclinação

### Diabetes rate at all regional medical centers PERCENTAGE OF PATIENTS WITH DIABETES



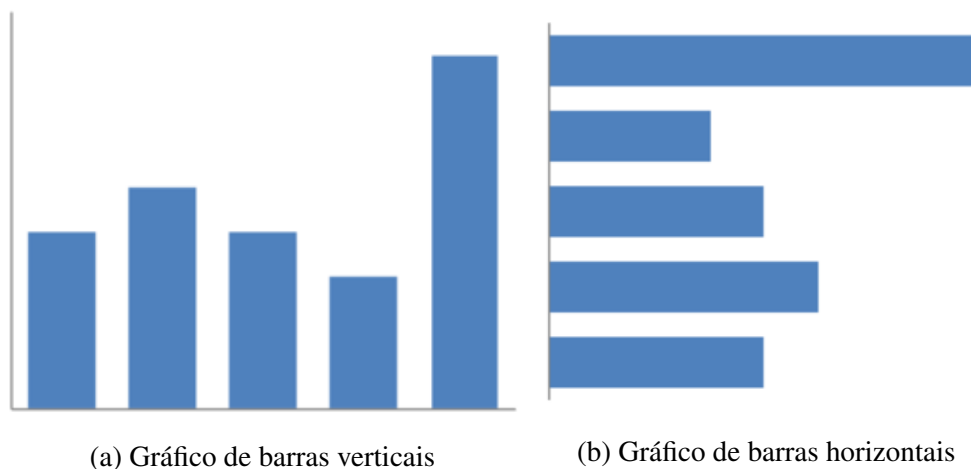
Fonte: <https://www.storytellingwithdata.com/chart-guide>. Acesso em: 28 abr. 2021

### 3.1.3 Gráfico de barras

Os gráficos de barras são utilizados para ilustrar dados categóricos como, por exemplo, o valor de vendas de cada mês ou produto. Segundo KNAFLIC (2019), gráficos de barras são simples e comuns, o que facilita a leitura pelo público em geral. Os gráficos de barras podem ser apresentados de diversas formas: verticalmente, como ilustra a figura 7a; horizontalmente, conforme a figura 7b; de forma empilhada, como mostra a figura 8a e a figura 8b.

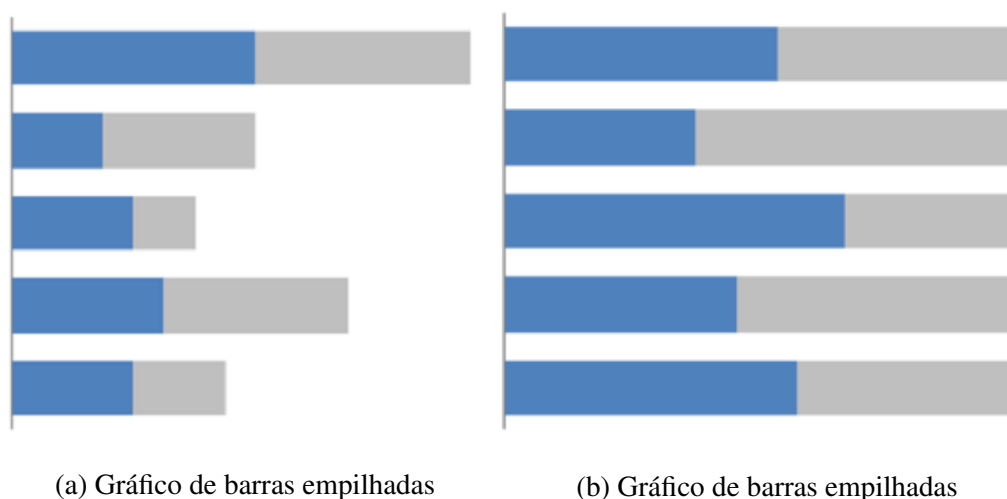
Gráficos de barras empilhadas costumam ser utilizados para a comparação de totais entre categorias e também para visualizar subcomponentes dentro de uma categoria.

Figura 7 – Gráficos de barras



Fonte: <https://www.storytellingwithdata.com/chart-guide>. Acesso em: 28 abr. 2021

Figura 8 – Gráficos de barras empilhadas

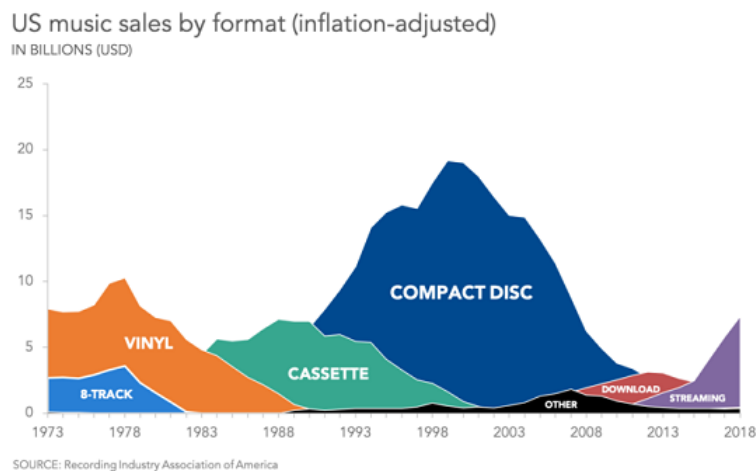


Fonte: <https://www.storytellingwithdata.com/chart-guide>. Acesso em: 28 abr. 2021

### 3.1.4 Gráfico de área

Os gráficos de área são semelhantes aos gráficos de linha, conectando pontos através de linhas contínuas, com a diferença de que a área abaixo das linhas é preenchida com alguma cor. Geralmente são utilizados quando o objetivo é destacar quedas e aumentos de várias séries ao longo do tempo. A figura 9 exemplifica um gráfico de área utilizado para exibir as vendas de música por formato de dispositivo nos Estados Unidos numa faixa de tempo.

Figura 9 – Gráfico de área



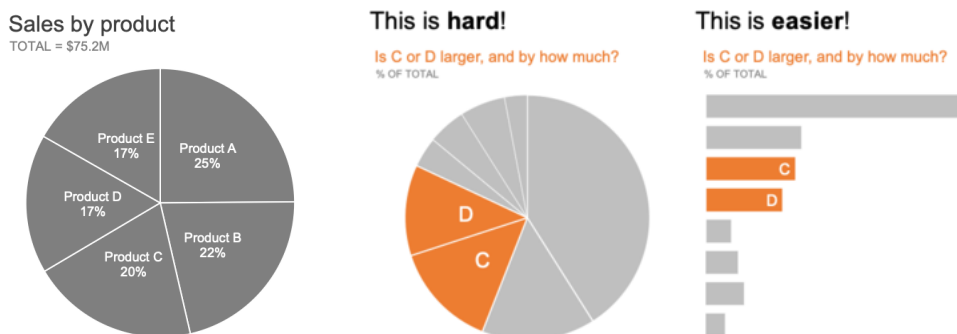
Fonte: <https://www.storytellingwithdata.com/chart-guide>. Acesso em: 28 abr. 2021

### 3.1.5 Gráfico de pizza

Os gráficos de pizza são utilizados para ilustrar dados categóricos em forma de uma pizza dividida em fatias, cada uma representando uma categoria com tamanho proporcional ao total dos valores utilizados. Segundo LIMA (2017 apud KNAFLIC, 2019), círculos são um símbolo universal de perfeição, o que torna agradável sua visualização. A figura 10a exemplifica o uso de um gráfico de pizza utilizado para exibir as vendas agrupadas por produtos.

Porém, segundo KNAFLIC (2019), gráficos de pizza são ruins e devem ser evitados por um simples motivo: a dificuldade em distinguir qual fatia é maior. Como alternativa, é recomendado o uso de gráficos de barra como mostra a figura 10b. Pelo mesmo motivo, gráficos de pizza em 3D também devem ser evitados.

Figura 10 – Exemplo de gráfico de pizza e sua comparação com um gráfico de barras



(a) Gráfico de pizza

(b) Comparação entre gráfico de pizza e gráfico de barras

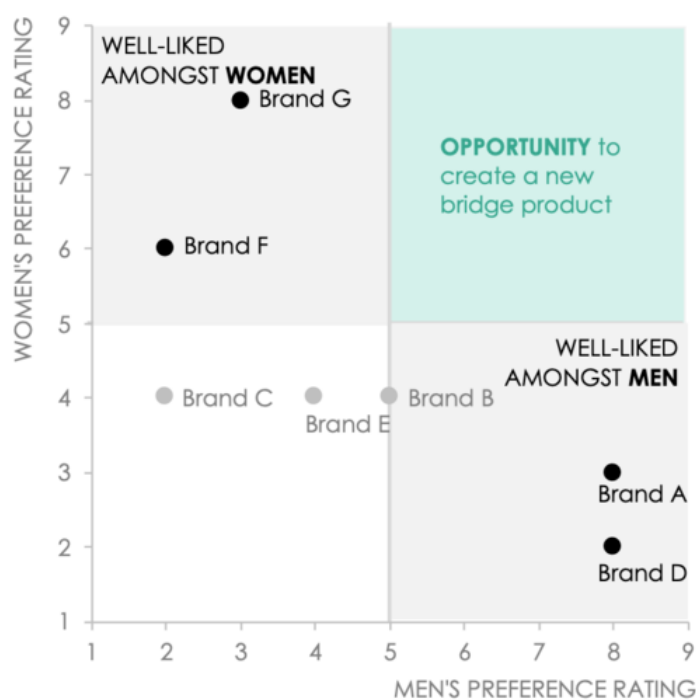
Fonte: <https://www.storytellingwithdata.com/chart-guide>. Acesso em: 28 abr. 2021

### 3.1.6 Gráfico de dispersão

Gráficos de dispersão são recomendados para exibir a relação entre dois tipos de dados, exibindo-os em um eixo horizontal e um eixo vertical para facilitar a visualização de uma possível relação. São comumente utilizados no meio científico, porém também são utilizados em empresas. A figura 11 exemplifica um gráfico de dispersão mostrando a relação entre o cuidado dos lábios entre homens e mulheres.

Figura 11 – Gráfico de dispersão

Lip care is polarizing between men and women



Fonte: <https://www.storytellingwithdata.com/chart-guide>. Acesso em: 28 abr. 2021

### 3.1.7 Tabela

Apesar de simples, tabelas apresentam os dados de forma direta e detalhada através de linhas e colunas, podendo ser utilizadas para inúmeros casos como, por exemplo, valor de vendas de cada produto ao longo de cada mês. Porém, segundo KNAFLIC (2019), deve-se ter cuidado com pequenos detalhes como evitar bordas e cores escuras para não desfocar a atenção do público aos dados. Para destacar os dados apresentados, pode-se colorir as células em tons de uma mesma cor de forma que os valores mais importantes fiquem com um tom mais forte e o valores menos importantes fiquem com tons mais fracos. Quando uma tabela utiliza essa forma de coloração, define-se ela como um mapa de calor. A figura 12 exemplifica uma tabela e um mapa de calor, respectivamente.



Figura 12 – Exemplo de tabela e mapa de calor

	A	B	C
Category 1	15%	22%	42%
Category 2	40%	36%	20%
Category 3	35%	17%	34%
Category 4	30%	29%	26%
Category 5	55%	30%	58%
Category 6	11%	25%	49%

(a) Tabela

	A	B	C
Category 1	15%	22%	42%
Category 2	40%	36%	20%
Category 3	35%	17%	34%
Category 4	30%	29%	26%
Category 5	55%	30%	58%
Category 6	11%	25%	49%

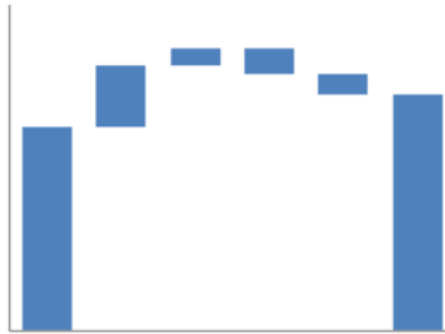
(b) Mapa de calor

Fonte: <https://www.storytellingwithdata.com/chart-guide>. Acesso em: 28 abr. 2021

### 3.1.8 Gráfico de cascata

Os gráficos de cascata são utilizados para fracionar as partes de um gráfico de barras empilhadas para destacar o ponto de partida, os aumentos ou reduções e o ponto final. Um exemplo de uso é na análise de crescimento populacional de animais. A figura 13 ilustra um gráfico de cascata.

Figura 13 – Gráfico de cascata



Fonte: <https://www.storytellingwithdata.com/chart-guide>. Acesso em: 28 abr. 2021

## 3.2 DASHBOARDS

*Dashboard* é um painel visual para o fornecimento de visualização rápida de indicadores de desempenho através de elementos como gráficos e tabelas (MALIK, 2005). Através de um *dashboard*, todo o volume de dados extraído de um sistema de informação é resumido em um conjunto de gráficos e tabelas para facilitar e agilizar a compreensão. Segundo RASMUSSEN, BANSAL & CHEN (2013), o uso de *dashboards* implica nos seguintes benefícios:

- Facilidade em identificar e corrigir tendências negativas.
- Facilidade em tomar decisões melhores para o negócio.
- Facilidade em perceber a medição de eficiência e ineficiência do negócio.
- Facilidade em alinhar estratégias e objetivos.
- Aumento da produtividade dos funcionários.
- Economia de tempo dos funcionários devido à eliminação da necessidade de produzir muitos relatórios e, por consequência, aumento do tempo disponível para a análise dos dados.
- Poucos requisitos para treinamentos, pois é fácil de compreender.
- Funcionários conseguem gerar relatórios detalhados com novas perspectivas.

### 3.2.1 Tipos de *dashboards*

Segundo RASMUSSEN, BANSAL & CHEN (2013), há três principais tipos de *dashboards*:

- *Dashboard* estratégico: contém indicadores essenciais para a validação da estratégia de negócio a longo prazo, a fim de facilitar a compreensão da eficiência da estratégia em questão comparada a métricas referentes ao negócio. Como exemplo, pode-se considerar a aplicação para métodos OKR (*Objectives and Key Results*), que visam a definição e acompanhamento de metas.
- *Dashboard* tático: contém indicadores que contribuem para a tomada de decisão a médio prazo. Por exemplo, pode-se aplicar para o acompanhamento de projeção mensal de vendas para alertar aos gestores a possibilidade ou impossibilidade de atingir a meta de vendas daquele mês.
- *Dashboard* operacional: contém indicadores que apresentam métricas referentes às operações do negócio a fim de auxiliar os analistas a identificar e corrigir falhas nos processos operacionais do negócio. Por exemplo, em uma empresa de transportes, apresentaria métricas referentes às entregas realizadas para mensuração de entregas atrasadas ou extravaiadas.

### 3.2.2 Construção de *dashboards*

Segundo KNAFLIC (2019), deve-se considerar os seguintes pontos para a construção de uma solução de visualização de dados:

- Entender o contexto: identificar quem é o público alvo, de qual informação esse público precisa, e como exibir essa informação. Por exemplo, um *dashboard* de vendas para uma empresa do ramo de alimentos, o público alvo seria o setor comercial; a informação seria o desempenho de vendas por produto, que seria exibida através do faturamento mensal referente a cada produto ao longo do ano.
- Escolher um visual apropriado: selecionar os gráficos adequados, com base nas informações que serão exibidas.
- Eliminar a saturação: desenfaturar o título do gráfico, dando ênfase para o gráfico em si; remover bordas e linhas do gráfico; desenfaturar linhas e legendas dos eixos x e y; remover a variação nas cores, escolhendo a mesma cor e variando apenas o tom; legendar as linhas diretamente para eliminar o esforço dos olhos de ir e vir entre as legendas e os dados.
- Destacar o que é mais importante: destacar através de cores fortes e espaçamento os dados que necessitam de mais atenção como, por exemplo, informações recentes ou que apontam falhas.

- Contar uma história: escolher quais dados serão necessários e exibi-los de forma adequada para que o público alvo entenda a situação em detalhes para posteriormente tomar uma decisão melhor.

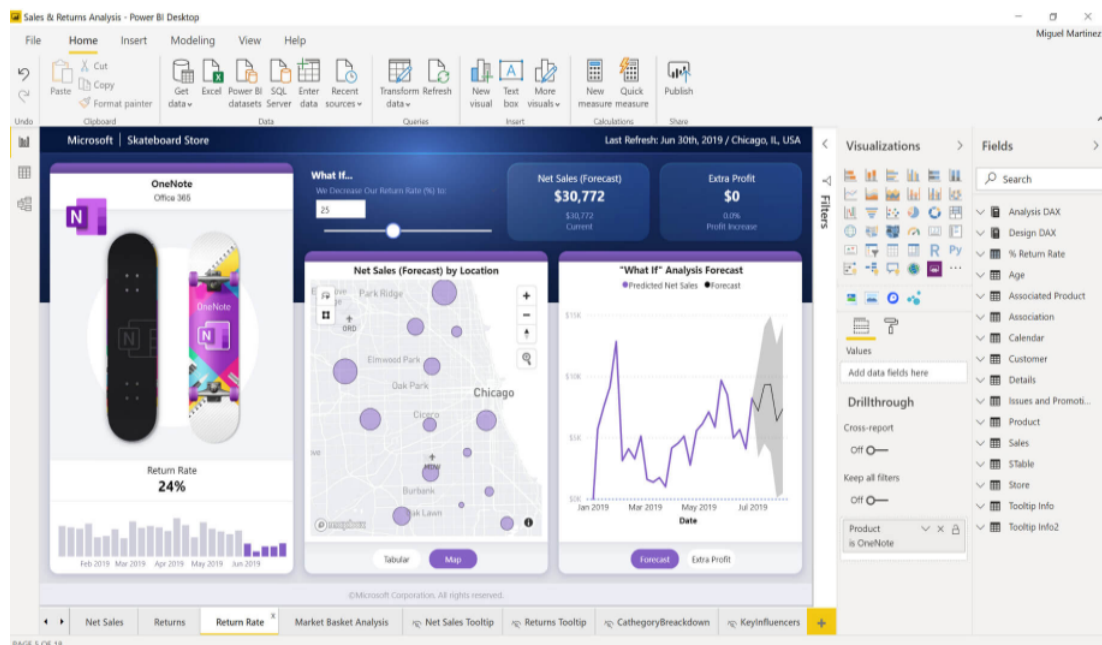
## 4 FERRAMENTAS DE VISUALIZAÇÃO

Com o crescimento da demanda por soluções de análise de dados, empresas desenvolveram e disponibilizaram ferramentas para facilitar o desenvolvimento de interfaces gráficas para visualização de dados através de gráficos e tabelas.

### 4.1 POWER BI

O Power BI<sup>1</sup> é um conjunto de serviços de software, aplicativos e conectores para análise de dados lançado em 2015 pela empresa estadunidense Microsoft Corporation. Através das soluções oferecidas pelo Power BI, é possível realizar integrações com diversas fontes de dados como, por exemplo, planilhas eletrônicas, bancos de dados e serviços de nuvem para extrair dados e com eles construir *dashboards* com gráficos e tabelas interativas. A versão básica do Power BI é gratuita, porém a versão profissional, que conta com mais recursos, custa 9,99 dólares mensais por usuário; e a versão *premium*, recomendada para empresas maiores, custa 20,00 dólares mensais por usuário. A Figura 14 mostra a interface gráfica do Power BI para criação de *dashboards*.

Figura 14 – Interface gráfica do Power BI para criação de *dashboards*



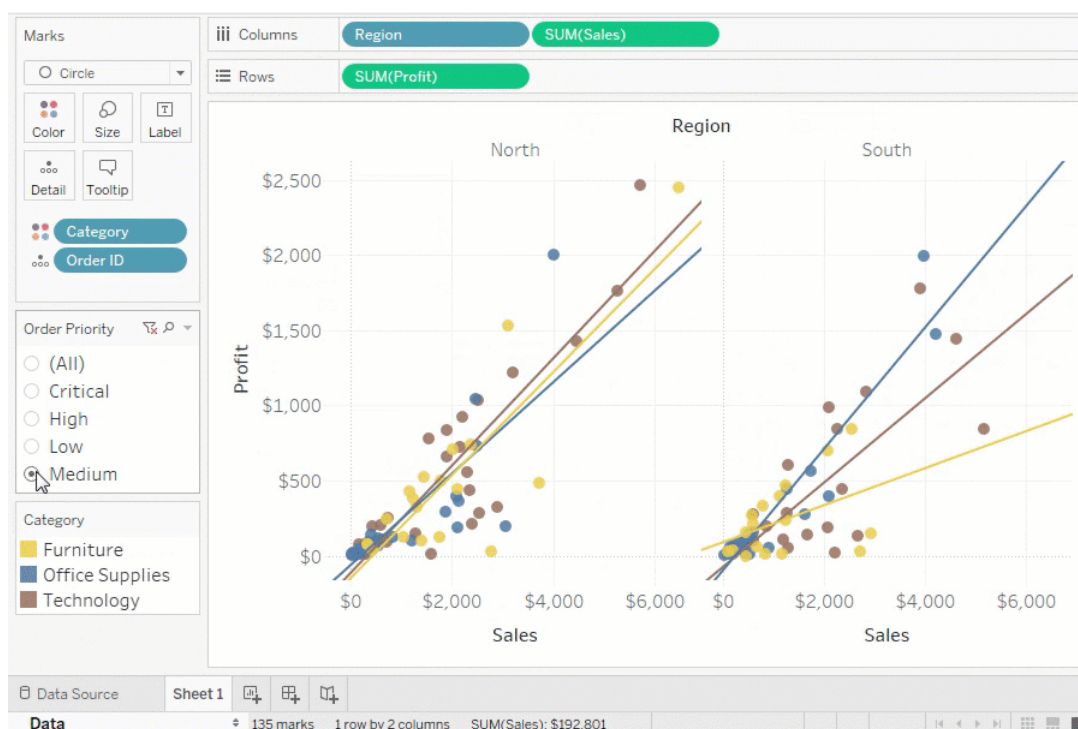
Fonte: <https://powerbi.microsoft.com/pt-br/>. Acesso em: 16 jun. 2021

<sup>1</sup> <https://powerbi.microsoft.com/pt-br/>. Acesso em: 28 abr. 2021

## 4.2 TABLEAU

O Tableau<sup>2</sup> é uma plataforma desenvolvida pela empresa estadunidense Tableau Software para análise visual de dados. Assim como o Power BI, através da plataforma Tableau é possível realizar integrações com fontes de dados como, por exemplo, bancos de dados e serviços em nuvens para extrair dados e construir *dashboards*. Informações de planos e preços não estão disponíveis no *website* da plataforma, devendo serem obtidas com a equipe comercial da Tableau. Há uma versão gratuita, porém é disponibilizada apenas para projetos de acesso público. A Figura 15 mostra a interface gráfica do Tableau para criação de *dashboards*.

Figura 15 – Interface gráfica do Tableau para criação de *dashboards*



Fonte: <https://www.tableau.com/pt-br/>. Acesso em: 16 jun. 2021

## 4.3 HIGHCHARTS

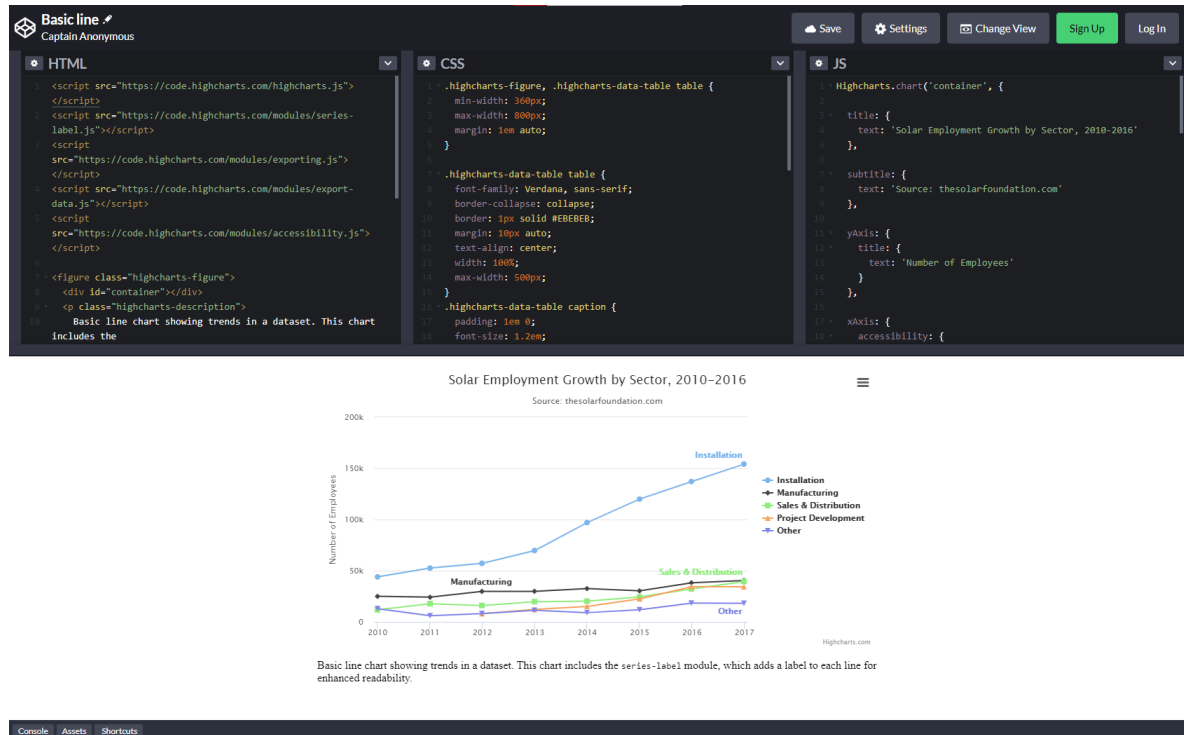
Highcharts<sup>3</sup> é uma biblioteca escrita em JavaScript e lançada em 2009 para a construção de gráficos. Ao contrário de ferramentas como Power BI e Tableau, o Highcharts não oferece a possibilidade de integração com fontes de dados. Para a construção dos gráficos, deve-se utilizar linguagens de programação e marcação para o desenvolvimento da interface gráfica e carregamento dos dados. Recursos simples da biblioteca são disponibilizados gratuitamente, porém a licença para obter todos recursos está disponível por 535 dólares. A Figura 16 mostra um exemplo de código-fonte nas linguagens de marcação HTML (*HyperText Markup Language*), de

<sup>2</sup> <https://www.tableau.com/pt-br/>. Acesso em: 28 abr. 2021

<sup>3</sup> <https://www.highcharts.com/>. Acesso em: 28 abr. 2021

estilo CSS (*Cascading Style Sheets*) e de programação JavaScript para a criação de *dashboards* utilizando a biblioteca Highcharts.

Figura 16 – Exemplo de código-fonte usando a biblioteca Highcharts para criação de *dashboards*



Fonte: <https://www.highcharts.com/demo/line-basic>. Acesso em: 16 jun. 2021

Dentre as soluções pesquisadas, ferramentas de visualização de dados como PowerBI e Tableau mostraram-se melhores que bibliotecas de linguagens de programação devido à facilidade de uso que oferecem, juntamente com recursos sofisticados que diminuem o esforço necessário e aumentam a produtividade.

## 5 PROPOSTA DE SOLUÇÃO

A proposta de solução para este trabalho, definida com base no diagrama ER referente aos dados provenientes do uso de veículos de transporte de carga e disponibilizados por uma empresa, e com base no referencial teórico citado nos capítulos anteriores, é a construção de *dashboards* utilizando-se a ferramenta PowerBI, escolhida devido à qualidade e custo baixo, que consultará um *data warehouse* que será construído para essa finalidade. Como a empresa disponibiliza diversos gráficos, visando abranger vários itens que possam ser do interesse de seus usuários sem uma demanda específica, a proposta de *dashboards* contemplará diferentes indicadores, procurando exibir um conjunto significativo de informações. O acesso ao banco de dados será fornecido pela empresa apenas no segundo semestre do ano 2021. A partir desse acesso, será possível definir o processo de ETL para construção do *data warehouse*.

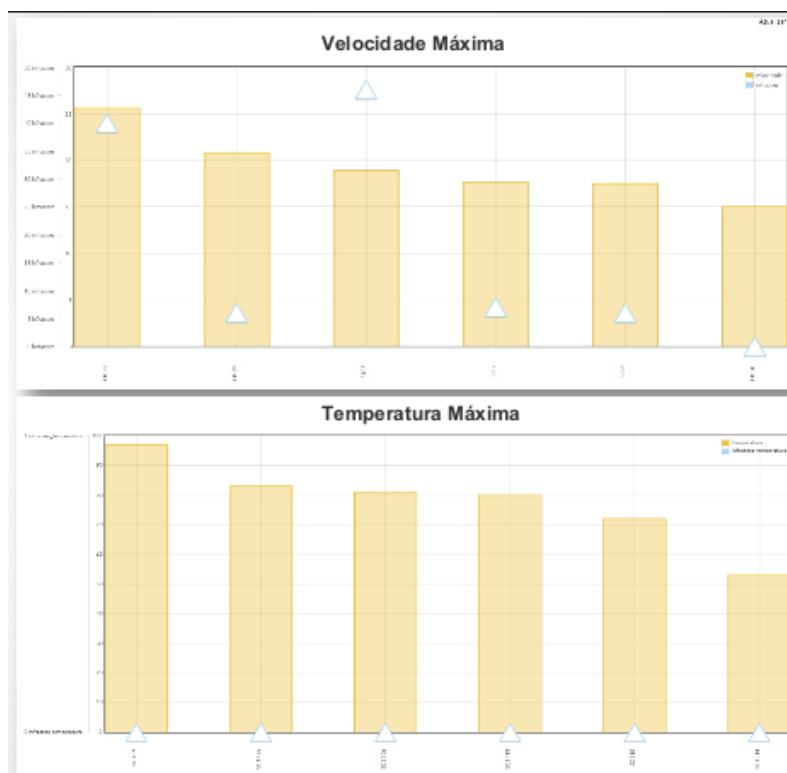
### 5.1 DADOS

Dentre os diversos dados produzidos pela empresa em questão, destacam-se os dados provenientes do uso de veículos de transporte de carga que contém seus acessórios. Os dados produzidos são armazenados em um banco de dados MariaDB e aqueles que foram disponibilizados pela empresa para este trabalho estão apresentados na Figura 62 localizada no Apêndice A.

Os dados armazenados que serão utilizados para este trabalho, cujas tabelas estão destacadas em vermelho, fornecem as seguintes informações operacionais dos veículos, coletadas a cada 10 minutos ou quando o evento ocorre (neste caso, as infrações): tempo trabalhado, velocidade máxima, velocidade momentânea, aceleração, desaceleração, tempo com carga, temperatura máxima, temperatura momentânea, horímetro, odômetro, rotação, velocidade de reversão, impacto, posição e estado da máquina. Tais dados foram escolhidos devido à importância para o monitoramento da integridade dos veículos monitorados pela empresa em questão. Atualmente, esses dados já são exibidos em um sistema *web* dessa empresa, porém em forma de gráficos avulsos simples, um (ou dois) por tela, gerados através da linguagem de programação PHP (*Hypertext Preprocessor*). A Figura 17 mostra exemplos de gráficos do sistema *web* da empresa fornecedora dos dados.



Figura 17 – Exemplo de gráficos do sistema *web* da empresa fornecedora dos dados



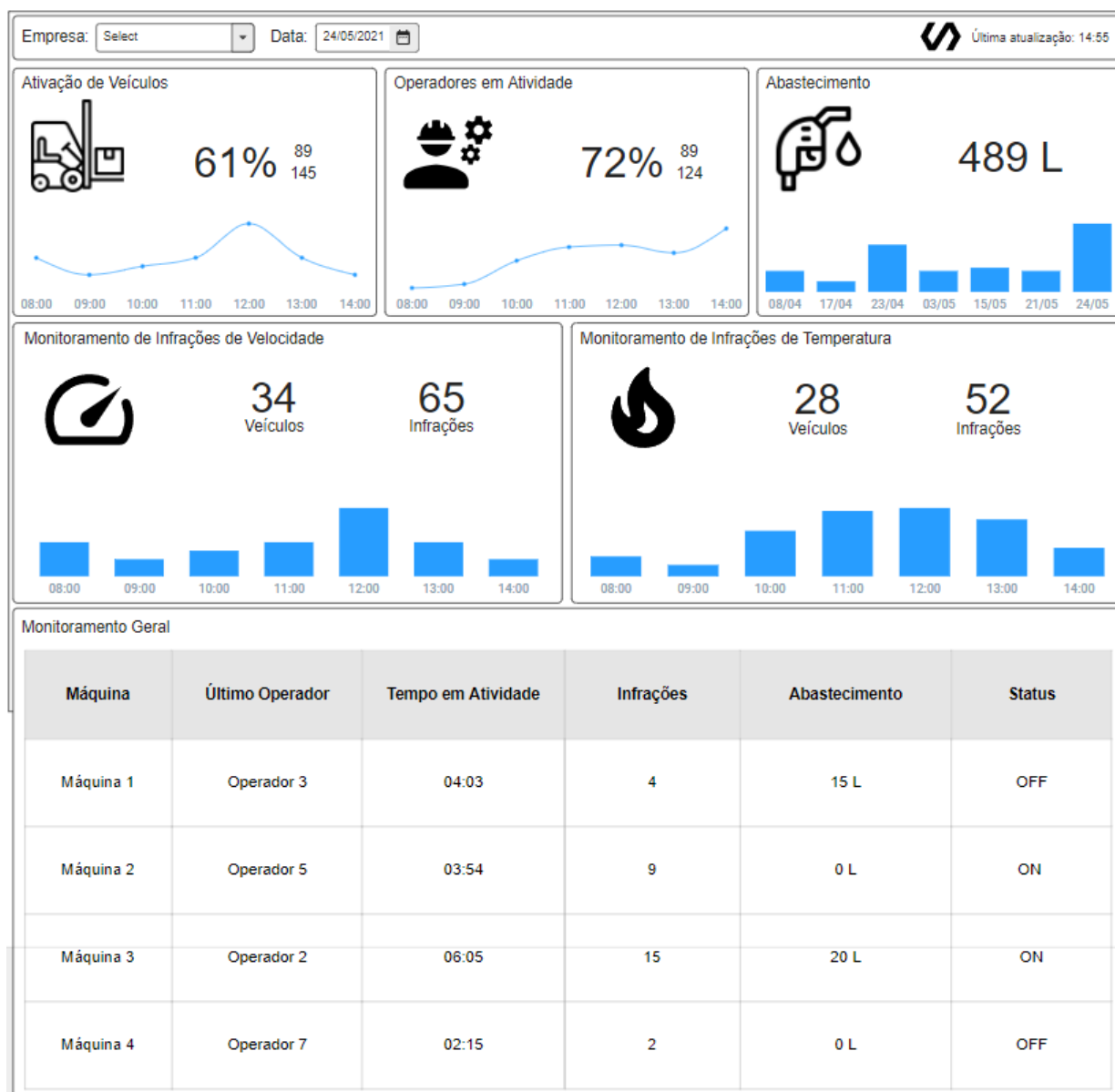
Fonte: Empresa fornecedora dos dados (2021)

## 5.2 MOCKUP DOS DASHBOARDS

Com o conhecimento de quais dados serão disponibilizados foi elaborado dois esboços de *dashboards* operacionais referentes aos dados dos veículos de transporte conforme mostra a Figura 18, para um *dashboard* geral de todas máquinas da empresa e a Figura 26, para um *dashboard* específico por máquina, que serão implementados utilizando a ferramenta Power BI devido à praticidade e diversidade de recursos oferecidos e o baixo custo.

## 5.2.1 Dashboard geral de veículos

Figura 18 – Esboço do *dashboard* geral



Fonte: O Autor (2021)

A estrutura do *dashboard* específico começa, conforme destacado na Figura 19, com uma caixa de filtros de empresa, referente aos clientes da empresa fornecedora dos dados; e dia, referente ao dia em que os dados foram coletados. Portanto, as informações apresentadas nos demais indicadores serão referentes aos dados coletados da máquina selecionada naquele dia selecionado. Na extremidade direita, é apresentado o logotipo da empresa selecionada no primeiro filtro e o último horário de atualização do *dashboard*.

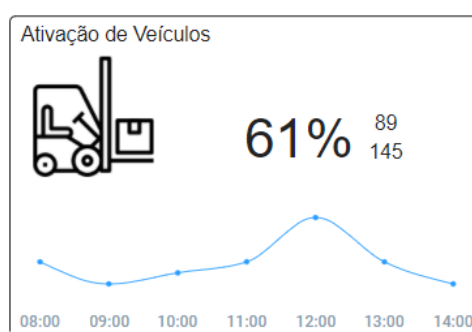
Figura 19 – Esboço da caixa de filtro



Fonte: O Autor (2021)

Logo abaixo, conforme destacado na Figura 20, encontra-se o indicador de ativação de veículos, que mostra: a quantidade de veículos em atividade no momento, a quantidade total de veículos da empresa, a proporção e a quantidade de veículos ativos ao longo do dia através de um gráfico de linhas.

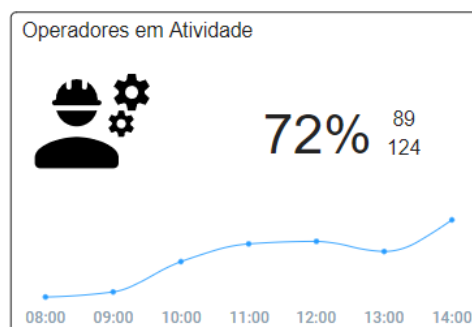
Figura 20 – Esboço do indicador de ativação de veículos



Fonte: O Autor (2021)

Ao lado do indicador de ativação de veículos, conforme destacado na Figura 21, encontra-se o indicador de operadores em atividade, que mostra: a quantidade de operadores em atividade no momento, a quantidade total de operadores da empresa, a proporção e a quantidade de operadores em atividade ao longo do dia através de um gráfico de linhas.

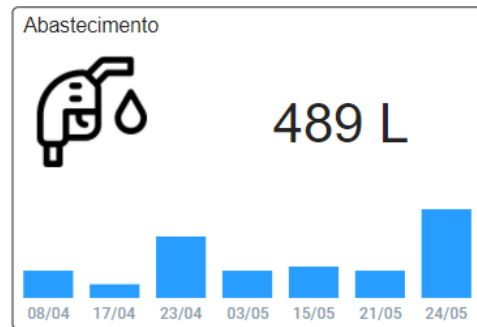
Figura 21 – Esboço do indicador de operadores em atividade



Fonte: O Autor (2021)

Ao lado direito do indicador de operadores em atividade, conforme destacado na Figura 22, encontra-se o indicador de abastecimento, que mostra: a quantidade de combustível abastecido em todos veículos naquele dia e a quantidade abastecida nos últimos dias através de um gráfico de barras verticais.

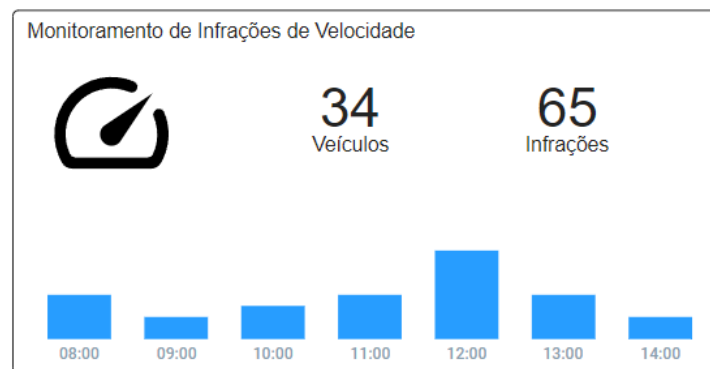
Figura 22 – Esboço do indicador de abastecimento geral



Fonte: O Autor (2021)

Abaixo dos três indicadores citados anteriormente, conforme destacado na Figura 23, encontra-se o indicador de monitoramento de infrações de velocidade, que mostra: a quantidade de veículos que cometeram infrações, a quantidade total de infrações e a quantidade de infrações a nível de hora através de um gráfico de barras verticais.

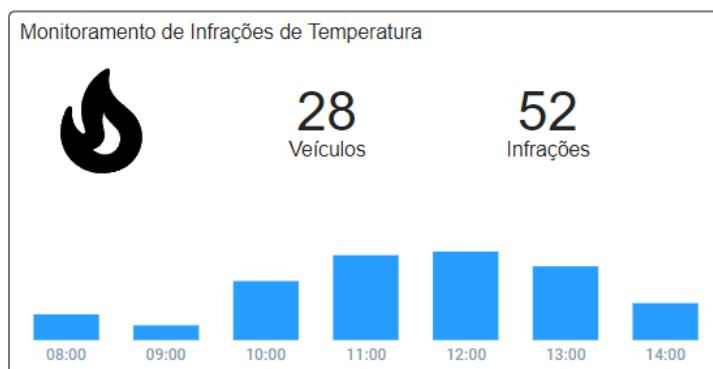
Figura 23 – Esboço do indicador de monitoramento de infrações de velocidade



Fonte: O Autor (2021)

Ao lado do indicador de monitoramento de infrações de velocidade, conforme destacado na Figura 24, encontra-se o indicador de monitoramento de infrações de temperatura, que mostra: a quantidade de veículos que cometeram infrações, a quantidade total de infrações e a quantidade de infrações a nível de hora através de um gráfico de barras verticais.

Figura 24 – Esboço do indicador de monitoramento de infrações de temperatura



Fonte: O Autor (2021)

Por fim, conforme destacado na Figura 25, encontra-se a tabela de monitoramento geral, que mostra os dados gerais por veículo: o nome do veículo, o nome do último operador, o tempo que o veículo esteve ligado até o momento naquele dia, a quantidade total de infrações registradas por aquele veículo, a quantidade de combustível abastecida no dia e o *status* atual (ligado/desligado).

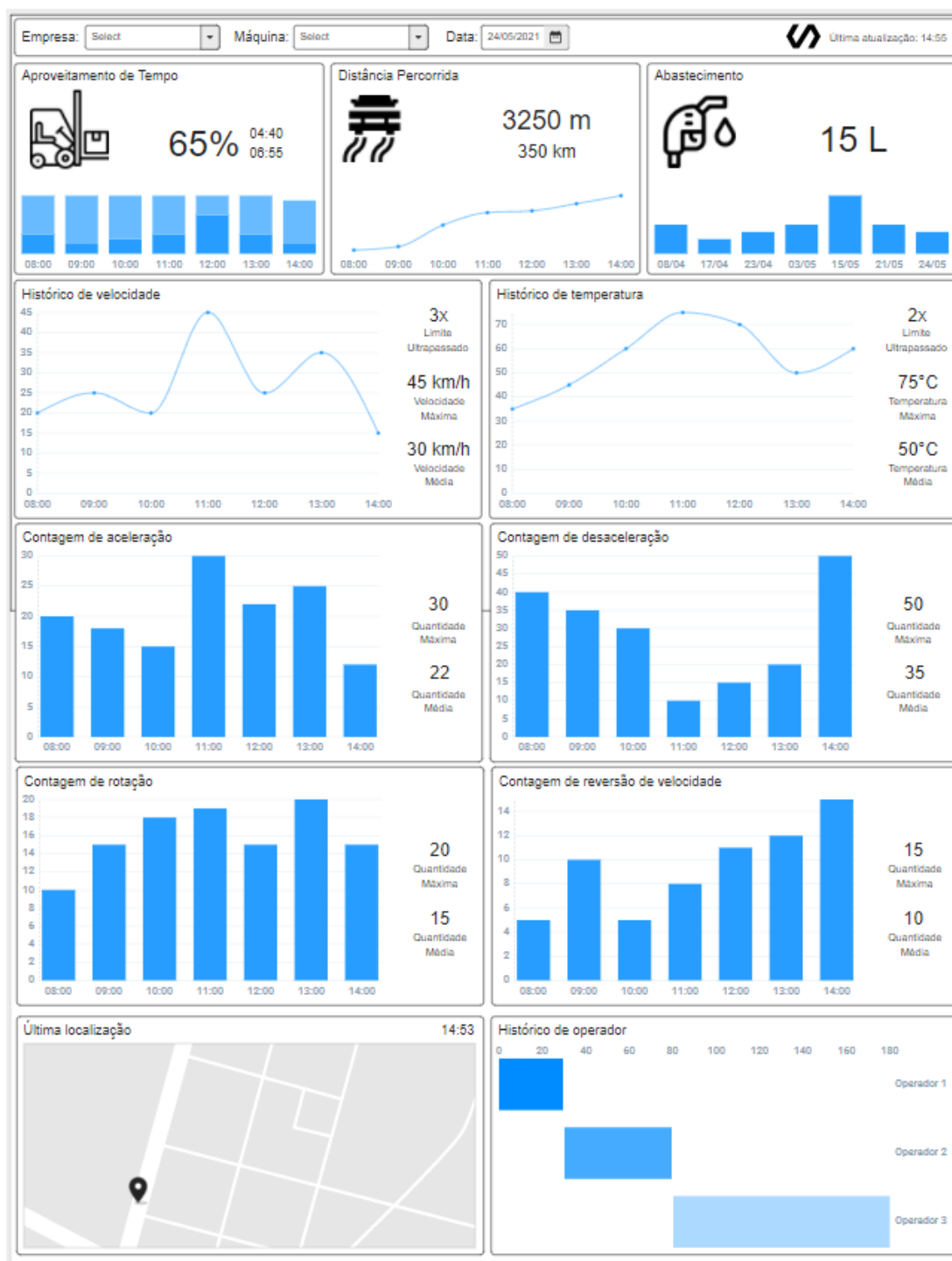
Figura 25 – Esboço da tabela de dados gerais

Máquina	Último Operador	Tempo em Atividade	Infrações	Abastecimento	Status
Máquina 1	Operador 3	04:03	4	15 L	OFF
Máquina 2	Operador 5	03:54	9	0 L	ON
Máquina 3	Operador 2	08:05	15	20 L	ON
Máquina 4	Operador 7	02:15	2	0 L	OFF

Fonte: O Autor (2021)

## 5.2.2 Dashboard específico por veículo

Figura 26 – Esboço do *dashboard* específico por veículo



Fonte: O Autor (2021)

A estrutura do *dashboard* específico começa, conforme destacado na Figura 27, com uma caixa de filtros de empresa, referente aos clientes da empresa fornecedora dos dados; máquina, referente aos veículos de carga; e dia, referente ao dia em que os dados foram coletados.

Portanto, as informações apresentadas nos demais indicadores serão referentes aos dados coletados da máquina selecionada naquele dia selecionado. Na extremidade direita, é apresentado o logotipo da empresa selecionada no primeiro filtro e o último horário de atualização do *dashboard*.

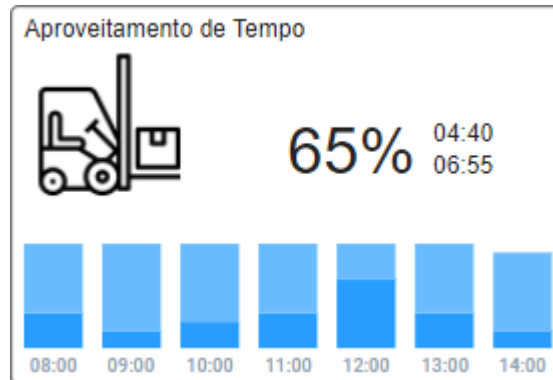
Figura 27 – Esboço da caixa de filtro



Fonte: O Autor (2021)

Logo abaixo do filtro, tem-se uma linha contendo três principais indicadores com a informação resumida. O primeiro, destacado na Figura 28, mostra: a quantidade de horas e minutos em que a máquina esteve com carga no dia selecionado, ou seja, transportando uma mercadoria; a quantidade de horas e minutos do início do expediente até o último momento disponível naquele dia; a proporção em porcentagem referente aos dois valores citados anteriormente; e o detalhamento das informações a nível de hora através de um gráfico de barras empilhadas.

Figura 28 – Esboço do indicador de aproveitamento de tempo



Fonte: O Autor (2021)

O segundo, destacado na Figura 29, mostra: a quantidade de metros percorridos pela máquina até o último momento disponível do dia selecionado; a quantidade de quilômetros que a máquina percorreu desde o início de seu uso até o último momento disponível; e o detalhamento da distância percorrida ao longo do dia através de um gráfico de linhas.

Figura 29 – Esboço do indicador de distância percorrida



Fonte: O Autor (2021)

O terceiro, destacado na Figura 30, mostra: a quantidade de combustível em litros abastecida por aquela máquina no dia selecionado e um gráfico de barras mostrando a quantidade nos últimos abastecimentos.

Figura 30 – Esboço do indicador de abastecimento

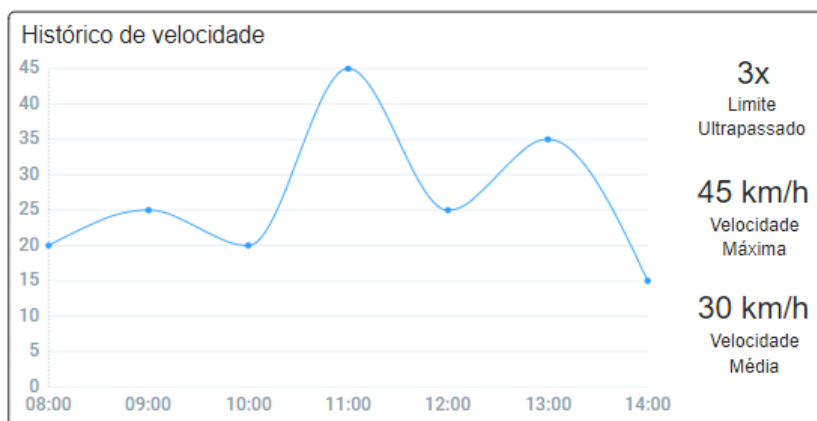


Fonte: O Autor (2021)

Na linha abaixo do primeiro trio de indicadores, tem-se dois indicadores. O primeiro, à esquerda, destacado na Figura 31, mostra: um gráfico de linhas mostrando a velocidade do veículo ao longo do dia; a quantidade de vezes em que o limite de velocidade foi ultrapassado até o último momento disponível; a velocidade máxima atingida e a velocidade média ao longo do dia.



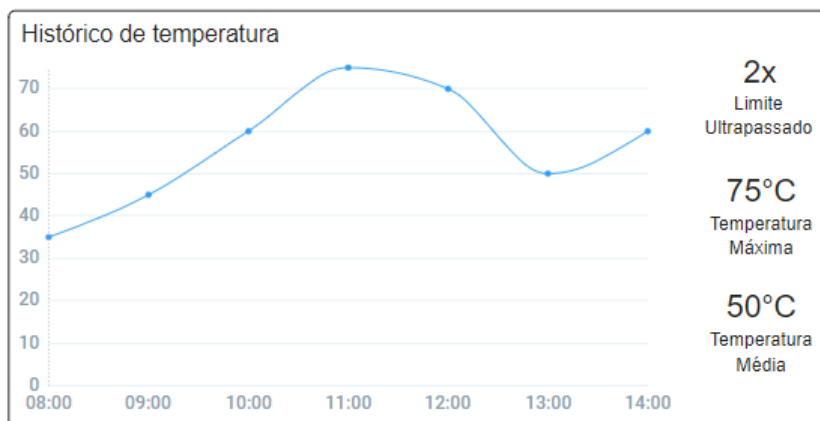
Figura 31 – Esboço do indicador de velocidade



Fonte: O Autor (2021)

O segundo, à direita, destacado na Figura 32, mostra: um gráfico de linhas mostrando a temperatura do veículo ao longo do dia; a quantidade de vezes em que o limite de temperatura foi ultrapassado; a temperatura máxima atingida até o último momento disponível e a temperatura média ao longo do dia.

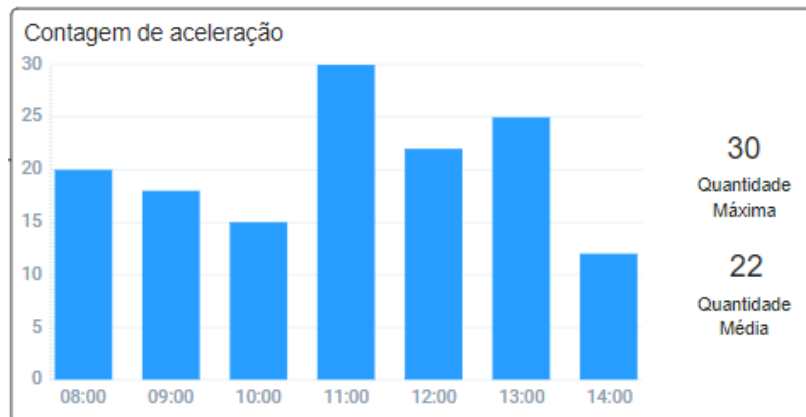
Figura 32 – Esboço do indicador de temperatura



Fonte: O Autor (2021)

Na linha abaixo dos indicadores de velocidade e temperatura, tem-se também dois indicadores. O primeiro, à esquerda, destacado na Figura 33, mostra: um gráfico de barras mostrando a quantidade de acelerações por hora ao longo do dia; a quantidade máxima de acelerações por hora e a quantidade média de acelerações ao longo do dia.

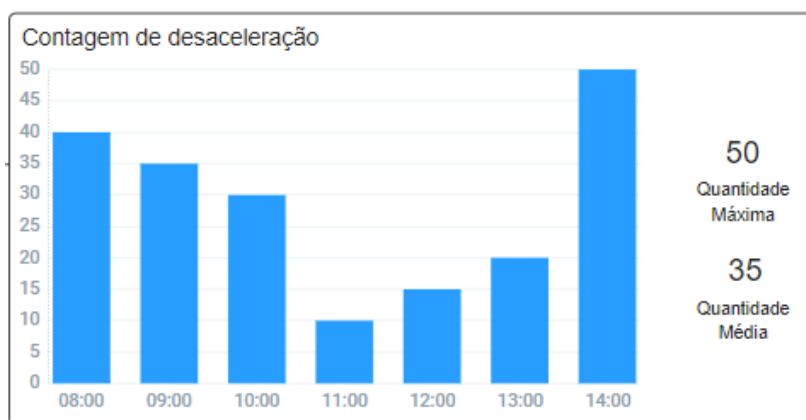
Figura 33 – Esboço do indicador de aceleração



Fonte: O Autor (2021)

O segundo, à direita, destacado na Figura 34, mostra: um gráfico de barras mostrando a quantidade de desacelerações por hora ao longo do dia; a quantidade máxima de desacelerações por hora e a quantidade média de desacelerações ao longo do dia.

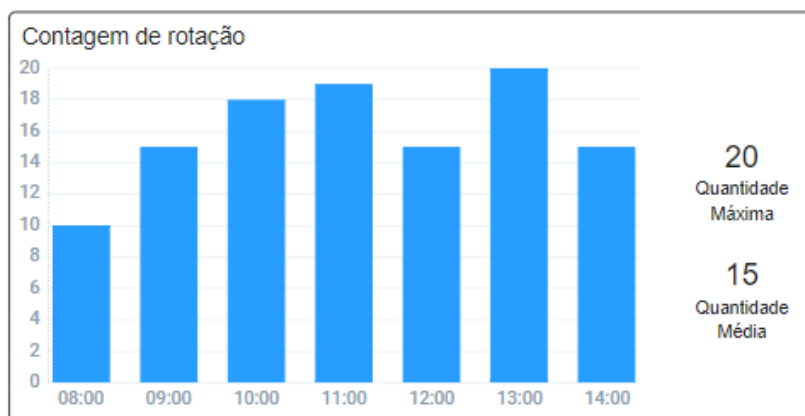
Figura 34 – Esboço do indicador de desaceleração



Fonte: O Autor (2021)

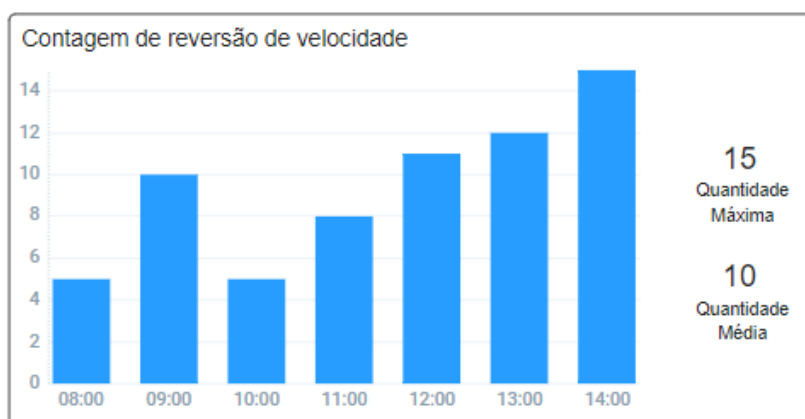
Na penúltima linha, tem-se a mesma situação da linha anterior, porém com dados de rotação no lado esquerdo, conforme destacado na Figura 35, e reversão de velocidade no lado direito, conforme destacado na Figura 36.

Figura 35 – Esboço do indicador de rotação



Fonte: O Autor (2021)

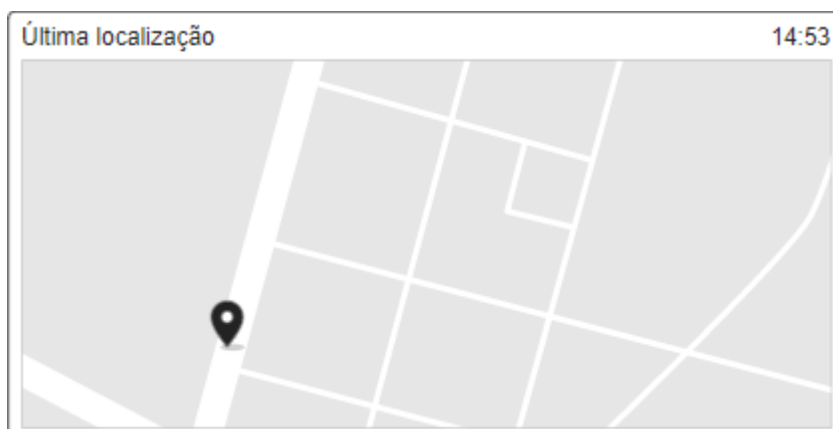
Figura 36 – Esboço do indicador de reversão de velocidade



Fonte: O Autor (2021)

Na última linha, tem-se dois indicadores. O primeiro, à esquerda, destacado na Figura 37, mostra: um mapa contendo a ilustração de um alfinete indicando a última localização disponível do veículo naquele dia.

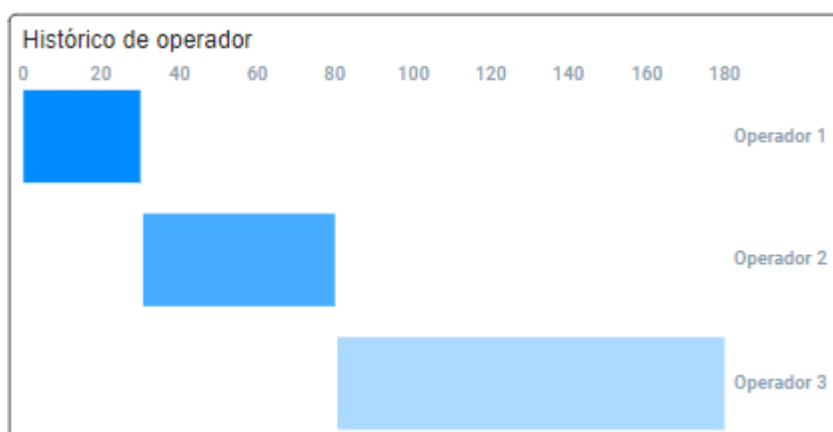
Figura 37 – Esboço do indicador de localização



Fonte: O Autor (2021)

O segundo, à direita, destacado na Figura 38, mostra: um gráfico de barras que indica os operadores que utilizaram aquela máquina ao longo do dia.

Figura 38 – Esboço do indicador de histórico de operador



Fonte: O Autor (2021)

### 5.3 REESTRUTURAÇÃO DOS *DASHBOARDS*

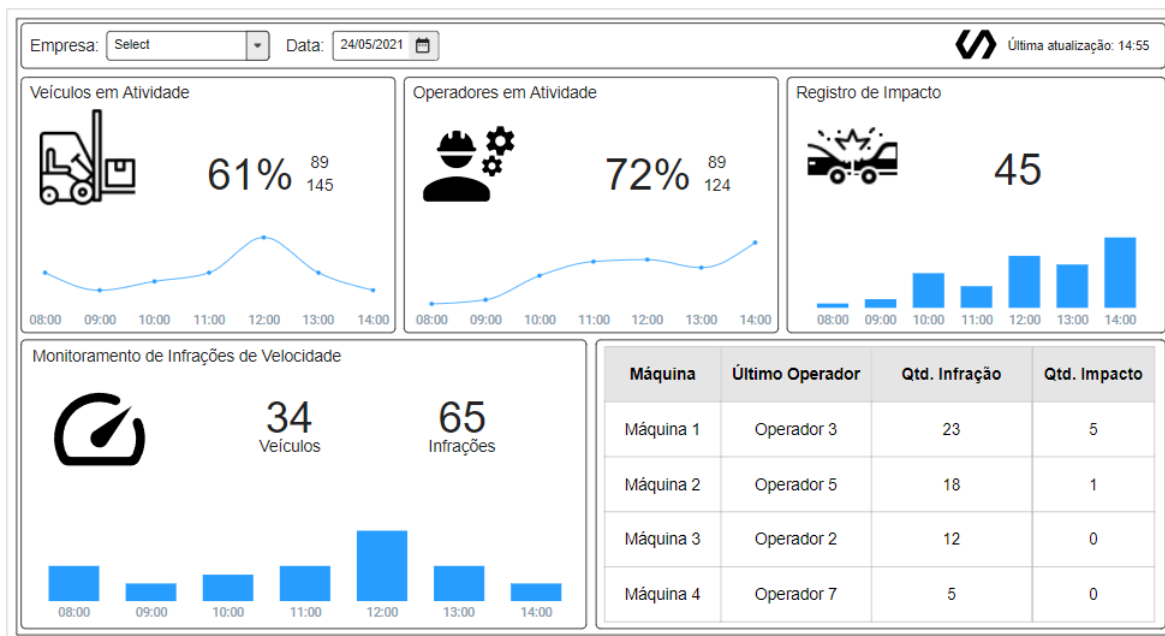
Como descrito anteriormente, a proposta foi elaborada com base no diagrama ER do banco de dados e o acesso à base seria fornecido apenas no segundo semestre do ano.

Chegado o segundo semestre, a empresa, por questões de segurança, optou por fornecer os dados através de planilhas em vez de fornecer acesso ao banco de dados. Com o recebimento das planilhas, foi constatado que muitos dados citados no diagrama ER não existem ainda devido a limitações do sistema da empresa que coleta os dados. Para contornar essa falta de dados, foi necessário refazer os esboços dos *dashboards*.

### 5.3.1 Novo *dashboard* geral de veículos

O *dashboard* geral de veículos foi reestruturado conforme a Figura 42, focando na exibição de informações referentes à velocidade e impacto dos veículos.

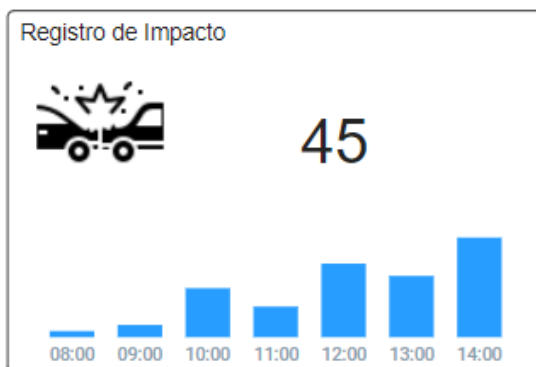
Figura 39 – Esboço do novo *dashboard* geral de veículos



Fonte: O Autor (2021)

Segundo o gestor da empresa, não há dados referentes ao abastecimento. O módulo do sistema que realizaria a coleta foi desenvolvido para apenas dois clientes, porém eles não fizeram uso. Para substituir o indicador de abastecimento, foi elaborado um indicador de registro de impacto, conforme mostra a Figura 40. No *label* é mostrado a quantidade total de impactos ocorridos no dia filtrado, tendo o detalhamento a nível de hora mostrado num gráfico de barras verticais.

Figura 40 – Esboço do indicador de registro de impacto



Fonte: O Autor (2021)

O indicador de monitoramento de infrações de temperatura foi removido porque o gestor da empresa informou que não são registradas as infrações de temperatura. Para substituí-lo, foi realocada a tabela de dados gerais, porém reduzindo a quantidade de colunas para exibir, conforme a Figura 41, somente o nome do veículo, o nome do operador, a quantidade de infrações de velocidade e a quantidade de infrações de impacto, tornando, assim, o *dashboard* focado no monitoramento do excesso de velocidade e colisões dos veículos.

Figura 41 – Esboço da nova tabela de dados gerais

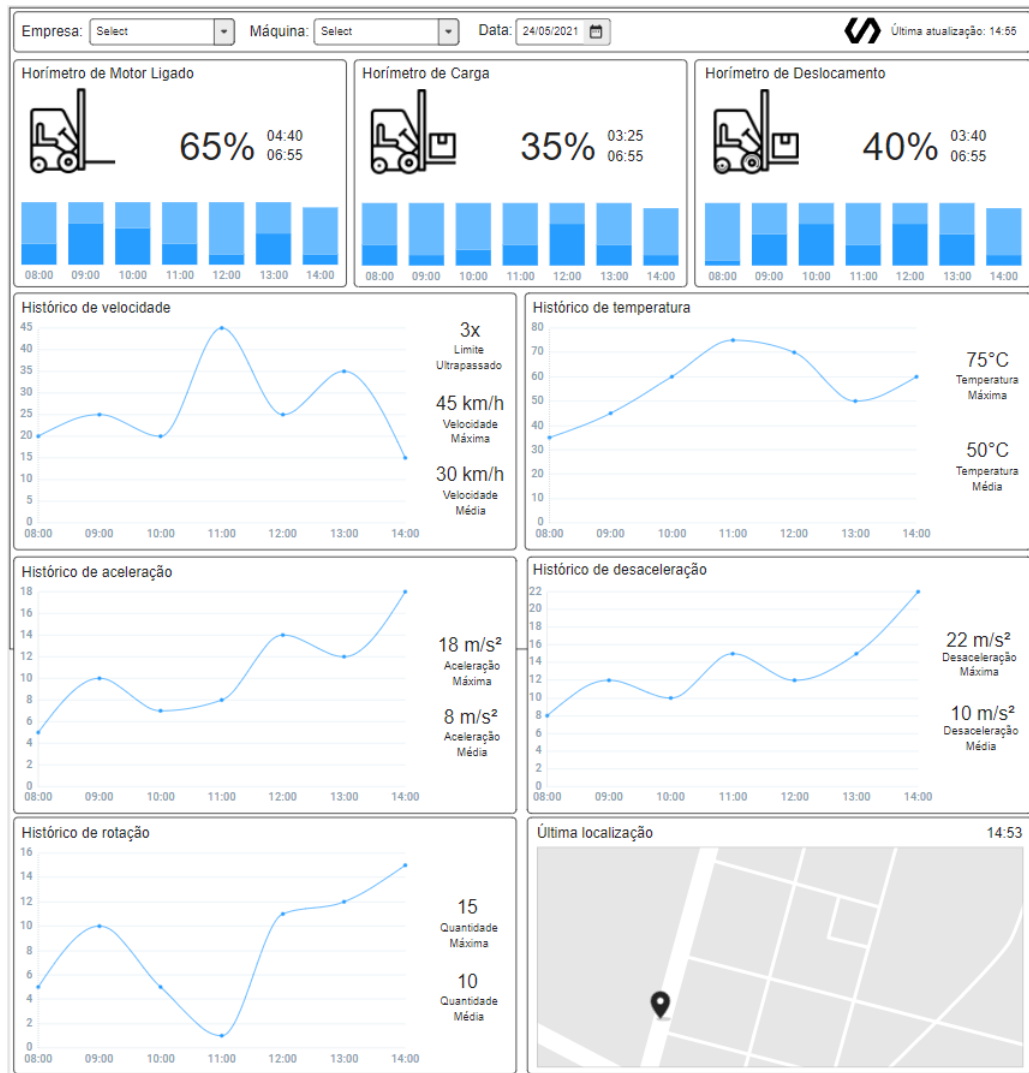
Máquina	Último Operador	Qtd. Infração	Qtd. Impacto
Máquina 1	Operador 3	23	5
Máquina 2	Operador 5	18	1
Máquina 3	Operador 2	12	0
Máquina 4	Operador 7	5	0

Fonte: O Autor (2021)

### 5.3.2 Novo *dashboard* específico por veículo

O *dashboard* específico por veículo foi reestruturado conforme a Figura 42, focando na exibição de informações referentes ao tempo de uso e às características físicas do uso de veículos como velocidade, temperatura, aceleração, desaceleração, rotação, e a localização do veículo.

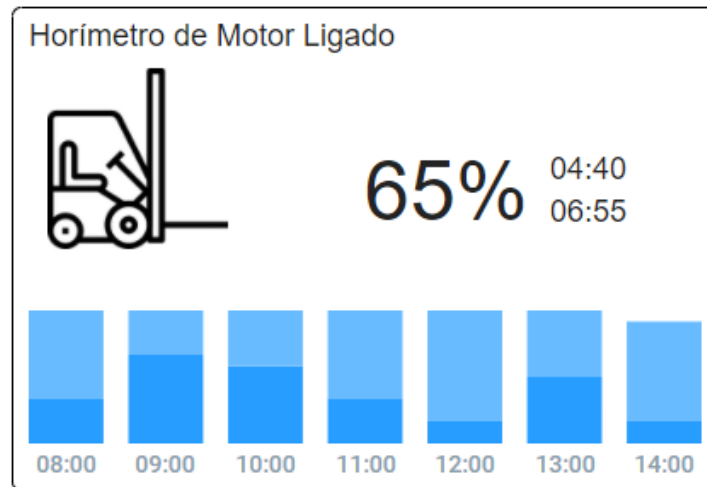
Figura 42 – Esboço do novo *dashboard* específico de veículos



Fonte: O Autor (2021)

Conforme descrito anteriormente, o sistema da empresa em questão não coleta os dados de abastecimento, logo no segundo *dashboard* o indicador de abastecimento também foi removido. Para a quilometragem, o gestor da empresa informou que apenas um cliente registra a coleta, então o indicador de distância percorrida também foi removido. Assim, para preencher o espaço desses indicadores, o indicador de aproveitamento de tempo foi dividido em três indicadores: horímetro de motor ligado, horímetro de carga e horímetro de deslocamento. O horímetro de motor ligado, conforme mostra a Figura 43, segue o mesmo modelo do indicador de aproveitamento de tempo, porém utilizando apenas os dados de tempo de quando o veículo está com o motor ligado; para o horímetro de carga, apenas o tempo de quando o veículo está carregado; e para o horímetro de deslocamento, apenas o tempo de quando o veículo está em deslocamento.

Figura 43 – Esboço do indicador de horímetro de motor ligado



Fonte: O Autor (2021)

Para o indicador de histórico de temperatura, foi removido a *label* que contabiliza a quantidade de infrações de temperatura porque o gestor da empresa informou que esse dado também não é coletado.

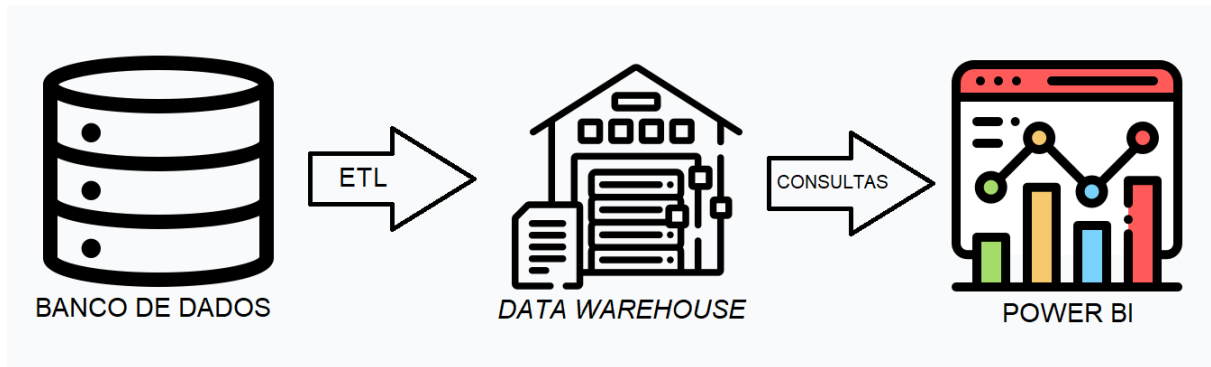
O indicador de contagem de reversão de velocidade foi removido também devido a ausência de dados, e o indicador de histórico de operador removido para diminuir a extensão do *dashboard* e focar na máquina.

## 5.4 EXTRAÇÃO E TRANSFORMAÇÃO DOS DADOS

A extração dos dados, que estão armazenados em um banco de dados MariaDB, será realizada através da linguagem de consulta SQL (*Standard Query Language*). Após a extração dos dados, estes serão armazenados em um *data warehouse* sob a estrutura de esquema em estrela conforme mostra a Figura 45 que será mantido e posteriormente consultado pela ferramenta PowerBI como mostra a Figura 44. Inicialmente, será realizada uma carga inicial dos dados já existentes e, após, com o tempo serão realizadas inserções periódicas dos dados novos.



Figura 44 – Etapas para extração e consulta dos dados

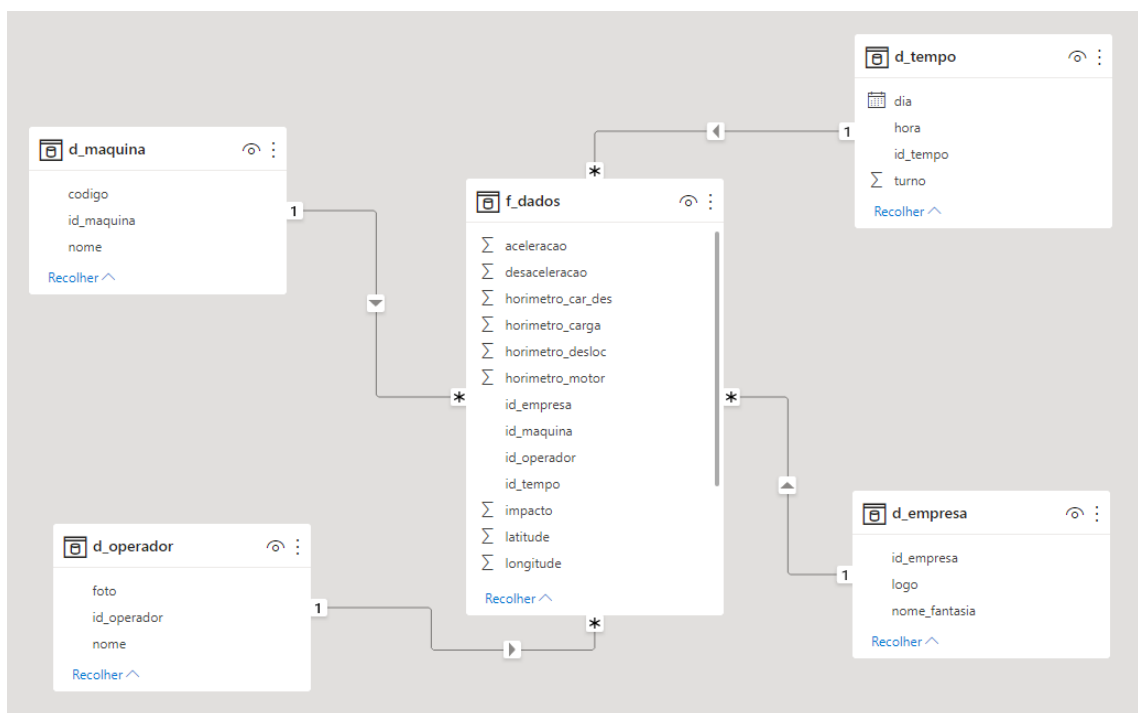


Fonte: O Autor (2021), com ícones retirados do *website* <https://www.flaticon.com/>

### 5.4.1 Data Warehouse

Para este trabalho, a tabela de fatos é apresentada sob o prefixo "f\_" e as tabelas de dimensão sob o prefixo "d\_". A tabela de fatos de dados (f\_dados) encaixa-se no tipo *Periodic Snapshot Fact Table* devido ao armazenamento periódico dos dados provenientes das máquinas que são coletados a cada 10 minutos (ou a cada infração ocorrida) pela empresa em questão, utilizando os campos id\_empresa, id\_maquina, id\_operador e id\_tempo como chave primária composta; e as tabelas dimensões de d\_tempo, d\_empresa, d\_maquina e d\_operador no tipo SCD Tipo 1, armazenando dados de tempo e das entidades empresa, máquina e operador.

Figura 45 – Esboço da estrutura do data warehouse



Fonte: O Autor (2021)

## 6 DESENVOLVIMENTO DA SOLUÇÃO

Este capítulo apresenta o desenvolvimento dos *dashboards* no PowerBI, explicando, também, todas etapas anteriores necessárias para o desenvolvimento como a preparação e extração dos dados para a carga no *data warehouse*.

### 6.1 DADOS

A empresa forneceu os dados através de planilhas no formato CSV (*comma-separated values*), uma para cada tabela do banco de dados, que foram armazenadas numa pasta do Google Drive. Dentre as planilhas fornecidas, encontram-se as tabelas:

- **Dados:** contém os valores referentes às características gerais de uso dos veículos, organizados em 20 colunas: *id\_maquina*, *date*, *id\_operador*, *velocidade*, *velocidade\_infracao*, *rotacao*, *carga\_com*, *carga\_sem*, *carga\_infracao*, *aceleracao*, *desaceleracao*, *temperatura*, *temperatura\_infracao*, *impacto\_x*, *impacto\_y*, *impacto\_z*, *hor1*, *hor2*, *hor3* e *hor4*.
- **Horimetro:** contém a quantidade de horas referente ao uso dos veículos, organizada em 3 colunas: *id\_maquina*, *date* e *valor*.
- **Impacto:** contém os dados referentes aos impactos físicos dos veículos, organizados em 9 colunas: *id\_maquina*, *date*, *id\_operador*, *velocidade*, *lat*, *lng*, *x*, *y*, *z*.
- **Acionamento:** contém os dados referentes aos horários em que os veículos foram ligados e desligados, organizados em 6 colunas: *id\_acionamento*, *id\_maquina*, *id\_operador*, *date\_acionamento*, *date\_desligamento* e *date\_insercao*.
- **Empresa has máquina:** contém os vínculos entre empresas e veículos, organizados em 2 colunas: *id\_maquina* e *id\_empresa*.
- **Empresa has operador:** contém os vínculos entre empresas e operadores de veículos, organizados em 2 colunas: *id\_operador* e *id\_empresa*.

A fim de manter total sigilo dos dados, a empresa não forneceu planilhas referentes às tabelas de empresas, máquinas e operadores. Para contornar esse obstáculo, as tabelas dimensão *d\_empresa*, *d\_maquina* e *d\_operador* do *data warehouse* foram preenchidas com dados fictícios, não havendo a necessidade de criar tabelas intermediárias devido a semelhança entre as estruturas das tabelas do banco de dados da empresa e das tabelas dimensão do *data warehouse*.

## 6.2 EXTRAÇÃO, TRANSFORMAÇÃO E CARGA

Com os dados em mãos, foi criado um banco de dados MySQL versão 5.6.51 através do serviço de hospedagem Umblar, escolhido por oferecer preços acessíveis. Nele, foram criadas as estruturas para armazenar os dados das planilhas e o *data warehouse*.

O Algoritmo 1 mostra a *query* elaborada para realizar a carga de dados referentes à tabela fato *f\_dados*.

Algoritmo 1 – Consulta para a carga de dados gerais

```
1 INSERT INTO f_dados
2 SELECT o.id_empresa ,
3        d.id_maquina ,
4        d.id_operador ,
5        LEFT(CONVERT(CONVERT(TIMESTAMP(d.date), SIGNED), CHAR), 11),
6        MAX(d.velocidade) AS velocidade_maxima ,
7        SUM(d.velocidade_infracao) AS qtd_inf_velocidade ,
8        MAX(d.temperatura) AS temperatura_maxima ,
9        SUM(d.temperatura_infracao) AS qtd_inf_temperatura ,
10       MAX(d.aceleracao) AS aceleracao_maxima ,
11       MAX(d.desaceleracao) AS desaceleracao_maxima ,
12       MAX(d.rotacao) AS rotacao_maxima ,
13       SUM(CASE
14           WHEN (d.impacto_x + d.impacto_y + d.impacto_z) > 0 THEN 1
15           ELSE 0
16       END) AS qtde_impacto ,
17       SPLIT_STR(e.coord, '#', 1) AS latitude ,
18       SPLIT_STR(e.coord, '#', 2) AS longitude ,
19       d.hor1 ,
20       d.hor2 ,
21       d.hor3 ,
22       d.hor4
23 FROM dados d
24 INNER JOIN empresa_has_operador o ON d.id_operador = o.id_operador
25 LEFT JOIN
26     (SELECT i.id_maquina ,
27         LEFT(CONVERT(CONVERT(TIMESTAMP(i.date), SIGNED), CHAR), 11),
28         MAX(CONCAT(CONVERT(i.lat, CHAR), '#', CONVERT(i.lng, CHAR)))
29     FROM impacto i
30     WHERE i.lat != 0
31         AND i.lng != 0
32     GROUP BY LEFT(CONVERT(CONVERT(TIMESTAMP(i.date), SIGNED), CHAR), 11),
33              i.id_maquina) e
34     ON LEFT(CONVERT(CONVERT(TIMESTAMP(d.date), SIGNED), CHAR), 11)
35     = e.id_tempo
36 AND d.id_maquina = e.id_maquina
37 GROUP BY d.id_maquina ,
```

```

38     d.id_operador ,
39     LEFT(CONVERT(CONVERT(TIMESTAMP(d.date), SIGNED), CHAR), 11)

```

Fonte: O Autor (2021)

O Algoritmo 2 mostra a *query* elaborada para realizar a carga de dados referentes à tabela dimensão d\_tempo.

#### Algoritmo 2 – Consulta para carga de dados de tempo

```

1  INSERT INTO d_tempo
2  SELECT DISTINCT LEFT(CONVERT(CONVERT(TIMESTAMP(date), SIGNED),
3  CHAR), 11) AS id_tempo ,
4  CONVERT(date, DATE) AS dia ,
5  CONVERT(CONCAT(LEFT(CONVERT(CONVERT(TIMESTAMP(date),
6  SIGNED), CHAR), 11), '000'), TIME) AS hora ,
7  CONVERT(FLOOR(CONVERT(LEFT(RIGHT(CONVERT(date,
8  SIGNED), 6), 2), SIGNED)/6), SIGNED) + 1 AS turno
9  FROM dados

```

Fonte: O Autor (2021)

O Algoritmo 3 mostra a *query* elaborada para realizar a carga de dados fictícios na tabela dimensão d\_empresa.

#### Algoritmo 3 – Consulta para a carga de dados de empresa

```

1  INSERT INTO d_empresa
2  SELECT DISTINCT id_empresa ,
3  CONCAT('Empresa_', CONVERT(id_empresa, CHAR)) AS nome ,
4  'https://lucaswmolin.com/tcc/img/logotipo.png' AS imagem
5  FROM empresa_has_maquina
6  ORDER BY id_empresa ASC

```

Fonte: O Autor (2021)

O Algoritmo 4 mostra a *query* elaborada para realizar a carga de dados fictícios na tabela dimensão d\_operador.

#### Algoritmo 4 – Consulta para a carga de dados de operador

```

1  INSERT INTO d_operador
2  SELECT DISTINCT id_operador ,
3  CONCAT('Operador_', CONVERT(id_operador, CHAR)) AS nome ,
4  'https://lucaswmolin.com/tcc/img/pessoa.png' AS foto
5  FROM empresa_has_operador
6  ORDER BY id_operador ASC

```

Fonte: O Autor (2021)

O Algoritmo 5 mostra a *query* elaborada para realizar a carga de dados fictícios na tabela dimensão d\_maquina.

#### Algoritmo 5 – Consulta para a carga de dados de operador

```
1 INSERT INTO d_maquina
2 SELECT DISTINCT id_maquina ,
3             CONCAT( 'M' , CONVERT(id_maquina , CHAR)) AS codigo ,
4             CONCAT( 'Maquina_' , CONVERT(id_maquina , CHAR)) AS nome
5 FROM empresa_has_maquina
6 ORDER BY id_maquina ASC
```

Fonte: O Autor (2021)

Para realizar a carga dos dados, foi desenvolvido um extrator, utilizando a linguagem de programação Python versão 3.8, que realiza a leitura das planilhas e a inserção de seus dados para o banco de dados MySQL. Posteriormente, o extrator executa uma série de *queries* escritas em MySQL que faz o processamento e a transformação necessária para a inserção dos dados no *data warehouse*. Registros da tabela dados que não continham *id\_empresa* foram desconsiderados.

O código-fonte inicia com a importação das bibliotecas *os*, *mysql.connector*, *dotenv*, *pandas*, *time* e *calendar*. Após, conforme mostra o Algoritmo 6, foi escrita a função *connect()*, para realizar a conexão com o banco de dados. As credenciais do banco de dados foram armazenadas em um arquivo de formato ENV.

#### Algoritmo 6 – Função *connect()*

```
1 def connect():
2     load_dotenv()
3
4     db_host = os.getenv('DB_HOST')
5     db_name = os.getenv('DB_NAME')
6     db_user = os.getenv('DB_USER')
7     db_pw = os.getenv('DB_PW')
8     db_port = os.getenv('DB_PORT')
9
10    connection = mysql.connector.connect(host=db_host ,
11                                       database=db_name ,
12                                       user=db_user ,
13                                       password=db_pw ,
14                                       port=db_port)
15
16    if connection.is_connected():
17        return connection
18    else:
19        raise Exception('falha_conexao_bd')
```

Fonte: O Autor (2021)

Após a função *connect()*, foram escritas as funções *readCSVFile()* para a leitura das planilhas enviadas pelo gestor da empresa, e *getTableNome* para identificação das tabelas a partir das colunas das planilhas conforme mostra o Algoritmo 7.

Algoritmo 7 – Funções *readCSVFile()* e *getTableNome()*

```
1 def readCSVFile(fileName):
2     data = pd.read_csv(fileName)
3
4     columns = ',_'.join(data.columns)
5
6     print("\nDados:", columns)
7     print("Tabela:", getTableName(columns), '\n')
8
9     return getTableName(columns), columns, data
10
11
12 def getTableName(columns):
13     tables = {
14         "id_maquina ,_date ,_valor": "horimetro",
15         ...
16         "id_maquina ,_id_empresa": "empresa_has_maquina",
17         "id_operador ,_id_empresa": "empresa_has_operador"
18     }
19
20     table = tables.get(columns)
21     if table is None:
22         raise Exception('tabela_nao_encontrada')
23     else:
24         return table
```

Fonte: O Autor (2021)

Com a conexão ao banco de dados e a leitura das planilhas, foi escrita a função *insertData()* que realiza a transferência dos dados das planilhas para as tabelas no banco de dados conforme mostra o Algoritmo 8.

Algoritmo 8 – Função *insertData()*

```
1 def insertData(conn, table, columns, data):
2     cursor = conn.cursor()
3
4     records = data.to_records(index=False)
5     values = [tuple([str(s) for s in list(r)]) for r in list(records)]
6
7     cvalues = ',_'.join(['%s' for c in data.columns])
8
```

```

9      sql = "INSERT INTO_" + table + "(" + columns + \
10         ") VALUES_" + cvalues + ")"
11
12     print(sql, '\n')
13
14     ct = 0
15     cg = 0
16     lim = 1000
17     mx = len(values)
18     tvalues = []
19     for v in values:
20         tvalues.append(v)
21         ct += 1
22         cg += 1
23
24     if ct == lim or cg == mx:
25         cursor.executemany(sql, tvalues)
26
27         print("%.2f" % (cg/mx * 100) + "%")
28
29         tvalues.clear()
30         ct = 0

```

Fonte: O Autor (2021)

Após a função de transferência dos dados das planilhas para o banco de dados, foi escrita a função *loadDataWarehouse()* que executa as *queries* de transformação e carga dos dados para o *data warehouse* e esvazia as tabelas com os dados de planilhas conforme mostra o Algoritmo 9.

#### Algoritmo 9 – Função *loadDataWarehouse()*

```

1  def loadDataWarehouse(conn):
2      queries = [
3          "queries/d_empresa.txt",
4          "queries/d_maquina.txt",
5          "queries/d_operador.txt",
6          "queries/d_tempo.txt",
7          "queries/f_dados.txt",
8          "queries/truncate_empresa_has_operador.txt",
9          "queries/truncate_empresa_has_maquina.txt",
10         "queries/truncate_horimetro.txt",
11         "queries/truncate_impacto.txt",
12         "queries/truncate_dados.txt",
13         "queries/truncate_acionamento.txt"
14     ]
15
16     for query in queries:

```

```

17     print('\nExecutando_' + query)
18     file = open(query, "r")
19     sql = file.read()
20
21     cursor = conn.cursor()
22     cursor.execute(sql)
23
24     print('\n' + str(cursor.rowcount) + '_registros')

```

Fonte: O Autor (2021)

Por fim, foi escrita a função *writeLog()*, conforme o Algoritmo 10, que realiza a escrita do resultado da extração em um arquivo de texto, conforme a Figura 46, para o usuário verificar o histórico de extrações, podendo constatar se ocorreu uma falha e, se ocorreu, por qual motivo.

#### Algoritmo 10 – Função *writeLog()*



```

1 def writeLog(status, msg):
2     ts = calendar.timegm(time.gmtime())
3
4     arquivo = open('logs/' + status + str(ts) + '.txt', 'w')
5     arquivo.write(msg)
6     arquivo.close()

```

Fonte: O Autor (2021)

Figura 46 – Pasta contendo os arquivos de *log* das extrações

Nome	Data de modificação	Tipo	Tamanho
 ERRO1634965298	23/10/2021 02:01	Documento de Texto	1 KB
 OK1634965103	23/10/2021 01:58	Documento de Texto	1 KB

Fonte: O Autor (2021)

As funções são executadas conforme a função principal *main()* mostrada no Algoritmo 11.

#### Algoritmo 11 – Função *main()*

```

1 def main():
2     conn = connect()
3
4     fileList = [
5         "files/acionamento.csv",
6         "files/impacto.csv",
7         "files/horimetro.csv",
8         "files/dados.csv",
9         "files/empresa_has_maquina.csv",

```



```

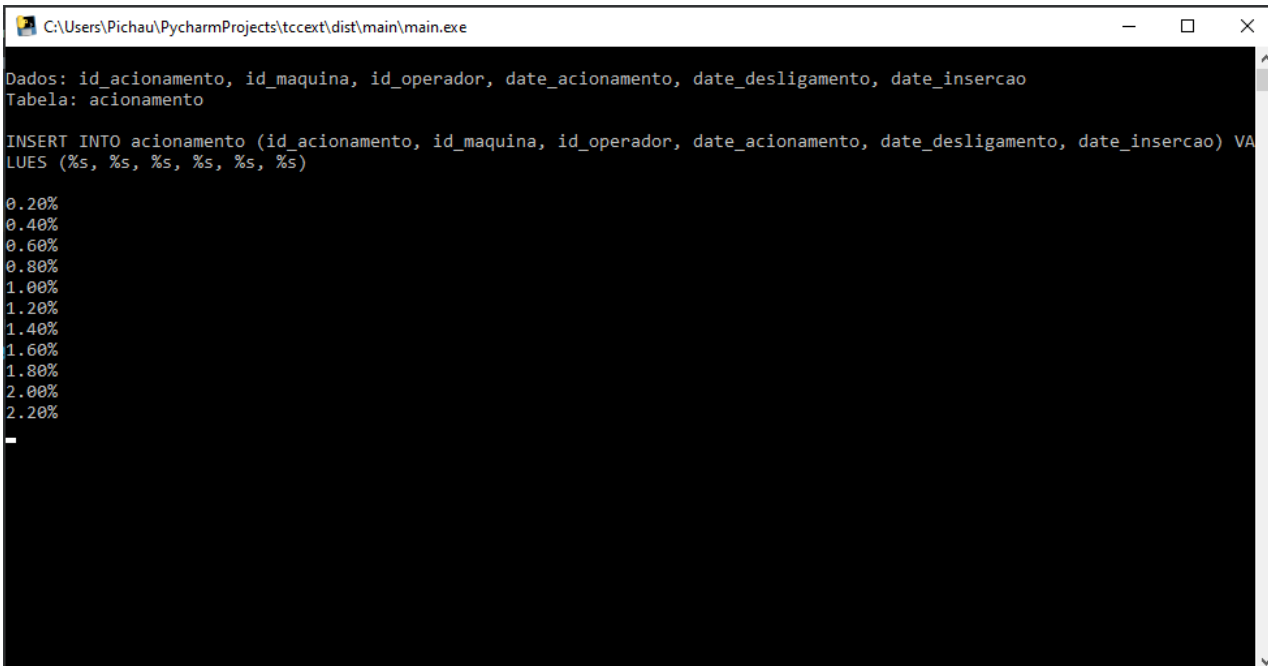
10         "files/empresa_has_operador.csv"
11     ]
12
13     for file in filesList:
14         parameters = readCSVFile(file)
15         insertData(conn, parameters[0], parameters[1], parameters[2])
16
17     loadDataWarehouse(conn)
18
19     conn.commit()
20     conn.close()
21
22     writeLog('OK', 'extracao_realizada.')
23
24
25 if __name__ == '__main__':
26     try:
27         main()
28     except Exception as e:
29         writeLog('ERRO', str(e))

```

Fonte: O Autor (2021)

O extrator pode ser executado manualmente quando o gestor da empresa possui dados novos, ou automaticamente a partir do agendador de tarefas de um servidor. O ideal seria que a empresa tivesse fornecido acesso direto ao banco de dados, para assim facilitar a automatização do processo; porém a empresa optou por fornecer os dados em forma de planilhas. A Figura 47 mostra a execução do extrator em andamento.

Figura 47 – Screenshot da execução do extrator



```
C:\Users\Pichau\PycharmProjects\tccext\dist\main\main.exe
Dados: id_acionamento, id_maquina, id_operador, date_acionamento, date_desligamento, date_insercao
Tabela: acionamento
INSERT INTO acionamento (id_acionamento, id_maquina, id_operador, date_acionamento, date_desligamento, date_insercao) VALUES (%s, %s, %s, %s, %s, %s)
0.20%
0.40%
0.60%
0.80%
1.00%
1.20%
1.40%
1.60%
1.80%
2.00%
2.20%
```

Fonte: O Autor (2021)

### 6.3 CONSTRUÇÃO

Conforme citado anteriormente, a ferramenta escolhida para a construção dos *dashboards* foi o Microsoft Power BI<sup>1</sup>, tendo sido necessário estudá-lo desde os conceitos básicos aos avançados.

Com a carga dos dados para o *data warehouse*, iniciou-se a elaboração de consultas específicas para cada indicador utilizando a linguagem SQL. O Power BI dispõe da linguagem interna DAX *Data Analysis Expressions* para manipulação de dados, porém devido à praticidade foi optado por utilizar principalmente SQL. Para cálculos pequenos como, por exemplo, o cálculo de produtividade de veículo, foi utilizada a linguagem DAX, conforme mostra o Algoritmo 12.

#### Algoritmo 12 – Cálculo de produtividade utilizando a linguagem DAX

```
1 produtividade = DIVIDE([d_hora_ativo], [d_hora_motor]) * 100
```

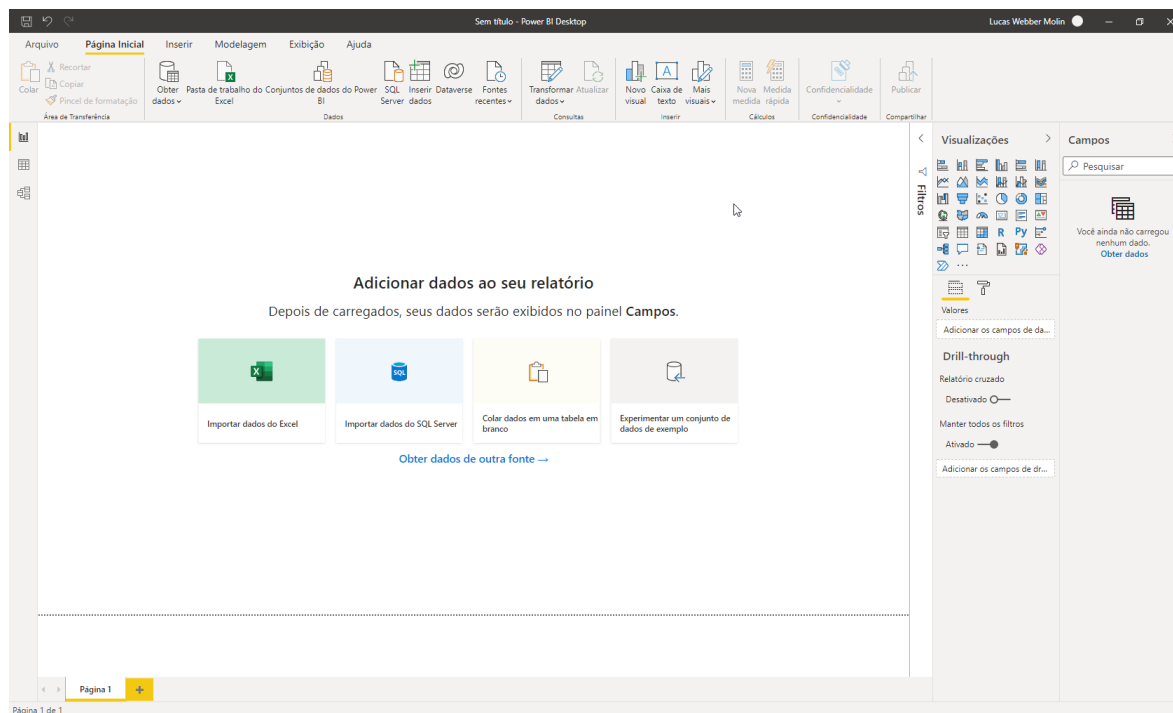
Fonte: O Autor (2021)

No Power BI, o primeiro passo foi criar um arquivo de relatório. Ao abrir o arquivo, a ferramenta apresenta-se conforme a Figura 48.

O passo seguinte foi a criação de fontes de dados, clicando-se no botão *Obter dados* para escolher o conector para o banco de dados MySQL e preencher suas credenciais de conexão

<sup>1</sup> <https://powerbi.microsoft.com/pt-br/>

Figura 48 – Tela inicial da ferramenta Power BI



Fonte: O Autor (2021)

(servidor, porta, banco de dados, usuário e senha) e a consulta SQL. Esse passo foi repetido para cada fonte de dados.

### 6.3.1 Consultas para o *dashboard* geral de veículos

O Algoritmo 13 mostra a consulta que retorna a quantidade de veículos e operadores em atividade a cada hora para cada dia de cada empresa. Essa consulta é utilizada para os gráficos dos indicadores de veículos e operadores em atividade.

Algoritmo 13 – Consulta para veículos e operadores em atividade

```

1 SELECT de.nome_fantasia AS empresa ,
2     dt.dia ,
3     TIME_FORMAT(dt.hora , "%H") AS hora ,
4     COUNT(DISTINCT fd.id_maquina) AS qtd_maquina ,
5     COUNT(DISTINCT fd.id_operador) AS qtd_operador
6 FROM f_dados fd
7 INNER JOIN d_tempo dt ON fd.id_tempo = dt.id_tempo
8 INNER JOIN d_empresa de ON fd.id_empresa = de.id_empresa
9 GROUP BY fd.id_empresa ,
10     dt.dia ,
11     TIME_FORMAT( dt.hora , "%H" )
    
```

Fonte: O Autor (2021)

O Algoritmo 14 mostra a consulta que retorna a quantidade total de veículos e operadores que estiveram em atividade em cada dia para cada empresa. Essa consulta é utilizada para exibir o total nos indicadores de veículos e operadores em atividade.

#### Algoritmo 14 – Consulta para veículos e operadores do dia

```
1 SELECT de.nome_fantasia AS empresa ,
2     dt.dia ,
3     COUNT(DISTINCT fd.id_maquina) AS qtd_maquina ,
4     COUNT(DISTINCT fd.id_operador) AS qtd_operador
5 FROM f_dados fd
6 INNER JOIN d_tempo dt ON fd.id_tempo = dt.id_tempo
7 INNER JOIN d_empresa de ON fd.id_empresa = de.id_empresa
8 GROUP BY fd.id_empresa ,
9     dt.dia
```

Fonte: O Autor (2021)

O Algoritmo 15 mostra a consulta que retorna a quantidade de infrações de velocidade cometidas por todos veículos a cada hora para cada dia de cada empresa. É utilizada para o indicador de monitoramento de infrações de velocidade.

#### Algoritmo 15 – Consulta para o indicador de monitoramento de infrações de velocidade

```
1 SELECT de.nome_fantasia AS empresa ,
2     dt.dia ,
3     TIME_FORMAT(dt.hora , "%H") AS hora ,
4     SUM(fd.velocidade_infracao) AS inf_velocidade
5 FROM f_dados fd
6 INNER JOIN d_tempo dt ON fd.id_tempo = dt.id_tempo
7 INNER JOIN d_empresa de ON fd.id_empresa = de.id_empresa
8 GROUP BY de.nome_fantasia ,
9     dt.dia ,
10    TIME_FORMAT(dt.hora , "%H")
11 ORDER BY fd.id_tempo
```

Fonte: O Autor (2021)

O Algoritmo 16 mostra a consulta que retorna a quantidade de impactos de veículos a cada hora para cada dia de cada empresa. É utilizada para o indicador de registro de impacto.

#### Algoritmo 16 – Consulta para o indicador de registro de impacto

```
1 SELECT de.nome_fantasia AS empresa ,
2     dt.dia ,
3     TIME_FORMAT(dt.hora , "%H") AS hora ,
4     SUM(fd.impacto) AS qtd_impacto
5 FROM f_dados fd
6 INNER JOIN d_tempo dt ON fd.id_tempo = dt.id_tempo
7 INNER JOIN d_empresa de ON fd.id_empresa = de.id_empresa
```

```

8 GROUP BY fd.id_empresa ,
9         dt.dia ,
10        TIME_FORMAT(dt.hora , "%H")

```

Fonte: O Autor (2021)

O Algoritmo 17 mostra a consulta que retorna para cada dia de cada empresa os veículos e seus respectivos operadores, a quantidade de infrações de velocidade, a quantidade de impactos e a produtividade (isto é, a relação entre horas do veículo com carga ou deslocamento e horas com motor ligado). Os dados retornados são utilizados para a tabela de dados gerais.

#### Algoritmo 17 – Consulta para a tabela de dados por máquina

```

1 SELECT de.nome_fantasia AS empresa ,
2        dm.nome AS maquina ,
3        GROUP_CONCAT(DISTINCT dp.nome SEPARATOR ',_') AS operadores ,
4        dt.dia ,
5        SUM(fd.velocidade_infracao) AS qtd_inf_velocidade ,
6        SUM(fd.impacto) AS qtd_impacto ,
7        TRUNCATE(MAX(fd.horimetro_car_des)/
8        MAX(fd.horimetro_motor) * 100, 2) as produtividade
9 FROM f_dados fd
10 INNER JOIN d_tempo dt ON fd.id_tempo = dt.id_tempo
11 INNER JOIN d_empresa de ON fd.id_empresa = de.id_empresa
12 INNER JOIN d_maquina dm ON fd.id_maquina = dm.id_maquina
13 INNER JOIN d_operador dp ON fd.id_operador = dp.id_operador
14 GROUP BY fd.id_empresa ,
15         fd.id_maquina ,
16         dt.dia

```

Fonte: O Autor (2021)

### 6.3.2 Consultas para o *dashboard* específico por veículo

O Algoritmo 18 mostra a consulta que retorna o valor de horímetro de motor ligado, com carga, em deslocamento e com carga/em deslocamento a cada hora em cada dia para cada veículo de cada empresa. É utilizada para os indicadores de horímetro.

#### Algoritmo 18 – Consulta para os indicadores de horímetro

```

1 SELECT de.nome_fantasia AS empresa ,
2        dm.nome AS maquina ,
3        dt.dia ,
4        TIME_FORMAT(dt.hora , "%H") AS hora ,
5        MAX(fd.horimetro_motor) AS hora_motor ,
6        MAX(fd.horimetro_carga) AS hora_carga ,
7        MAX(fd.horimetro_desloc) AS hora_desc ,
8        MAX(fd.horimetro_car_des) AS hora_ativo

```

```

9 FROM f_dados fd
10 INNER JOIN d_tempo dt ON fd.id_tempo = dt.id_tempo
11 INNER JOIN d_empresa de ON fd.id_empresa = de.id_empresa
12 INNER JOIN d_maquina dm ON fd.id_maquina = dm.id_maquina
13 GROUP BY de.nome_fantasia ,
14           dm.nome ,
15           dt.dia ,
16           TIME_FORMAT(dt.hora , "%H")
17 ORDER BY fd.id_tempo

```

Fonte: O Autor (2021)

O Algoritmo 19 mostra a consulta que retorna para cada hora a velocidade máxima, a quantidade de infrações de velocidade, a temperatura máxima, a aceleração máxima, a desaceleração máxima e a rotação máxima de cada veículo de cada empresa em cada dia. É utilizada para os indicadores de velocidade, temperatura, aceleração, desaceleração e rotação.

#### Algoritmo 19 – Consulta para os indicadores de velocidade, temperatura etc

```

1 SELECT de.nome_fantasia AS empresa ,
2        dm.nome AS maquina ,
3        dt.dia ,
4        TIME_FORMAT(dt.hora , "%H") AS hora ,
5        MAX(fd.velocidade) AS velocidade ,
6        SUM(fd.velocidade_infracao) AS inf_velocidade ,
7        MAX(fd.temperatura) AS temperatura ,
8        MAX(fd.aceleracao) AS aceleracao ,
9        MAX(fd.desaceleracao) AS desaceleracao ,
10       MAX(fd.rotacao) AS rotacao
11 FROM f_dados fd
12 INNER JOIN d_tempo dt ON fd.id_tempo = dt.id_tempo
13 INNER JOIN d_empresa de ON fd.id_empresa = de.id_empresa
14 INNER JOIN d_maquina dm ON fd.id_maquina = dm.id_maquina
15 GROUP BY de.nome_fantasia ,
16           dm.nome ,
17           dt.dia ,
18           TIME_FORMAT(dt.hora , "%H")
19 ORDER BY fd.id_tempo

```

Fonte: O Autor (2021)

O Algoritmo 20 mostra a consulta que retorna o horário aproximado e as coordenadas geográficas do último impacto e a quantidade de impactos de cada veículo de cada empresa em cada dia. É utilizado para o indicador de localização e último impacto.

#### Algoritmo 20 – Consulta para o indicador de impacto e localização

```

1 SELECT de.nome_fantasia AS empresa ,
2        dm.nome AS maquina ,

```

```

3      dtt.dia ,
4      dtt.hora ,
5      fdd.latitude ,
6      fdd.longitude ,
7      a.qtd_impacto
8 FROM
9      (SELECT fd.id_empresa ,
10         fd.id_maquina ,
11         dt.dia ,
12         MAX(fd.id_tempo) AS m_tempo ,
13         SUM(fd.impacto) AS qtd_impacto
14 FROM f_dados fd
15 INNER JOIN d_tempo dt ON fd.id_tempo = dt.id_tempo
16 WHERE fd.latitude IS NOT NULL
17        AND fd.longitude IS NOT NULL
18 GROUP BY fd.id_empresa ,
19         fd.id_maquina ,
20         dt.dia) a
21 INNER JOIN f_dados fdd ON a.id_empresa = fdd.id_empresa
22 AND a.id_maquina = fdd.id_maquina
23 AND a.m_tempo = fdd.id_tempo
24 INNER JOIN d_empresa de ON a.id_empresa = de.id_empresa
25 INNER JOIN d_maquina dm ON a.id_maquina = dm.id_maquina
26 INNER JOIN d_tempo dtt ON a.m_tempo = dtt.id_tempo

```

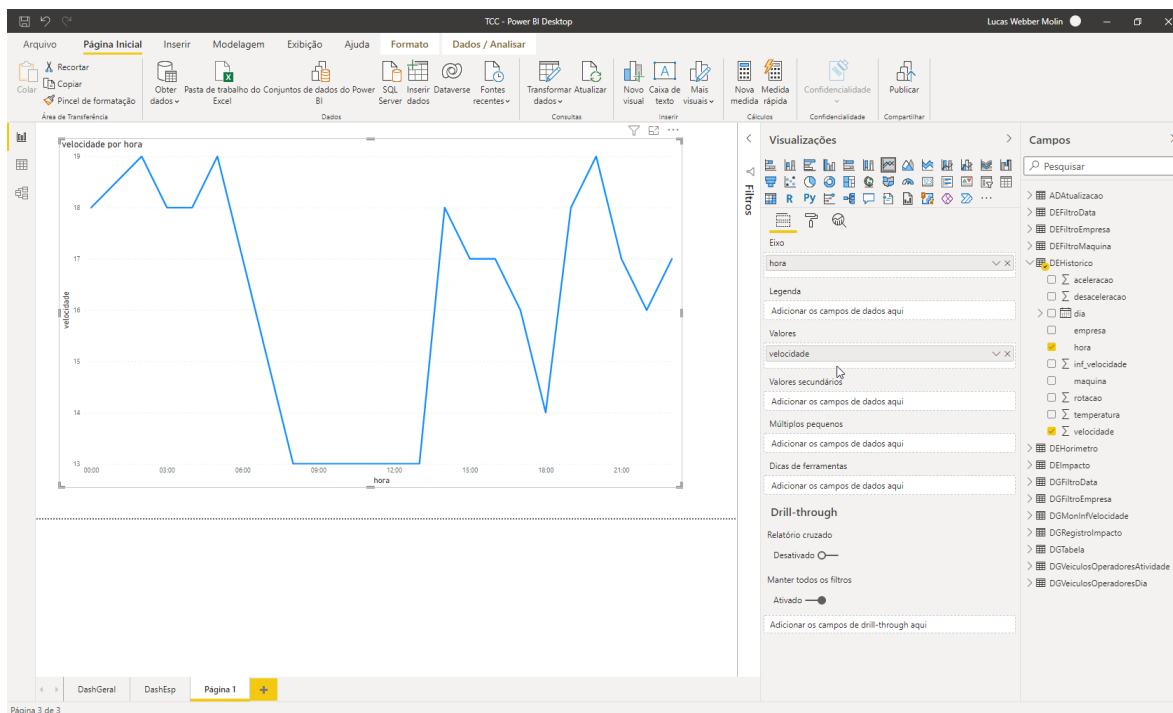
Fonte: O Autor (2021)

### 6.3.3 Construção dos *dashboards*

Com a carga dos dados para o PowerBI, foi iniciada a etapa de construção dos indicadores. Conforme mostra a Figura 49, a construção é relativamente simples, bastando escolher o tipo de gráfico e arrastar as colunas de valores que representam o eixo X e as séries do eixo Y. Caso não tivesse sido elaborada uma consulta SQL para cada indicador, apenas carregado o *data warehouse* diretamente para o Power BI, teria sido necessário aprofundar o conhecimento na linguagem DAX para manipular os dados internamente.

Para a lapidação do visual do gráfico, há a aba de Formato (ilustrada com o ícone de rolo de pintura), que disponibiliza opções para escolher, por exemplo, as cores, títulos, tamanho etc do gráfico. A aba de Análise (ao lado direito da aba de Formato) oferece a possibilidade de inserir linhas referentes ao valor médio, máximo, mínimo, entre outros, dos valores inseridos para o eixo Y.

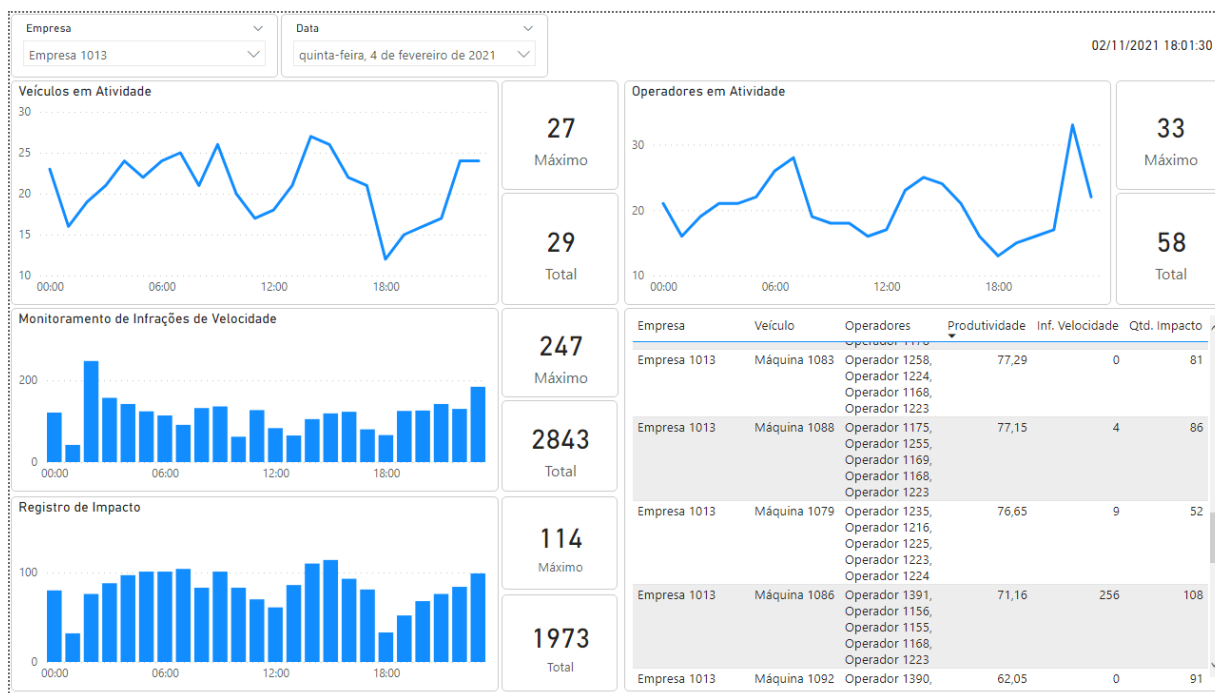
Figura 49 – Construção de gráficos na ferramenta Power BI



Fonte: O Autor (2021)

Após a montagem dos gráficos, obteve-se para o *dashboard* geral de veículos o resultado mostrado na Figura 50.

Figura 50 – Resultado final do *dashboard* geral



Fonte: O Autor (2021)

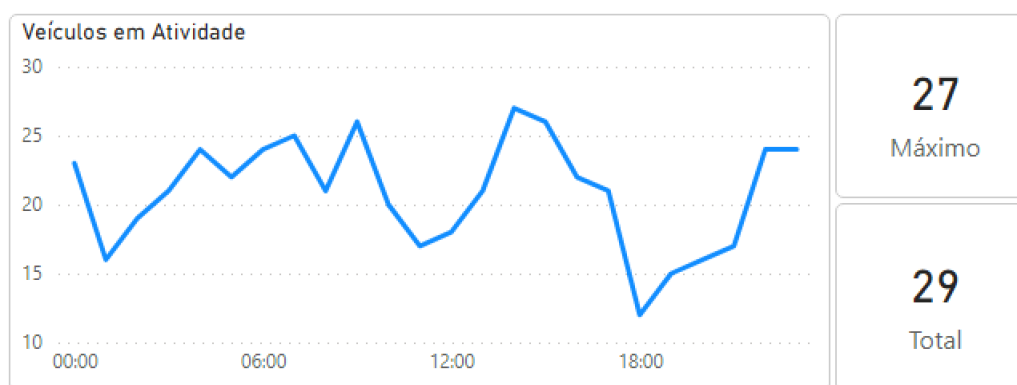


Devido à sugestão de proporção padrão do Power BI de 16:9 para melhor visualização, ambos os *dashboards* foram ajustados, tendo indicadores realocados, removidos ou unificados. Também, devido a limitações na estilização dos componentes visuais, o totalizador dos indicadores foi realocado para a lateral direita em vez de permanecer no topo.

No caso do *dashboard* geral, na primeira linha posicionou-se os filtros de empresa e data, além de um *label* mostrando o último horário de execução das consultas SQL.

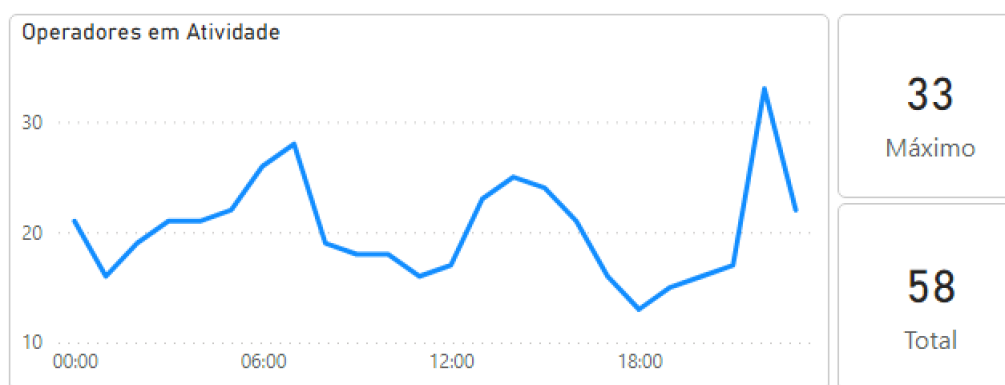
Na segunda linha, como mostram as Figuras 51 e 52, posicionou-se os indicadores de veículos e operadores em atividade, mostrando gráficos de linha com a quantidade total a cada hora e a quantidade máxima numa hora e a quantidade total do dia todo. Assim, o usuário pode acompanhar a lotação do local ao longo do dia.

Figura 51 – Indicador de Veículos em Atividade



Fonte: O Autor (2021)

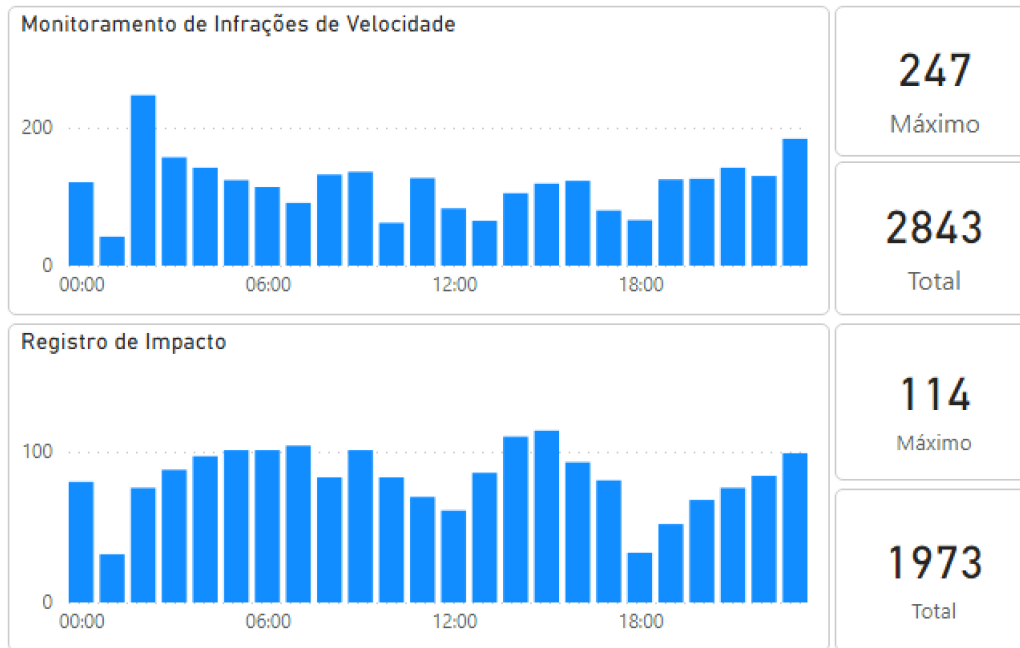
Figura 52 – Indicador de Operadores em Atividade



Fonte: O Autor (2021)

Na terceira linha, como mostram a Figura 53, foram posicionados os indicadores de monitoramento de infrações de velocidade e de registro de impacto. Foram mostradas suas quantidades a nível de hora, a quantidade máxima numa hora e a quantidade total no dia, permitindo ao usuário saber em quais horários deve-se ter mais cuidado para evitar acidentes.

Figura 53 – Indicadores de Infrações de Velocidade e Impacto



Fonte: O Autor (2021)

Por fim, tem-se uma tabela, como mostra a Figura 54, que mostra um resumo da quantidade de infrações de velocidade e impacto para cada veículo no dia filtrado, juntamente com o nível de produtividade e a lista de operadores que utilizaram aquele veículo. Assim, o usuário pode identificar quais veículos e operadores são mais produtivos (ou menos).

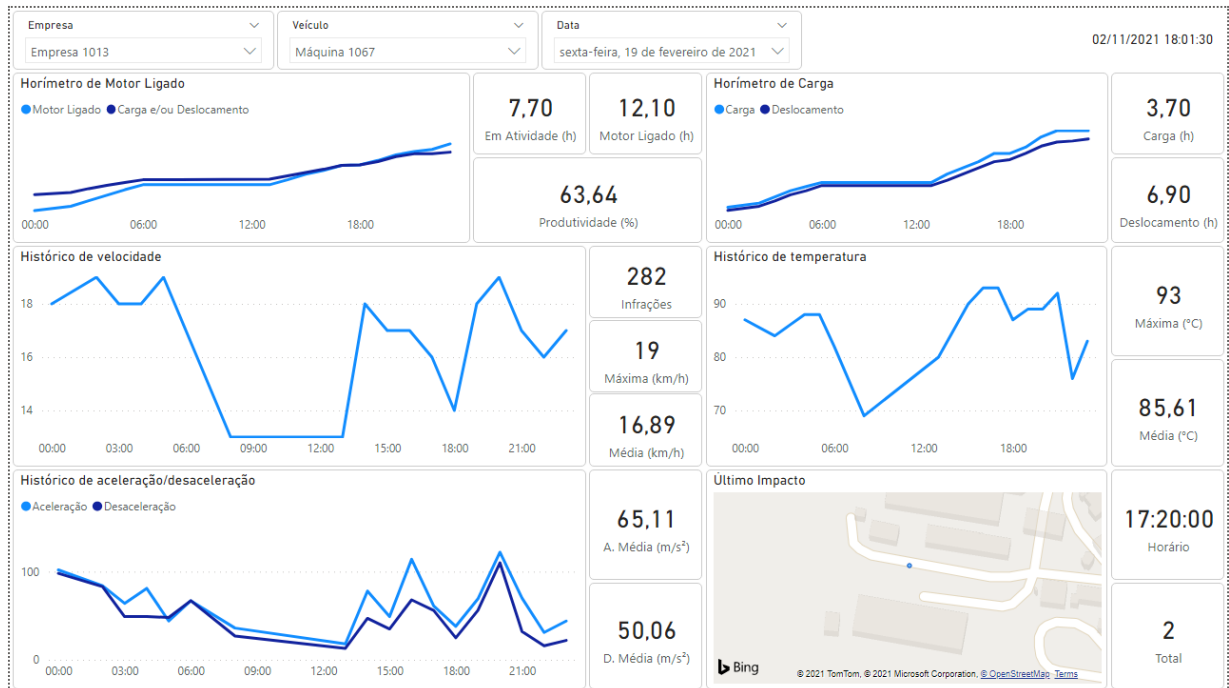
Figura 54 – Tabela de produtividade e infrações

Empresa	Veículo	Operadores	Produtividade	Inf. Velocidade	Qtd. Impacto
Empresa 1013	Máquina 1083	Operador 1258, Operador 1224, Operador 1168, Operador 1223	77,29	0	81
Empresa 1013	Máquina 1088	Operador 1175, Operador 1255, Operador 1169, Operador 1168, Operador 1223	77,15	4	86
Empresa 1013	Máquina 1079	Operador 1235, Operador 1216, Operador 1225, Operador 1223, Operador 1224	76,65	9	52
Empresa 1013	Máquina 1086	Operador 1391, Operador 1156, Operador 1155, Operador 1168, Operador 1223	71,16	256	108
Empresa 1013	Máquina 1092	Operador 1390,	62,05	0	91

Fonte: O Autor (2021)

Para o *dashboard* específico por veículo, obteve-se o resultado mostrado na Figura 55.

Figura 55 – Resultado final do *dashboard* específico

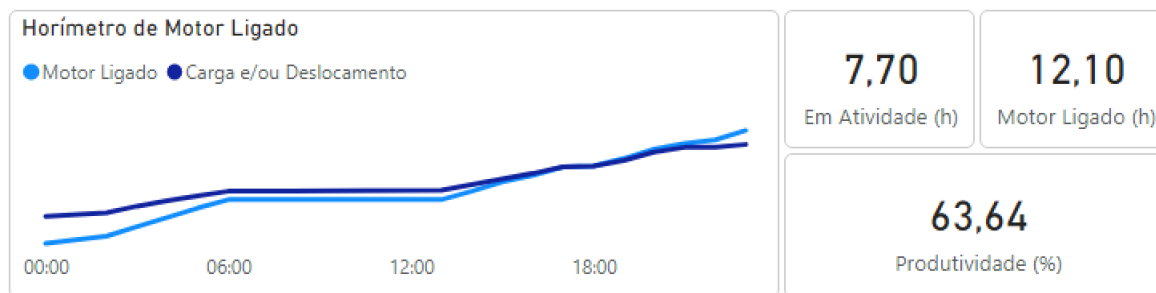


Fonte: O Autor (2021)

Na primeira linha foram apresentados os filtros de empresa, veículo e data, além do *label* com o último horário de atualização das fontes de dados.

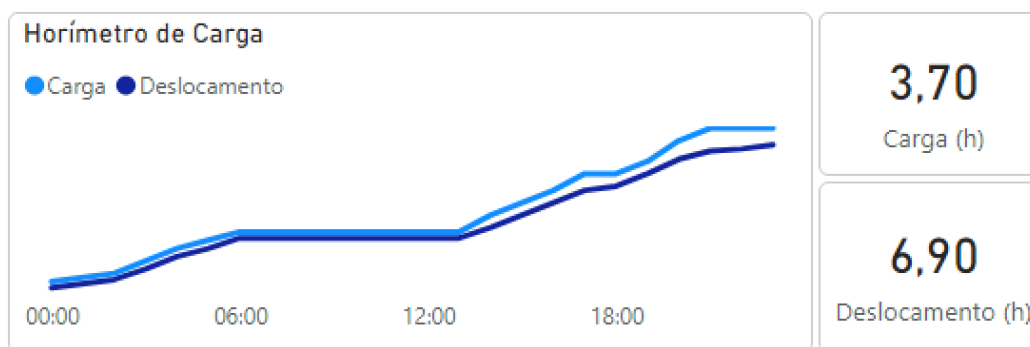
Na segunda linha, foram posicionados os indicadores de horímetro de motor ligado e o horímetro de carga e deslocamento. Para o indicador de horímetro de motor ligado, como mostra a Figura 56, foi construído um gráfico de linhas mostrando a contagem de horas para motor ligado e também para veículo em atividade (ou seja, quando o veículo estava com carga e/ou em deslocamento, podendo-se, assim, calcular a produtividade, conforme orientado pelo gestor da empresa. Para o indicador de horímetro de carga e deslocamento, como mostra a Figura 57, foi construído um gráfico de linhas mostrando ambas as informações, com um cartão com o valor total de cada. Assim, é possível para o usuário identificar em quais horários o veículo é mais utilizado e qual sua respectiva produtividade.

Figura 56 – Indicador de horímetro de motor ligado



Fonte: O Autor (2021)

Figura 57 – Indicador de horímetro de carga e deslocamento



Fonte: O Autor (2021)

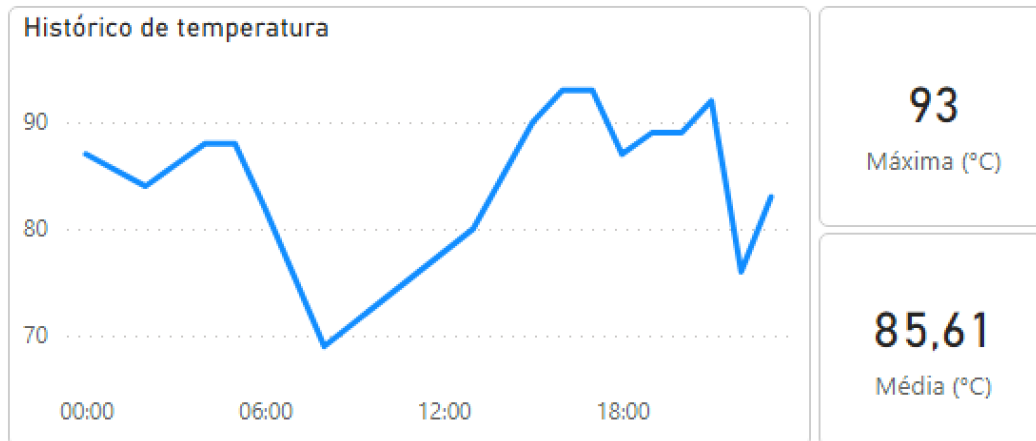
Na terceira linha, como mostram as Figuras 58 e 59, foram posicionados os indicadores de histórico de velocidade e histórico de temperatura, conforme o esboço apresentado no planejamento, mostrando a velocidade e a temperatura máxima atingida pelo veículo a cada hora, além de informações como valor médio, máximo e contagem de infrações. Com esse histórico, é possível para o usuário identificar má conduta por parte do operador e até mesmo possíveis falhas no motor do veículo.

Figura 58 – Indicador de histórico de velocidade



Fonte: O Autor (2021)

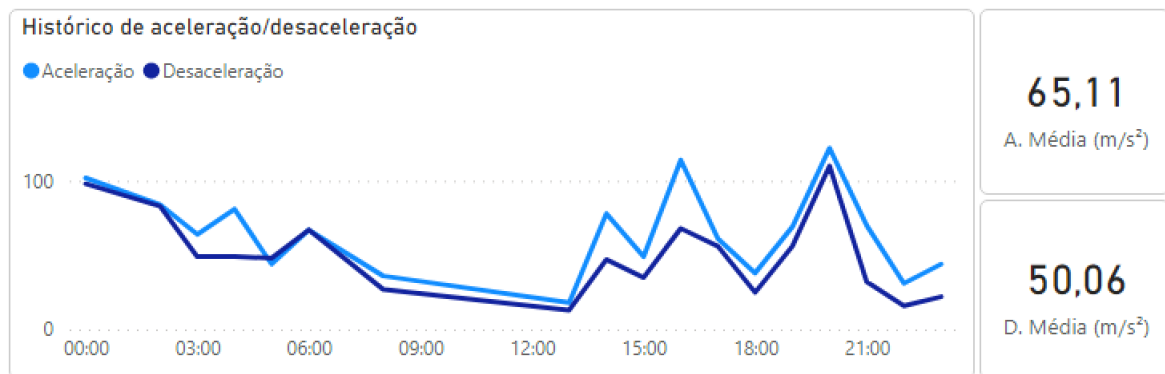
Figura 59 – Indicador de histórico de temperatura



Fonte: O Autor (2021)

Na quarta linha, foram posicionados o indicador de histórico de aceleração e desaceleração e o indicador de último impacto. Para o indicador de aceleração e desaceleração, como mostra a Figura 60, foi construído um gráfico de linha conforme o esboço, porém unificando as duas séries de valores que mostram uma possível correlação direta. Ao lado, foram apresentadas as informações de valor médio e máximo. Assim, é possível identificar má conduta por parte do operador ou problemas no veículo.

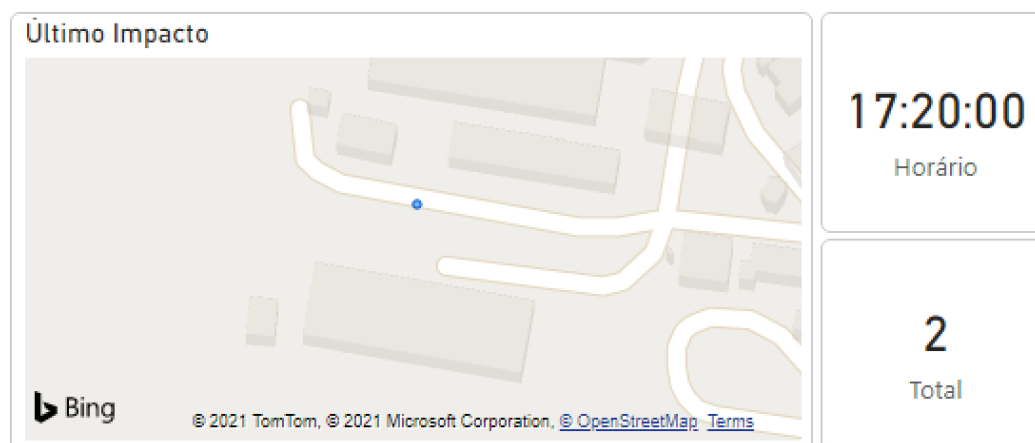
Figura 60 – Indicador de histórico de aceleração e desaceleração



Fonte: O Autor (2021)

Para o indicador de impacto, como mostra a Figura 61, foi posicionado um mapa que mostra a localização geográfica exata do último impacto do veículo, juntamente com o horário aproximado e a quantidade total de impactos no dia filtrado. O indicador de histórico de rotação não foi incluído devido à falta de espaço, mas também devido a sua forte relação com a velocidade, tornando-o redundante. Com o mapa, torna-se possível identificar os locais mais frequentes de colisões e também localizar veículos e operadores.

Figura 61 – Indicador de último impacto



Fonte: O Autor (2021)

O visual de ambos os *dashboards* foram construídos visando o foco na transmissão da informação para o usuário, por isso foram usadas poucas cores, espaçamento entre os indicadores, e um fundo claro.

Os *dashboards* podem ser compartilhados em páginas *web*, mas para isso é necessária a assinatura do Power BI Pro.

## 7 CONCLUSÕES

A fim de alcançar o objetivo deste trabalho, estudou-se a bibliografia fundamental de ciência de dados, começando por *business intelligence*, passando por *data warehouse*, ETL, e finalizando em gráficos e *dashboards*. Após, deu-se início ao estudo dos dados fornecidos pela empresa envolvida.

Com a base estabelecida, deu-se início à modelagem dos *dashboards* considerando a bibliografia referente à visualização de dados, e, após, a modelagem do *data warehouse* considerando a bibliografia referente à *business intelligence*. Com o planejamento concluído, deu-se início aos estudos da ferramenta Power BI.

Com o fornecimento dos dados pela empresa envolvida, deu-se início ao desenvolvimento de um *script* para a extração dos dados e construção do *data warehouse* para o desenvolvimento do produto final, que foram dois *dashboards* operacionais focados no monitoramento da integridade de veículos de transporte de carga. Os *dashboards* foram construídos utilizando-se a ferramenta Power BI através das linguagens SQL e DAX para a consulta e manipulação de dados.

Como a empresa está em processo de modificação de seus sistemas, analisando alternativas de ferramentas de desenvolvimento e de visualização, o objetivo desse trabalho foi desenvolver uma proposta de visualização de dados com *dashboards* através da utilização da ferramenta PowerBI como uma sugestão de solução para a empresa. Esse trabalho proporcionou uma experiência significativa de engenharia de dados, e de conhecimento da ferramenta PowerBI.

Para melhorias futuras, poderiam ser realizados o planejamento e a construção de outros *dashboards* baseados em técnicas avançadas de análise de dados, envolvendo, por exemplo, aprendizado de máquina para identificar correlações entre os dados.

## REFERÊNCIAS

- AMARAL, F. **Aprenda mineração de dados: teoria e prática**. 1. ed. Rio de Janeiro: Alta Books, 2016. 240 p.
- BASSO, D. **Big Data**. 1. ed. Curitiba: Contentus, 2020. 96 p.
- DÍAZ, J. **Introduccion al Business Intelligence**. 1. ed. Barcelona: Editorial UOC, 2010. 238 p.
- INMON, W. **Building the Data Warehouse**. 3. ed. Nova Iorque: John Wiley & Sons, Inc., 2002. 428 p.
- KIMBALL, R.; ROSS, M. **The Data Warehouse Toolkit: The definitive guide to dimensional modeling**. 1. ed. Indianapolis: John Wiley & Sons, Inc., 2009. 286 p.
- KNAFLIC, C. **storytelling com dados: um guia sobre visualização de dados para profissionais de negócios**. 1. ed. Rio de Janeiro: Alta Books, 2019. 256 p.
- LIMA, M. **The Book of Circles: Visualizing spheres of knowledge**. 1. ed. Nova Iorque: Princeton Architectural Press, 2017. 272 p.
- MALIK, S. **Enterprise Dashboards: Design and best practices for it**. 1. ed. Hoboken: John Wiley & Sons, Inc., 2005. 222 p.
- PRIMAK, F. **Decisões com BI (Business Intelligence)**. 1. ed. Rio de Janeiro: Ciência Moderna, 2020. 168 p.
- RASMUSSEN, N.; BANSAL, M.; CHEN, C. **Business Dashboards: A visual catalog for design and deployment**. 3. ed. Hoboken: John Wiley & Sons, Inc., 2013. 564 p.
- RIBEIRO, D. M. **Visualização de dados na Internet**. Dissertação (Mestrado) — Pontifícia Universidade Católica de São Paulo, São Paulo, 2015.
- SHARDA, R.; DELEN, D.; TURBAN, E. **Business Intelligence, analytics, and data science: A managerial perspective**. 4. ed. Harlow: Pearson Global, 2018. 514 p.



