

## PROJETO SANJA LOG

Fabiana Vasques de Araujo

([https://www.linkedin.com/in/fabiana-vasques-7b9b5927?utm\\_source=share&utm\\_campaign=share\\_via&utm\\_content=profile&utm\\_medium=android\\_app](https://www.linkedin.com/in/fabiana-vasques-7b9b5927?utm_source=share&utm_campaign=share_via&utm_content=profile&utm_medium=android_app))

Geovane Afonso Leite Nunes

([https://www.linkedin.com/in/geovanealunes?utm\\_source=share&utm\\_campaign=share\\_via&utm\\_content=profile&utm\\_medium=android\\_app](https://www.linkedin.com/in/geovanealunes?utm_source=share&utm_campaign=share_via&utm_content=profile&utm_medium=android_app))

João Nicholas dos Santos

([https://www.linkedin.com/in/nicholas-fran%C3%A7a-a87b19250?utm\\_source=share&utm\\_campaign=share\\_via&utm\\_content=profile&utm\\_medium=ios\\_app](https://www.linkedin.com/in/nicholas-fran%C3%A7a-a87b19250?utm_source=share&utm_campaign=share_via&utm_content=profile&utm_medium=ios_app))

Kathellyn Monaliza Rodrigues Pereira

([https://www.linkedin.com/in/kathellyn-monaliza-rodrig%C3%AAs-pereira-6a304622b?utm\\_source=share&utm\\_campaign=share\\_via&utm\\_content=profile&utm\\_medium=android\\_app](https://www.linkedin.com/in/kathellyn-monaliza-rodrig%C3%AAs-pereira-6a304622b?utm_source=share&utm_campaign=share_via&utm_content=profile&utm_medium=android_app))

Mariane Aparecida da Silva Ferreira

([https://www.linkedin.com/in/mariane-aparecida-da-silva-ferreira-1b8152175?utm\\_source=share&utm\\_campaign=share\\_via&utm\\_content=profile&utm\\_medium=android\\_app](https://www.linkedin.com/in/mariane-aparecida-da-silva-ferreira-1b8152175?utm_source=share&utm_campaign=share_via&utm_content=profile&utm_medium=android_app))

Nicolas Alves de Souza

([https://www.linkedin.com/in/nicolas-alves-650054172?utm\\_source=share&utm\\_campaign=share\\_via&utm\\_content=profile&utm\\_medium=android\\_app](https://www.linkedin.com/in/nicolas-alves-650054172?utm_source=share&utm_campaign=share_via&utm_content=profile&utm_medium=android_app))

William Henrique Vieira Leão

([https://www.linkedin.com/in/william-henrique-vieira-le%C3%A3o-4426a71a1?utm\\_source=share&utm\\_campaign=share\\_via&utm\\_content=profile&utm\\_medium=android\\_app](https://www.linkedin.com/in/william-henrique-vieira-le%C3%A3o-4426a71a1?utm_source=share&utm_campaign=share_via&utm_content=profile&utm_medium=android_app))

Professor M2 ou Orientador: Marcus Vinicius do Nascimento

Professor P2: Jean Carlos Lourenco Costa

## Resumo do projeto:

Este projeto visa fazer uma análise profunda de rotas, custos e produtividade de uma empresa embarcadora, com o auxílio das tecnologias, Jira Software, Github, Power Bi, SQL e Python. O objetivo deste projeto é proporcionar à empresa uma abordagem estratégica para otimizar suas operações de transporte, melhorar a eficiência e reduzir os custos logísticos, resultando em um serviço mais ágil. A plataforma Jira Software permite com que, a equipe em colaboração gerencie as tarefas e as entregas para as sprints negociadas com o cliente. A tecnologia GitHub é um sistema de controle de versão distribuído, o que significa que permite que várias pessoas trabalhem em um mesmo projeto ao mesmo tempo, mantendo um histórico de todas as alterações feitas no código, permitindo que eles trabalhem em conjunto no mesmo código, revisem as alterações uns dos outros e forneçam feedback por meio de pull requests e comentários. Para a demonstração de dados a ferramenta utilizada é o PowerBi, permitindo visualizar, analisar e compartilhar dados relevantes do projeto, proporcionando uma tomada de decisão mais eficaz e uma execução mais bem-sucedida do projeto.

## Abstract:

This project aims to carry out an in-depth analysis of routes, costs and productivity of a shipping company, with the help of technologies, Jira Software, Github, Power Bi, SQL and Python. The objective of this project is to provide the company with a strategic approach to optimize its transport operations, improve efficiency and reduce logistics costs, resulting in a more agile service. The Jira Software platform allows the collaborative team to manage tasks and deliveries for sprints negotiated with the client. GitHub technology is a distributed version control system, which means it allows multiple people to work on the same project at the same time, keeping a history of all changes made to the code, allowing them to work together on the same code, Review each other's changes and provide feedback through pull requests and comments. To demonstrate data, the tool used is PowerBi, allowing you to view, analyze and share relevant project data, providing more effective decision-making and more successful execution of the project.

## **1. Contextualização do projeto**

O setor de transporte se enquadra de maneira importante na economia global, pois por meio deste mercado tanto o movimento de pessoas quanto de mercadorias é facilitado em uma escala sem fronteiras e limitações geográficas.

Contudo, as empresas que atuam por meio do transporte de carga têm enfrentado desafios tanto administrativos quanto logísticos, onde qualquer falha na operação pode vir a ocasionar comprometimento no funcionamento gerando resultados negativos ao cliente.

Tendo em vista que no Brasil a redução de custos na logística de transporte é um dos desafios constantes as empresas deste ramo, este projeto visa com base em uma situação que nos é dada, efetuar a coleta de dados e por meio da análise desenvolver programação matemática que selecionara um melhor elemento, com relação ao critério que nos disponibilizara alternativas que serão trabalhadas entregando um produto ao cliente.

## **2. Objetivos**

Os objetivos estabelecidos para esta sprint consistem em:

- i) Análise profunda de rotas, custos e produtividade de uma empresa embarcadora;
- ii) Organização, atribuição e execução de tarefas para desenvolvimento do produto que visa melhor eficiência, redução dos custos logísticos e serviço mais ágil;
- iii) Criação de um visualizador de indicadores em BI, bem como dashboard;
- iv) Aplicação do método de transporte para otimização da distribuição;
- v) Desenvolvimento em ferramenta SQL para execução de banco de dados;

## **3. Métodos analíticos e tecnologias utilizadas**

O nível de precisão, otimização e assertividade conferido pela tecnologia nas empresas é fundamental até mesmo para momentos de crise, em que são necessárias respostas rápidas para sobreviver perante as adversidades externas. Dado isso para organização do projeto e desenvolvimento do produto utilizamos as seguintes tecnologias.

### *3.1. Métodos analíticos utilizados*

Com base no problema que nos foi dado utilizaremos o Método de Transporte que trata-se do modelo de programação linear preocupado em minimizar os custos envolvidos no fornecimento dos requerimentos em diversos lugares de diversas fontes com diferentes custos relativos às várias combinações de fonte e locais de requerimento, envolvendo cliente, fornecedor e rota.

Uma vez que a programação otimiza e resolve problemas de transporte, direcionando a uma tomada de decisão bem como processos e alocação de recursos onde o produto deseja maximizar os lucros e minimizar os gastos de recurso, esta se aplica em nosso desenvolvimento.

Também utilizaremos a planilha “VRP Spreadsheet” como uma alternativa inovadora e acessível para lidar com os desafios complexos do roteamento de veículos. Desenvolvida como uma planilha no Microsoft Excel, essa aplicação de código aberto oferece uma abordagem flexível e intuitiva para representar, resolver e visualizar os resultados dos problemas de roteamento de veículos.

O principal propósito do VRP é otimizar a distribuição de recursos, como veículos e rotas, visando minimizar custos e maximizar a eficiência operacional. Ao possibilitar que os usuários criem e ajustem facilmente seus próprios modelos de roteamento, essa ferramenta se destaca pela sua capacidade de se adaptar a uma variedade de cenários logísticos.

Integrar o “Bing Maps” não só facilita a criação de rotas, mas também disponibiliza informações úteis, como a estimativa de tempo de viagem e a distância percorrida. Esses dados são fundamentais para melhorar o planejamento das rotas.

### 3.2. Tecnologias da Informação

A tecnologia da informação na logística facilita a gestão e os processos de uma organização relacionados para o transporte de suas mercadorias, assim como aumento da produtividade e cumprimento de requisitos.

Para organização entre os desenvolvedores do projeto na execução de suas atividades de maneira estratégica em pró de uma melhor produtividade e resultados para entrega do produto é utilizada as seguintes tecnologias abaixo:

- i) Jira Software para o monitoramento de tarefas e acompanhamento de projetos garantindo o gerenciamento de todas as suas atividades em único lugar.
- ii) GitHub para controle de versão dos arquivos assim como permissão para que os programadores contribuam no projeto.

Para desenvolvimento do produto como proposta de solução, de forma a ser aplicável ao cliente, como apoio de estudo, análise e execução de ferramenta é utilizadas as seguintes tecnologias abaixo:

- i) Power BI para avaliação e visualização dos dados coletados por meio de relatórios e dashboard.
- ii) Python para resolução do problema de otimização em programação linear.
- iii) SQL para realização de consultas dos dados necessários.

## 4. Coleta e descrição dos dados utilizados

### 4.1. Backlog Total

Na figura abaixo contemos o Backlog Total que relaciona as demandas atuais de serviço e a capacidade para atendê-las, bem como sua estimativa de tempo de realização de tarefas. Estas estão em ordem de trabalho que estão planejadas e programadas, sua lista consiste na User Story.

Rank	Prioridade	User Story	Estimativa	Sprint
1	Alta	Eu como desenvolvedor, quero analisar as rotas através das planilhas disponibilizadas para melhor organizar os dados	50m	1
2	Alta	Eu como desenvolvedora, quero utilizar como exemplo a Guia de Artefatos de Requisitos Ágeis para gerar o Backlog de Produto	1d	1
3	Média	Eu como desenvolvedor quero assistir vídeos de passo a passo que me orientem para criar o Github do projeto	1h45m	1

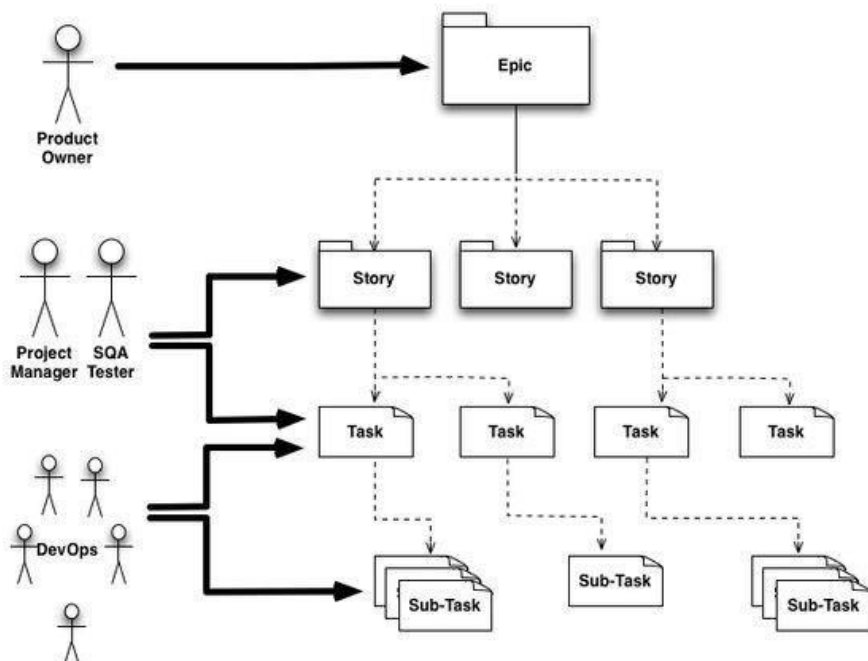
4	Média	Eu como desenvolvedor quero assistir vídeo aulas que me orientem para transformação dos dados em Power BI	2h	1
5	Alta	Eu como desenvolvedora quero reunir todas as informações dispostas no Backlog para realizar o relatório da Sprint 1 do Projeto	1h	1
6	Alta	Eu como desenvolvedora quero criar uma tela de evolução dos custos em KM (em BI)	1d	2
7	Alta	Eu como desenvolvedora quero preparar e apresentar um resumo da pesquisa para a equipe de desenvolvimento	40m	2
8	Alta	Eu como desenvolvedora quero reunir todas as informações dispostas no Backlog para realizar o relatório da Sprint 2 do Projeto	1h	2

9	Alta	Eu como desenvolvedora quero documentar os requisitos discutidos na reunião e compartilhá-los com a equipe	1h	2
10	Alta	Eu como desenvolvedora quero relacionar os dados em SQL	1d	3
11	Alta	Eu como desenvolvedora quero desenvolver a funcionalidade de importação de dados para permitir o carregamento das informações de entrega no sistema.	1d	3
12		Eu como desenvolvedora quero desenvolver a tela de evolução dos custos por unidade transportada (em BI)	2h	3
13	Média	Eu como desenvolvedora quero projetar a arquitetura do sistema, incluindo a escolha de algoritmos de otimização adequados e a estrutura de dados para armazenar as informações das rotas.	2d	3
14	Alta	Eu como desenvolvedora quero integrar o sistema com APIs de mapas e serviços de tráfego para obter informações atualizadas sobre as condições da estrada.	3h	3
15	Alta	Eu como desenvolvedora quero estabelecer uma reunião com os stakeholders para revisar e validar os requisitos do sistema, garantindo que todas as necessidades sejam compreendidas e consideradas.	40m	3
16	Alta	Eu como desenvolvedora quero reavaliar e ajustar a prioridade das funcionalidades antes de cada sprint com base no feedback do cliente, nas mudanças nos requisitos e no progresso realizado até o momento.	5h	3
17	Alta	Eu como desenvolvedora quero reunir todas as informações dispostas no Backlog para realizar o relatório da Sprint 3 do Projeto	1h	3
18	Alta	Eu como desenvolvedora quero fazer uma aplicação de otimização com base nas ferramentas disponibilizadas	1h	4
19	Alta	Eu como desenvolvedora quero realizar testes de aceitação e integração do sistema, garantindo que todas as funcionalidades estejam operacionais e atendam aos requisitos do cliente.	2d	4

20	Média	Eu como desenvolvedora quero identificar e corrigir quaisquer problemas ou bugs encontrados durante os testes.	2h	4
21		Realizar melhorias na interface de usuário com base no feedback dos usuários finais.	1d	4
22	Média	Eu como desenvolvedora quero fazer uma aplicação do contexto de Produtividade	2h	4
23	Alta	Eu como desenvolvedora quero reunir todas as informações dispostas no Backlog para realizar o relatório da Sprint 4 do Projeto	1h	4
24	Alta	Preparar o ambiente de produção e implantar a versão final do sistema	1h	4

#### 4.2. Backlog da Sprint

Dado ao problema que estamos desenvolvendo realizamos análises em busca da solução e desenvolvimento do produto, tivemos de avaliar o que faríamos como alavanque inicial do que executaríamos ou não, dado a isso o planejamento requer divisão de quatro sprints, após uma discussão, foram distinguidas as Stories e atribuições a cada integrante em cada sprint, no qual conforme o andamento são distinguidas as Task para entrega final das tarefas por cada integrante responsável.



#### 4.3. Burndown da Sprint 1

Para melhor acompanhamento da execução e conclusão da entrega das atividades atribuídas utilizamos o gráfico de burndown para calcular de forma eficiente se a equipe terá tempo hábil para concluir o trabalho, além deste determinar a data de conclusão do projeto, nos oferece insights sobre o funcionamento da equipe.

Há dois tipos de gráficos de burndown o de Agile e de Sprint, para nosso projeto aplicamos o de Sprint já que este é usado para Sprint curtas, conforme modelo na imagem abaixo, onde traçamos o

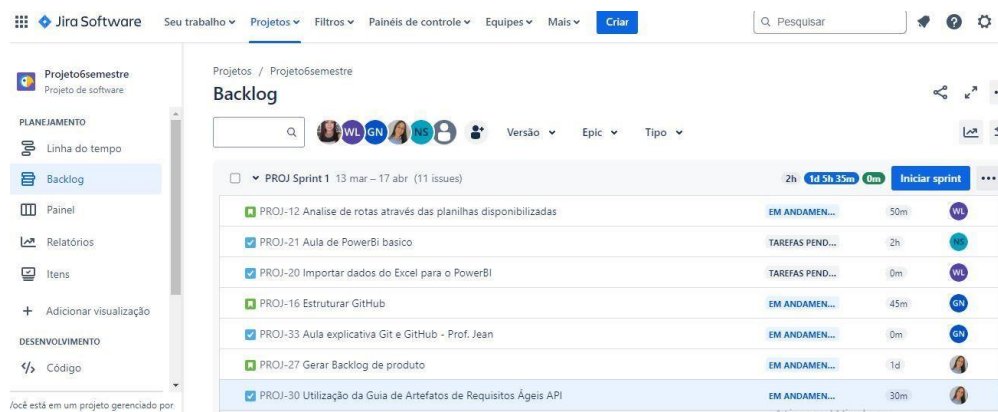
volume estimado do trabalho que precisa ser feito em relação ao tempo necessário para concluir este trabalho, colaborando para o planejamento.



- Eixo X (horizontal): é o eixo que representa o tempo que resta para concluir o projeto. Apresentado em dias.
- Eixo Y (vertical): é o eixo que representa o esforço remanescente necessário para concluir o projeto.
- Linha de trabalho real: representa o trabalho remanescente real. Onde temos imprevistos que surgiram resultando em um aumento de tempo necessário para concluir o trabalho. Esta consta menos linear devido a problemas no projeto e trabalhos não previstos.
- Linha de trabalho remanescente ideal (trabalho estimado): representa o volume de trabalho estimado no cenário ideal. Sendo uma trajetória mais reta do que a linha de trabalho real.

#### 4.4. Tecnologias empregadas

- Jira é um software permite com que, a equipe em colaboração gerencie as tarefas e as sprints negociadas com o cliente.

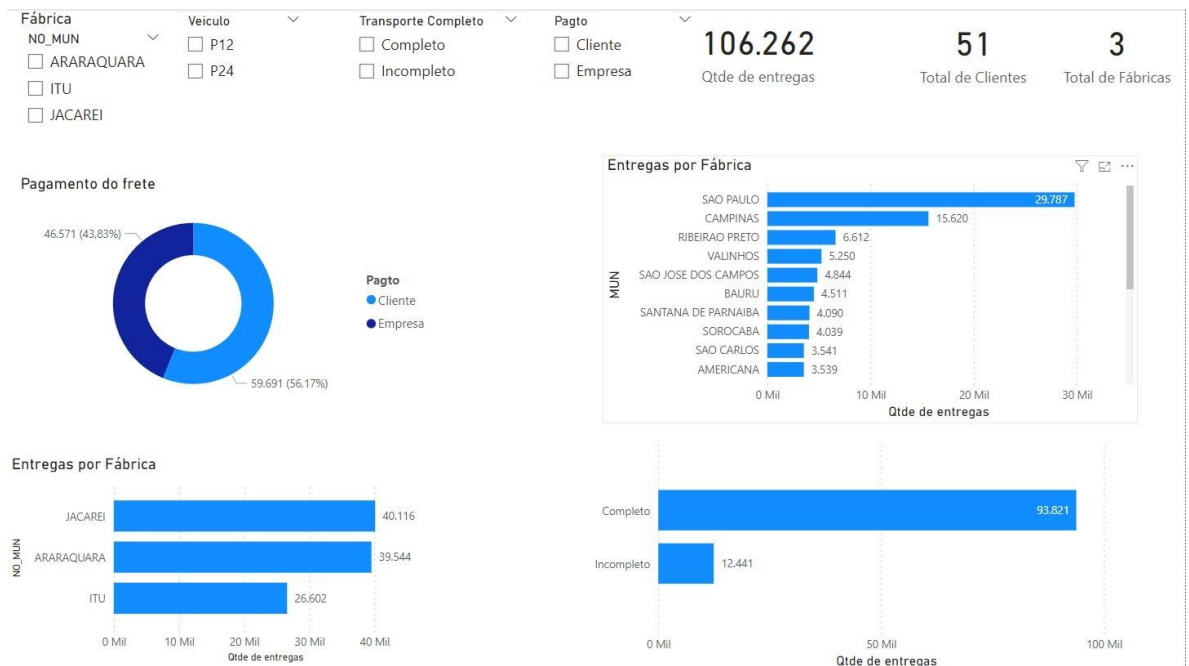


- A tecnologia GitHub é um sistema de controle de versão distribuído, o que significa que permite várias pessoas de trabalharem em um mesmo projeto ao mesmo tempo, mantendo um histórico de todas alterações feitas no código, permitindo que eles



trabalhem em conjunto no mesmo código, revisando a alteração uns dos outros e fornecendo feedback por meio de pull requests e comentários.

- Para demonstração de dados utilizamos a ferramenta Power BI permitindo visualizar, analisar e compartilhar dados relevantes do projeto, proporcionando uma tomada de decisão mais eficaz e uma execução mais bem sucedida do projeto, como o modelo desenvolvido abaixo.



- O Python contém uma linguagem projetada para dar ênfase no trabalho do desenvolvedor, facilitando a escrita de um código limpo, simples e legível, tanto em aplicações menores quanto em programas mais complexos.
- Sublime Text é um editor de código-fonte multi-plataforma e shareware com uma interface de programação de aplicativos para a linguagem Python. É um editor de código com interface minimalista e fácil de entender, sendo possível utilizar tanto em computadores quanto para celulares. É adaptável e pode servir para diferentes tipos de profissionais. Além de contar com plugins que oferecem recursos exclusivos para auxiliar no desenvolvimento, devido a isso utilizamos para desenvolvimento de código conforme imagem abaixo.

```

C:\Users\geova\Downloads\api3 (1).py - Sublime Text (UNREGISTERED)
File Edit Selection Find View Goto Tools Project Preferences Help

api3 (1).py
1 import pandas as pd
2 import numpy as np
3 import matplotlib as plt
4
5 #define o caminho do arquivo no computador
6 file = 'C:/Users/Pichau/Downloads/'
7
8 #abre os arquivos necessários
9 df = pd.read_csv(file+'Rotas.csv', sep=',', decimal='.',)
10 df2 = pd.read_csv(file+'Clientes.csv', sep=',', decimal='.',)
11 df3 = pd.read_csv(file+'Fabricas.csv', sep=',', decimal='.', encoding='latin1')
12
13 #checagem dos arquivos
14 #1
15 df.info()
16 #2
17 df2.info()
18 #3
19 df3.info()
20 #integra o df de Rotas com o df de Clientes e Fabricas
21 rotas_final = df.merge(df2, left_on='co_cliente',
22                        right_on='co_cliente',
23                        how='inner').merge(df3, left_on='co_fabrica',
24                                         right_on='co_fabrica',
25                                         how='inner')
26
27 #checagem do df final
28 rotas_final.info()
29 #visualiza o df final
30 rotas_final.head(5)
31 #cria nova coluna com condição
32 rotas_final['capacidade'] = None
33 rotas_final.loc[rotas_final['veiculo'] == 'P24', 'capacidade'] = 3600
34 rotas_final.loc[rotas_final['veiculo'] == 'P12', 'capacidade'] = 1800
35
36 #cria nova coluna com cálculo
37 rotas_final['produtividade'] = rotas_final['qtd.transp'] / rotas_final['capacidade'] * 100
38 #analises
39 #filtros
40
41 fabricas = rotas_final['co_fabrica'].unique()
42 print(fabricas)
43
44 #análise segmentada de uma fábrica
45 co_fabrica = 3423999
46 incoterm = 'CIF'
47
48 df_filtrado = rotas_final[(rotas_final['co_fabrica'] == co_fabrica) &
49                          (rotas_final['incoterm'] == incoterm)]
50 clientes = df_filtrado['co_cliente'].unique()
51 print(clientes)
52
53 #exportar a lista de clientes
54 with open('clientes.txt', 'w') as file:
55     for client in clientes:
56         file.write(str(client) + '\n')
57
58
59 #frete médio por fabrica e cliente
60
61 df_filtrado['frete/unidade'] = df_filtrado['Vir.frete'] / df_filtrado['qtd.transp']
62 frete_media = df_filtrado.groupby(['co_fabrica', 'co_cliente'])['frete/unidade'].mean().reset_index()
63 frete_media.info()
64 print(frete_media)
65
66 frete_media.to_csv('frete.csv', index=False)
67
68 #exporta o novo df para csv
69 rotas_final.to_csv('rotas_final.csv', index=False)

```

- SQL é uma linguagem de programação para armazenar e processar informações em um banco de dados relacional. No qual armazena informações em formato tabular, com linhas e colunas representando diferentes atributos de dados e as várias relações entre os valores dos dados. Nele pode se atualizar, remover, pesquisar e recuperar informações do banco de dados. Também pode usar SQL para manter e otimizar a performance do banco de dados.

## 5. Desenvolvimento de ferramentas

### 5.1. VRP Spreadsheet

O VRP foi usado no projeto para unificar dados das planilhas "rotas", "fábricas" e "clientes", aproveitando as informações de todas as latitudes e longitudes disponíveis da planilha "Rotas". Com a integração desses dados, o objetivo da tecnologia é entregar as distâncias em quilômetros e o tempo estimado entre uma latitude e longitude para cada rota planejada.

Essa abordagem permite uma análise abrangente e precisa das rotas disponíveis, levando em consideração não apenas a distância física entre os pontos, mas também as condições de tráfego e outras variáveis relevantes. Com esses dados pode-se calcular o valor do frete, mostrando para o cliente o R\$/km.

VRP Sanjalog - Excel

Arquivo Página Inicial Inserir Layout da Página Fórmulas Dados Revisão Exibir Desenvolvedor Ajuda VRP Spreadsheet Solver

0. Reset 1. Locations 2. Distances 3. Vehicles 4. Solution 5. Visualization 6. Solver Info

Sequence	Parameter	Value	Remarks
0.Interface	Language	English	Please refer to the manual for modifying the interface.
	Optional - Bing Maps Key	Ai8zKj0Be-pz8IwEz1n4LN5dUZLEldybA0i0s5NRO7xiMqLOlbus_xcisLz8oqXz	You can get a free trial key at <a href="https://www.bingmapsportal.com/">https://www.bingmapsportal.com/</a>
1.Locations	Number of depots	3	[1, 20]
	Number of customers	51	[5, 200]
2.Distances	Distance computation method	Bing Maps driving distances (km)	Recommendation: Use 'postcode, country' format for addresses
	Duration computation method	Bing Maps driving durations	
	Bing Maps travel mode	Driving	
	Bing Maps route type	Fastest	Recommendation: Use 'Fastest'
	Bing Maps route detail level	3	[0, 10]
	Average vehicle speed	70	
3.Vehicles	Number of vehicle types	1	
4.Solution	Do the vehicles return to their depot(s)?	Yes - only once at the end	
	Time window type	Hard	
	Backhauls?	No	If activated, delivery locations must be visited before pickup locat
5.Optional - Visualization	Visualization background	Bing Maps	
	Location labels	Location IDs	

VRP Solver Console 1.Locations 2.Distances

## 5.2. Power BI

O Power BI é uma plataforma de análise de informações e dados com visualização de negócios desenvolvida pela Microsoft. Ele permite que os usuários extraiam dados de várias fontes, como bancos de dados, serviços na nuvem e arquivos locais, e os transformem em insights visuais facilmente compreensíveis, como gráficos, tabelas e painéis interativos. Com o Power BI, os usuários podem criar relatórios personalizados, compartilhá-los com colegas e colaborar em análises, ajudando as organizações a tomar decisões informadas com base em dados.

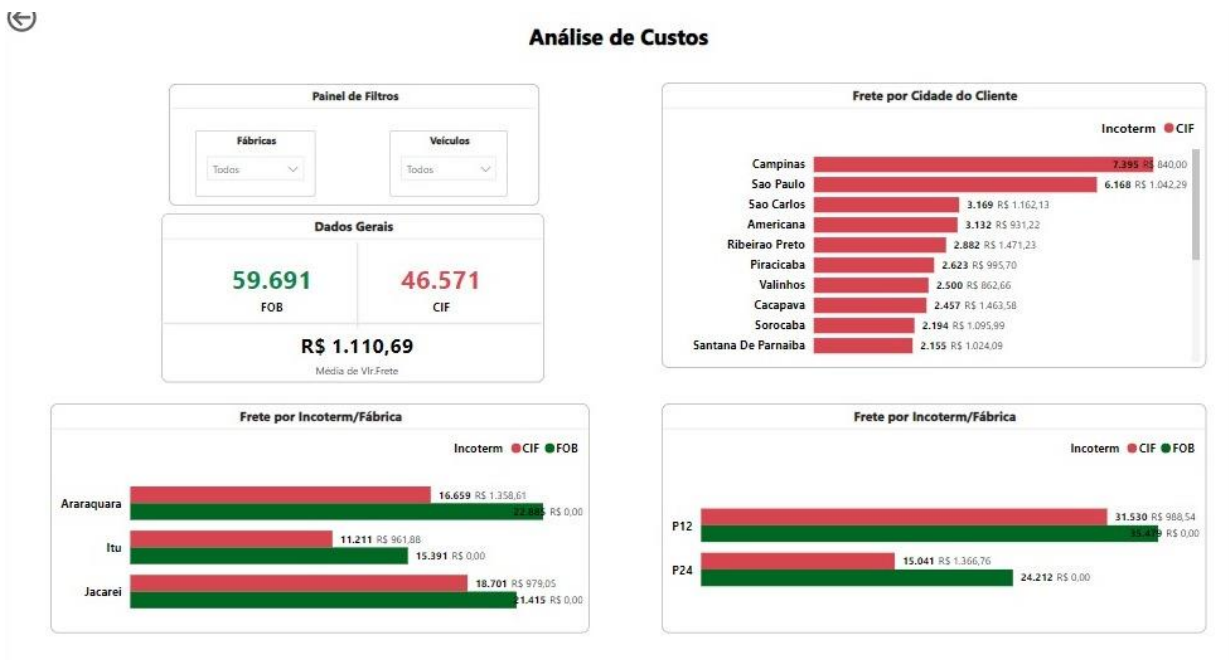
Foi desenvolvido nesta primeira imagem abaixo, alguns gráficos de colunas. Onde podemos ver separadamente o valor acumulado das entregas que foram e CIF e FOB por fábrica e o valor unitário por fábrica, logo à direita é possível ver o tipo de caminhão, sendo assim de 12 paletes ou de 24 paletes, da mesma forma como as fábricas nos mostra o valor total do CIF e FOB e também do valor unitário.



Nesta segunda imagem ao lado esquerdo desenvolvemos um gráfico onde facilita a visualização do valor total em CIF do frete por cidade, e logo à direita nos mostra o custo por cidade do cliente e em ambas podemos ver que aquilo dera é Campinas e logo em seguida São Paulo a maior cidade do Brasil.



E nestas duas últimas imagens podemos visualizar de forma dinâmica mais uma forma de demonstrar analises de custos de forma geral e por unidade.



### Análise de custos por unidade



### 5.3. Python

Python é uma ferramenta de programação amplamente utilizada no campo acadêmico para análise de dados. Com suas bibliotecas especializadas, como Pandas, NumPy e Matplotlib, é uma ferramenta poderosa para coletar, processar e visualizar dados em pesquisas científicas. Sua sintaxe simples e expressiva torna Python uma escolha popular entre os acadêmicos, permitindo a implementação eficiente de algoritmos e a realização de análises estatísticas avançadas.

No contexto deste projeto, Python é utilizado especificamente para análise de dados, onde trabalhamos com as planilhas fornecidas pelo cliente. Por meio da criação de códigos específicos, realizamos a integração desses dados, gerando resultados de extrema importância para o projeto.

Durante a segunda Sprint, desenvolvemos sete códigos com os seguintes objetivos:

1. Comparação dos custos de frete entre as fábricas e os clientes.
2. Cálculo dos custos totais de frete CIF e FOB, bem como a comparação entre eles.
3. Comparação dos custos de frete CIF e FOB por código de fábrica.
4. Criação de códigos que geram todos os resultados em uma planilha do Excel.

Essas atividades contribuíram significativamente para o avanço do projeto, fornecendo insights valiosos para a tomada de decisões.



Anexo, as imagens dos códigos criados:

```

Untitled3 (1).ipynb × Welcome
C: > Users > Noite > Downloads > Untitled3 (1).ipynb > ...
+ Code + Markdown ...
Select Kernel

import pandas as pd

# Carregar os dados das planilhas
Fabricas_df = pd.read_csv(r'C:\Users\Noite\Downloads\Fabricas.csv', encoding='latin1')
Clientes_df = pd.read_csv(r'C:\Users\Noite\Downloads\Clientes.csv', encoding='latin1')
Rotas_df = pd.read_csv(r'C:\Users\Noite\Downloads\Rotas.csv', encoding='latin1')

# Selecionar o valor do frete da primeira rota
Vlr_Frete = Rotas_df['Vlr.Frete'].iloc[0]

# Comparar os custos de frete entre fábricas e clientes
custo_fabricas = Fabricas_df['CO.Fabrica'].sum()
custo_clientes = Clientes_df['CO.Cliente'].sum()

# Exibir os resultados
print("Custo total de frete para as fábricas:", custo_fabricas)
print("Custo total de frete para os clientes:", custo_clientes)

# Selecionar o valor do frete da primeira rota
Vlr_Frete = Rotas_df['Vlr.Frete'].iloc[0]

[4] Python

... Custo total de frete para as fábricas: 10251519
Custo total de frete para os clientes: 118626

Restricted Mode 0 0 0 CRLF Cell 7 of 14

```

```

Untitled3 (1).ipynb × Welcome
C: > Users > Noite > Downloads > Untitled3 (1).ipynb > ...
+ Code + Markdown ...
Select Kernel

import pandas as pd

# Carregar os dados das planilhas
Fabricas_df = pd.read_csv(r'C:\Users\Noite\Downloads\Fabricas.csv', encoding='latin1')
Clientes_df = pd.read_csv(r'C:\Users\Noite\Downloads\Clientes.csv', encoding='latin1')
Rotas_df = pd.read_csv(r'C:\Users\Noite\Downloads\Rotas.csv', encoding='latin1')

# Selecionar o valor do frete da primeira rota
Vlr_Frete = Rotas_df['Vlr.Frete'].iloc[0]

# Comparar os custos de frete entre fábricas e clientes
custo_fabricas = Fabricas_df['CO.Fabrica'].sum()
custo_clientes = Clientes_df['CO.Cliente'].sum()

# Exibir os resultados
print("Custo total de frete para as fábricas:", custo_fabricas)
print("Custo total de frete para os clientes:", custo_clientes)

[5] Python

... Custo total de frete para as fábricas: 10251519
Custo total de frete para os clientes: 118626

```

```

Untitled3 (1).ipynb X Welcome
C: > Users > Noite > Downloads > Untitled3 (1).ipynb > ...
+ Code + Markdown ... Select Kernel

import pandas as pd

# Carregar os dados da planilha de Rotas
Rotas_df = pd.read_csv(r'C:\Users\Noite\Downloads\Rotas.csv', encoding='latin1')

# Selecionar o valor do frete da primeira rota
Vlr_Frete = Rotas_df['Vlr.Frete'].iloc[0]
Incoterm = Rotas_df['Incoterm'].iloc[0]

# Comparar os custos de frete entre CIF e FOB
if Incoterm == 'CIF':
    print("O custo total de frete CIF é:", Vlr_Frete)
elif Incoterm == 'FOB':
    print("O custo total de frete FOB é:", Vlr_Frete)
else:
    print("O Incoterm da primeira rota não é definido como CIF ou FOB.")

[6] Python
... O custo total de frete FOB é: 0.0

```

```

Untitled3 (1).ipynb X Welcome
C: > Users > Noite > Downloads > Untitled3 (1).ipynb > ...
+ Code + Markdown ... Select Kernel

import pandas as pd

# Carregar os dados da planilha de Rotas
Rotas_df = pd.read_csv(r'C:\Users\Noite\Downloads\Rotas.csv', encoding='latin1')

# Selecionar os dados relevantes
Incoterm = Rotas_df['Incoterm']
Qtd_pallets = Rotas_df['Qtd/pallets']
Qtd_transp = Rotas_df['Qtd.Transp']
Vlr_Frete = Rotas_df['Vlr.Frete']

# Calcular os custos totais de frete CIF e FOB
custo_total_CIF = sum(Qtd_pallets[Incoterm == 'CIF'] * Qtd_transp[Incoterm == 'CIF'] * Vlr_Frete[Incoterm == 'CIF'])
custo_total_FOB = sum(Qtd_pallets[Incoterm == 'FOB'] * Qtd_transp[Incoterm == 'FOB'] * Vlr_Frete[Incoterm == 'FOB'])

# Comparar os custos de frete CIF e FOB
if custo_total_CIF > custo_total_FOB:
    print("O custo total de frete CIF é maior do que o custo total de frete FOB.")
elif custo_total_CIF < custo_total_FOB:
    print("O custo total de frete CIF é menor do que o custo total de frete FOB.")
else:
    print("O custo total de frete CIF é igual ao custo total de frete FOB.")

[7] Python

```

```

Untitled3 (1).ipynb X Welcome
C: > Users > Noite > Downloads > Untitled3 (1).ipynb > ...
+ Code + Markdown ...
Select Kernel

import pandas as pd

# Carregar os dados da planilha de Rotas
Rotas_df = pd.read_csv(r'C:\Users\Noite\Downloads\Rotas.csv', encoding='latin1')

# Selecionar os dados relevantes
Incoterm = Rotas_df['Incoterm']
Qtd_pallets = Rotas_df['Qtd/pallets']
Qtd_transp = Rotas_df['Qtd.Transp']
Vlr_Frete = Rotas_df['Vlr.Frete']

# Calcular os custos totais de frete CIF e FOB
custo_total_CIF = sum(Qtd_pallets[Incoterm == 'CIF'] * Qtd_transp[Incoterm == 'CIF'] * Vlr_Frete[Incoterm == 'CIF'])
custo_total_FOB = sum(Qtd_pallets[Incoterm == 'FOB'] * Qtd_transp[Incoterm == 'FOB'] * Vlr_Frete[Incoterm == 'FOB'])

# Exibir os resultados
print("Custo total de frete CIF:", custo_total_CIF)
print("Custo total de frete FOB:", custo_total_FOB)

[8] Python

... Custo total de frete CIF: 2066694685380.0
Custo total de frete FOB: 242696929428.0

```

```

pip install xlswriter

[10] Python

... Collecting xlswriter
  Downloading XlsxWriter-3.2.0-py3-none-any.whl.metadata (2.6 kB)
  Downloading XlsxWriter-3.2.0-py3-none-any.whl (159 kB)
  ----- 0.0/159.9 kB ? eta -:-:-
  ----- 153.6/159.9 kB 4.6 MB/s eta 0:00:01
  ----- 159.9/159.9 kB 4.8 MB/s eta 0:00:00
Installing collected packages: xlswriter
Successfully installed xlswriter-3.2.0
Note: you may need to restart the kernel to use updated packages.

```

```

Untitled3 (1).ipynb X Welcome
C: > Users > Noite > Downloads > Untitled3 (1).ipynb > ...
+ Code + Markdown ...
Select Kernel

import pandas as pd
import xlswriter

# Carregar os dados da planilha de Rotas
Rotas_df = pd.read_csv(r'C:\Users\Noite\Downloads\Rotas.csv', encoding='latin1')

# Selecionar os dados relevantes
Incoterm = Rotas_df['Incoterm']
Qtd_pallets = Rotas_df['Qtd/pallets']
Qtd_transp = Rotas_df['Qtd.Transp']
Vlr_Frete = Rotas_df['Vlr.Frete']

# Calcular os custos totais de frete CIF e FOB
custo_total_CIF = sum(Qtd_pallets[Incoterm == 'CIF'] * Qtd_transp[Incoterm == 'CIF'] * Vlr_Frete[Incoterm == 'CIF'])
custo_total_FOB = sum(Qtd_pallets[Incoterm == 'FOB'] * Qtd_transp[Incoterm == 'FOB'] * Vlr_Frete[Incoterm == 'FOB'])

# Criar um novo arquivo Excel
nome_arquivo = 'custos_frete.xlsx'
workbook = xlswriter.Workbook(nome_arquivo)
worksheet = workbook.add_worksheet()

# Escrever os resultados na planilha
worksheet.write('A1', 'Custo total de frete CIF (BRL)')
worksheet.write('A2', custo_total_CIF)
worksheet.write('B1', 'Custo total de frete FOB (BRL)')
worksheet.write('B2', custo_total_FOB)

```



```

Untitled3 (1).ipynb X Welcome
C: > Users > Noite > Downloads > Untitled3 (1).ipynb > ...
+ Code + Markdown ...
Select Kernel

# Escrever os resultados na planilha
worksheet.write('A1', 'Custo total de frete CIF (BRL)')
worksheet.write('A2', custo_total_CIF)
worksheet.write('B1', 'Custo total de frete FOB (BRL)')
worksheet.write('B2', custo_total_FOB)

# Fechar o arquivo Excel
workbook.close()

print(f"Os resultados foram salvos em '{nome_arquivo}'")

[11] Python
... Os resultados foram salvos em 'custos_frete.xlsx'.

import pandas as pd
import xlswriter

# Carregar os dados da planilha de Rotas
Rotas_df = pd.read_csv(r'C:\Users\Noite\Downloads\Rotas.csv', encoding='latin1')

# Selecionar os dados relevantes
Incoterm = Rotas_df['Incoterm']
Qtd_pallets = Rotas_df['Qtd/pallets']

```

```

Untitled3 (1).ipynb X Welcome
C: > Users > Noite > Downloads > Untitled3 (1).ipynb > ...
+ Code + Markdown ...
Select Kernel

# Selecionar os dados relevantes
Incoterm = Rotas_df['Incoterm']
Qtd_pallets = Rotas_df['Qtd/pallets']
Qtd_transp = Rotas_df['Qtd.Transp']
Vlr_Frete = Rotas_df['Vlr.Frete']

# Calcular os custos totais de frete CIF e FOB
custo_total_CIF = sum(Qtd_pallets[Incoterm == 'CIF'] * Qtd_transp[Incoterm == 'CIF'] * Vlr_Frete[Incoterm == 'CIF'])
custo_total_FOB = sum(Qtd_pallets[Incoterm == 'FOB'] * Qtd_transp[Incoterm == 'FOB'] * Vlr_Frete[Incoterm == 'FOB'])

# Criar um novo arquivo Excel
nome_arquivo = 'comparativo_custos_frete.xlsx'
workbook = xlswriter.Workbook(nome_arquivo)
worksheet = workbook.add_worksheet()

# Escrever os resultados na planilha
worksheet.write('A1', 'Custo total de frete CIF (BRL)')
worksheet.write('A2', custo_total_CIF)
worksheet.write('B1', 'Custo total de frete FOB (BRL)')
worksheet.write('B2', custo_total_FOB)

# Comparar os custos totais de frete CIF e FOB
if custo_total_CIF > custo_total_FOB:
    resultado = "Custo total de frete CIF maior do que o custo total de frete FOB."
elif custo_total_CIF < custo_total_FOB:
    resultado = "Custo total de frete FOB maior do que o custo total de frete CIF."
else:
    resultado = "Custos totais de frete CIF e FOB são iguais."

print(resultado)

```

```

Untitled3 (1).ipynb × Welcome
C: > Users > Noite > Downloads > Untitled3 (1).ipynb > ...
+ Code + Markdown ...
Select Kernel

# Comparar os custos totais de frete CIF e FOB
if custo_total_CIF > custo_total_FOB:
    resultado = "Custo total de frete CIF é maior do que o custo total de frete FOB."
elif custo_total_CIF < custo_total_FOB:
    resultado = "Custo total de frete CIF é menor do que o custo total de frete FOB."
else:
    resultado = "Custo total de frete CIF é igual ao custo total de frete FOB."

worksheet.write('A4', resultado)

# Fechar o arquivo Excel
workbook.close()

print(f"Os resultados foram salvos em '{nome_arquivo}'")

[12] Python
... Os resultados foram salvos em 'comparativo_custos_frete.xlsx'.

import pandas as pd
import xlswriter

# Carregar os dados da planilha de Rotas
Rotas_df = pd.read_csv(r'C:\Users\Noite\Downloads\Rotas.csv', encoding='latin1')

```

```

Untitled3 (1).ipynb × Welcome
C: > Users > Noite > Downloads > Untitled3 (1).ipynb > ...
+ Code + Markdown ...
Select Kernel

import pandas as pd
import xlswriter

# Carregar os dados da planilha de Rotas
Rotas_df = pd.read_csv(r'C:\Users\Noite\Downloads\Rotas.csv', encoding='latin1')

# Selecionar os dados relevantes
Incoterm = Rotas_df['Incoterm']
Qtd_pallets = Rotas_df['Qtd/pallets']
Qtd_transp = Rotas_df['Qtd.Transp']
Vlr_Frete = Rotas_df['Vlr.Frete']
Codigo_Fabrica = Rotas_df['CO.Fabrica']

# Calcular os custos totais de frete CIF e FOB por código de fábrica
custos_por_fabrica = Rotas_df.groupby('CO.Fabrica').apply(lambda x: pd.Series({
    'Custo_total_CIF': (x.loc[x['Incoterm'] == 'CIF', 'Qtd/pallets'] *
                       x.loc[x['Incoterm'] == 'CIF', 'Qtd.Transp'] *
                       x.loc[x['Incoterm'] == 'CIF', 'Vlr.Frete']).sum(),
    'Custo_total_FOB': (x.loc[x['Incoterm'] == 'FOB', 'Qtd/pallets'] *
                       x.loc[x['Incoterm'] == 'FOB', 'Qtd.Transp'] *
                       x.loc[x['Incoterm'] == 'FOB', 'Vlr.Frete']).sum()
}))

# Criar um novo arquivo Excel
nome_arquivo = 'custos_frete_por_fabrica.xlsx'
workbook = xlswriter.Workbook(nome_arquivo)

```

```

x.loc[x['Incentivo'] == 'FOB', ['Frete']].sum()
)))

# Criar um novo arquivo Excel
nome_arquivo = 'custos_frete_por_fabrica.xlsx'
workbook = xlswriter.Workbook(nome_arquivo)
worksheet = workbook.add_worksheet()

# Escrever os resultados na planilha
row = 0
col = 0
for fabrica, custos in custos_por_fabrica.iterrows():
    worksheet.write(row, col, f'Código de Fábrica: {fabrica}')
    worksheet.write(row + 1, col, 'Custo total de frete CIF (BRL)')
    worksheet.write(row + 1, col + 1, custos['Custo_total_CIF'])
    worksheet.write(row + 2, col, 'Custo total de frete FOB (BRL)')
    worksheet.write(row + 2, col + 1, custos['Custo_total_FOB'])
    row += 4

# Fechar o arquivo Excel
workbook.close()

print(f"Os resultados foram salvos em '{nome_arquivo}'.")

```

[13] Python

... Os resultados foram salvos em 'custos\_frete\_por\_fabrica.xlsx'.

Restricted Mode 0/0 0/0 Spaces: 4 CRLF Cell 11 of 14

#### 5.4. SQL Workbench

A linguagem SQL serve para interagir com os nossos dados que estão armazenados dentro de um banco de dados relacional, permitindo a criação, a recuperação, a atualização e a exclusão desses dados. Devido a isso realizamos a coleta das informações que constam em Excel de uma empresa logística e a exportamos para ferramenta SQL, sendo informações do cliente, fabrica e rota, por meio do banco de dados conseguiremos os seguintes funcionamentos abaixo:

- Mecanismo de autorização para definir se o usuário tem permissão para executar a consulta e manipular os dados, com base em suas credenciais.
- Mecanismo Relacional para criar um plano eficiente de acesso aos dados, otimizando a consulta e verificando consultas semelhantes.
- Mecanismo de Armazenamento no qual executara a instrução SQL, lendo e gravando os dados nos arquivos de armazenamento físico.

Os componentes desenvolvidos com base nas informações contidas foram tabelas\_cliente, tabela\_fabricas, tabela\_rotas para serem usadas no armazenamento de dados em formato tabular, com colunas representando atributos e linhas representando registros, as tabelas são projetadas com base no modelo relacional. Segue abaixo imagem representativa do desenvolvimento.

The screenshot shows the MySQL Workbench interface. On the left, the 'SCHEMAS' pane displays a tree view with 'sakila' and 'sanjalog' databases. Under 'sanjalog', there are tables 'tabela\_clientes', 'tabela\_fabricas', and 'tabela\_rotas'. The 'Columns' pane for 'tabela\_rotas' lists fields like 'id\_veiculo', 'veiculo', 'dt\_emissao', 'dt\_entrega', 'mes\_base', 'ano', 'cliente', 'fabrica', 'incoterm', 'qtde\_pallets', 'qtde\_produto', 'moeda', 'valor\_frete', and 'distancia'.

The main editor shows a SQL query:

```
1 SELECT *
2 FROM tabela_rotas
3 where valor_frete >700
```

The 'Result Grid' displays the query results with columns: id\_veiculo, veiculo, dt\_emissao, dt\_entrega, mes\_base, ano, cliente, fabrica, incoterm, qtde\_pallets, qtde\_produto, moeda, valor\_frete, and distancia. The results show multiple rows of data, with the first few rows having values like 76, 77, 106502, etc.

Below the result grid, the 'Table: tabela\_rotas' structure is shown:

Columns:

- id\_veiculo int AI PK
- veiculo varchar(45)
- dt\_emissao varchar(45)
- dt\_entrega varchar(45)
- mes\_base varchar(40)
- ano varchar(40)
- cliente varchar(45)
- fabrica varchar(45)
- incoterm varchar(40)
- qtde\_pallets varchar(40)
- qtde\_produto varchar(40)
- moeda varchar(40)
- valor\_frete varchar(40)
- distancia varchar(40)

Procedimentos armazenados que refere se aos blocos de código SQL que podem ser definidos e armazenados no banco de dados. Eles são usados para realizar tarefas específicas, como cálculos complexos, validações de dados e processamento personalizado. Segue abaixo imagem representativa do desenvolvimento.

The screenshot shows a SQL script in the MySQL Workbench editor. The script includes the following commands:

```
1 CREATE DATABASE IF NOT EXISTS `sanjalog` /*!40100 DEFAULT CHARACTER SET utf8mb4 COLLATE utf8mb4_0900_ai_ci */ /*!80016 DEFAULT ENCRYPTION='N' */;
2 USE `sanjalog`;
3 -- MySQL dump 10.13 Distrib 8.0.36, for Win64 (x86_64)
4 --
5 -- Host: localhost    Database: sanjalog
6 --
7 -- Server version  8.0.37
8
9 /*!40101 SET @OLD_CHARACTER_SET_CLIENT=@@CHARACTER_SET_CLIENT */;
10 /*!40101 SET @OLD_CHARACTER_SET_RESULTS=@@CHARACTER_SET_RESULTS */;
11 /*!40101 SET @OLD_COLLATION_CONNECTION=@@COLLATION_CONNECTION */;
12 /*!50503 SET NAMES utf8 */;
13 /*!40103 SET @OLD_TIME_ZONE=@@TIME_ZONE */;
14 /*!40103 SET TIME_ZONE='+00:00' */;
15 /*!40014 SET @OLD_UNIQUE_CHECKS=@@UNIQUE_CHECKS, UNIQUE_CHECKS=0 */;
16 /*!40014 SET @OLD_FOREIGN_KEY_CHECKS=@@FOREIGN_KEY_CHECKS, FOREIGN_KEY_CHECKS=0 */;
17 /*!40101 SET @OLD_SQL_MODE=@@SQL_MODE, SQL_MODE='NO_AUTO_VALUE_ON_ZERO' */;
18 /*!40111 SET @OLD_SQL_NOTES=@@SQL_NOTES, SQL_NOTES=0 */;
19
20 --
21 -- Table structure for table `tabela_clientes`
22 --
23
24 DROP TABLE IF EXISTS `tabela_clientes`;
25 /*!40101 SET @saved_cs_client      = @@character_set_client */;
26 /*!50503 SET character_set_client = utf8mb4 */;
27 CREATE TABLE `tabela_clientes` (
28   `id_cliente` int NOT NULL AUTO_INCREMENT,
29   `cod_cliente` varchar(45) DEFAULT NULL,
30   `municipio` varchar(55) DEFAULT NULL,
31   `latitude` varchar(55) DEFAULT NULL,
```

Instruções SQL, como SELECT, INSERT, UPDATE e DELETE, são usadas para manipular os dados em tabelas. As instruções SQL permitem que os usuários realizem consultas, insiram novos dados, atualizem registros existentes e excluam dados. Segue abaixo imagem representativa de Incoterm FOB com valor de frete.

SQL File 5\* x

Don't Limit

```

1 SELECT incoterm, valor_frete
2 from tabela_rotas
3 WHERE incoterm LIKE '%FOB%' and valor_frete > 0

```

Result Grid

Filter Rows:

Export: | Wrap Cell Content: | Fetch rows:

incoterm	valor_frete
FOB	547.75
FOB	781.4
FOB	573.82
FOB	798.6
FOB	625.12
FOB	625.12
FOB	464.8
FOB	359.13
FOB	547.75
FOB	781.4
FOB	781.4
FOB	781.4
FOB	797.6
FOB	427.65
FOB	650.59