



Universidade Estadual de Campinas
FACULDADE DE TECNOLOGIA

PROJETO DE OTIMIZAÇÃO DE BANCO DE DADOS DE CALL CENTER

Danilo Matos

Gustavo Ferreira Lima

Jackson Rodrigues

Lucas Ferreira

Mateus de Almeida Frigo

Curso: Engenharia e Administração de Banco de Dados

Professor: Manoel A. da Silva Jr

Campinas

2024

Sumário

SUMARIZAÇÃO DOS RECURSOS DE OTIMIZAÇÃO NO MODELO STAR SCHEMA	4
PROJETO DE OTIMIZAÇÃO DE BANCO DE DADOS DE CALL CENTER.....	6
INTRODUÇÃO	6
ANÁLISE DAS TABELAS	6
DETALHAMENTO DAS TABELAS	6
DETALHAMENTO DOS ÍNDICES	12
ESTRATÉGIA DETALHADA DE OTIMIZAÇÃO NO MODELO STAR SCHEMA	14
DIAGRAMA DE MODELAGEM.....	15
MODELAGEM ESCOLHIDA.....	16
TIPOS DE DADOS	16
• Seleção Estratégica:	16
• Tipos de Dados Utilizados:.....	17
• Justificativa da Escolha:.....	17
ÍNDICES.....	18
• Criados em Campos Chave:.....	18
• Criados em Campos de Consulta Freqüente:	18
• Justificativa da Utilização:.....	18
DASHBOARD PARA ANÁLISE DETALHADA DO ATENDIMENTO AO CLIENTE	19
VISUALIZAÇÕES ESTRATÉGICAS	19
1. Tempo Médio de Atendimento por Nome do Produto:.....	19
2. Tempo Médio de Atendimento por Nome do Canal:	19
3. Tempo Médio de Atendimento por Nome do Departamento:.....	19
4. Tempo Médio de Atendimento por Nome do Atendente:	20
5. TMA por Motivo de Atendimento	20
6. Total de atendimentos e Mediana da Taxa de Atendimento por Mês:	20
7. Soma do Tempo de Resolução por Nome do Canal:.....	20
IMPORTAÇÃO E RELACIONAMENTO DE TABELAS:	20
KPIS ESTRATÉGICOS:.....	21
QUERY 1 - Quantidade_Atendimentos_Mês_Atendente:.....	21

QUERY 2 - tempo_médio_resolução_motivo:	21
QUERY 3 - Taxa_Resolução_Atendimento:	22
QUERY 4 - Tempo_Medio_Resolução_2023:.....	23
QUERY 5 - Tempo_Medio_Resolução_2024:.....	24
IMAGENS DO DASHBOARD	26
REPOSITÓRIO GITHUB COM INFORMAÇÕES ADICIONAIS	28

SUMARIZAÇÃO DOS RECURSOS DE OTIMIZAÇÃO NO MODELO STAR SCHEMA

Atendimentos - Call Center

Modelo Entidade-Relacionamento no Esquema Estrela

1. Modelagem do Banco de Dados:

- Identificar e definir as entidades e seus relacionamentos com base nos requisitos do sistema.
- Elaborar o modelo Entidade-Relacionamento, incluindo as tabelas principais (Fato Atendimento) e de dimensão (Cliente, Atendente, Motivo, Canal de Atendimento, Produto ou Serviço).
- Garantir a correta definição das chaves primárias e estrangeiras para manter a integridade referencial.

2. Escolhas de Campos e Tipos de Dados:

- Determinar os tipos de dados apropriados para cada campo em todas as tabelas do banco de dados.
- Assegurar a normalização do banco de dados para evitar redundância e inconsistência.
- Validar a consistência dos dados e garantir a precisão das informações armazenadas.

3. Otimização do Banco de Dados e Criação da Modelagem (Tabelas e Relacionamentos - Engenharia Reversa):

- Identificar oportunidades de otimização no banco de dados, como a criação de índices em campos frequentemente utilizados em consultas.
- Implementar técnicas de particionamento de tabelas para melhorar o desempenho das consultas.
- Escrever consultas SQL eficientes e otimizadas, evitando subconsultas desnecessárias e garantindo o uso adequado de índices.

4. População do Banco e Desenvolvimento dos KPIs (Indicadores-Chave de Desempenho):

- Cada tabela deve ser populada com 100.000 (cem mil) linhas.
- Popular cada uma das tabelas de acordo com o modelo de dados (colunas e tipo de dado de cada coluna) - 100 mil linhas cada tabela.
- Identificar os KPIs relevantes para monitorar o desempenho do call center, como taxa de resolução, tempo médio de atendimento, taxa de abandono, etc.
- Escrever consultas SQL para calcular e extrair os dados necessários para cada indicador-chave.
- Justificar e explicar o funcionamento de cada consulta e explicar como os KPIs refletem a eficiência da operação do call center.

5. Elaboração do Relatório (Dashboard) Gerencial:

- Nome do Responsável:
- Criação do dashboard gerencial (Power BI ou Excel) e compilação do relatório escrito.
- Compilar todas as informações coletadas ao longo do projeto em um relatório gerencial abrangente.
- Incluir análises sobre as escolhas de campos, otimizações do banco de dados e desenvolvimento dos KPIs.
- Apresentar os resultados de forma clara e objetiva, destacando insights e recomendações para a gestão estratégica do call center.

PROJETO DE OTIMIZAÇÃO DE BANCO DE DADOS DE CALL CENTER

INTRODUÇÃO

Este documento aprofunda a análise dos recursos de otimização utilizados no modelo Star Schema apresentado, detalhando as escolhas de tabelas, tipos de dados e índices de consulta, com o objetivo de aprimorar o desempenho e a eficiência do banco de dados.

ANÁLISE DAS TABELAS

O modelo Star Schema apresenta um design adequado para consultas analíticas complexas, com tabelas dimensionais e de fatos bem estruturadas. As tabelas dimensionais armazenam atributos descritivos dos dados de fato, enquanto a tabela de fatos centraliza os dados quantitativos e as chaves primárias para referenciar as dimensões.

DETALHAMENTO DAS TABELAS

1. TB_CLIENTE

A primeira tabela que temos é a tabela de Clientes. Ela será responsável por armazenar os dados cadastrais dos clientes que serão atendidos pelo nosso callcenter.

- Os dados que armazenaremos serão:

SEQCLIENTE: código sequencial único do tipo inteiro que identificará o cliente. É a chave primária da tabela.

NOME: campo do tipo varchar para guardar o nome completo do cliente.

EMAIL: campo do tipo varchar para armazenar o email do cliente.

TELEFONE: campo do tipo varchar para armazenar o telefone do cliente (limitado a 14 caracteres).

CPF: Campo do tipo varchar para armazenar o CPF do cliente (limitado a 11 caracteres).

DATA_CADASTRO: campo do tipo DATETIME para que seja utilizado para armazenar a data em que o cliente foi cadastrado.

2. TB_DEPARTAMENTO

A segunda tabela é a de Departamento, que fará o registro de todas as áreas que temos no nosso callcenter. Ela é importante pois servirá de base para cadastrarmos nossos funcionários posteriormente e associarmos a um departamento.

- Os dados que armazenaremos nessa tabela serão

SEQDEPARTAMENTO: código sequencial único do tipo inteiro que identificará o departamento. É a chave primária da tabela.

NOME_DEPARTAMENT: campo do tipo varchar para armazenar o nome do departamento.

DATA_CRIAÇÃO: campo do tipo datetime para armazenar a data em que o departamento foi criado.

3. TB_ATENDENTE

A próxima tabela é a de atendentes, que será utilizada para armazenar os dados dos atendentes do callcenter. Ela é criada posteriormente pois é a primeira tabela onde iremos referenciar um registro de outra tabela.

- Os dados que armazenaremos nessa tabela serão

SEQATENDENTE: código sequencial único do tipo inteiro que identificará o atendente. É a chave primária da tabela.

NOME_ATENDENTE: campo do tipo varchar para guardar o nome completo do atendente.

CARGO: campo do tipo varchar para armazenar o cargo do atendente.

SEQDEPARTAMENTO: campo do tipo inteiro, responsável por conectar um atendente a um departamento. É uma chave estrangeira que se conecta ao SEQDEPARTAMENTO da tabela TB_DEPARTAMENTO.

NIVEL_EXPERIENCIA: campo do tipo varchar para armazenar o nível de senioridade do atendente. Exemplo: Assistente, Júnior, Pleno, Sênior.

DATA_ADMISSAO: campo do tipo datetime para armazenar a data em que o atendente foi contratado pela empresa.

DATA_ALTERACAO: campo do tipo datetime para armazenar a data da última alteração que foi realizada no cadastro do atendente. Exemplo: o atendente subiu de Júnior para Pleno; com isso, é preciso registrar nesse campo a data em que foi alterado esse campo.

4. TB_MOTIVO_ATENDIMENTO

Essa tabela é responsável por armazenar os motivos pelo qual um cliente possa vir a contactar o nosso callcenter. Nela, também teremos uma conexão com a tabela de departamento, para fazer com que cada departamento seja responsável por atender determinado tipo de problema.

- Os dados que armazenaremos nessa tabela serão

SEQMOTIVO: código sequencial único do tipo inteiro que identificará o motivo. É a chave primária da tabela.

NOME_MOTIVO: campo do tipo varchar para guardar o motivo.

PRIORIDADE: campo do tipo inteiro para armazenar, para identificar de 1 a 10 qual o nível de prioridade que aquele problema deve ter no atendimento.

DATA_CRIACAO: campo do tipo datetime para armazenar a data em que o motivo foi cadastrado.

DATA_ALTERACAO: campo do tipo datetime para armazenar a data em que o motivo foi alterado pela ultima vez.

DEPARTAMENTO_RESPONSAVEL: campo do tipo inteiro, responsável por conectar um motivo a um departamento. É uma chave estrangeira que se conecta ao SEQDEPARTAMENTO da tabela TB_DEPARTAMENTO.

5. TB_PRODUTO

Essa é a tabela responsável por armazenar os dados dos produtos que nós atendemos no nosso callcenter. Ela também possui uma ligação com a tabela de DEPARTAMENTO, para que departamentos específicos possam ficar responsáveis pelos produtos específicos.

- Os dados que armazenaremos nessa tabela serão:

SEQPRODUTO: código sequencial único do tipo inteiro que identificará o produto. É a chave primária da tabela.

NOME_PRODUTO: campo do tipo varchar para armazenar o nome do produto.

DESCRICAO_PRODUTO: campo do tipo varchar para armazenar uma informação mais detalhada ou complementar do produto.

PRECO: campo do tipo decimal para armazenar o preço do produto.

DEPARTAMENTO_RESPONSAVEL: campo do tipo inteiro para armazenar o código do departamento responsável pelo produto. É uma chave estrangeira que se conecta ao SEQDEPARTAMENTO da tabela TB_DEPARTAMENTO.

DATA_CRIACAO: campo do tipo datetime para armazenar a data em que o produto foi cadastrado.

DATA_ALTERACAO: campo do tipo datetime para armazenar a data em que o produto foi alterado pela ultima vez.

STATUS_PRODUTO: campo do tipo char para armazenar se o produto está ativo ou não. As opções são: "A" para ativo "I" para inativo.

6. TB_CANALATENDIMENTO

Essa tabela é responsável por armazenar os dados dos tipos de canais pelo qual os clientes podem entrar em contato com o nosso callcenter. Exemplo: Whatsapp, Email, Telefone, Telegram, Chatbot.

- Os dados que armazenaremos nessa tabela serão:

SEQCANAL: código sequencial único do tipo inteiro que identificará o canal de atendimento. É a chave primária da tabela.

NOME_CANAL: campo do tipo varchar para armazenar o nome do canal de atendimento.

INFORMACOES_CANAL: campo do tipo varchar para armazenar dados complementares ou detalhadas do canal.

TEMPO_RESPOSTA: campo do tipo time para armazenar o quanto deve ser o tempo máximo de resposta do atendimento para o canal em questão.

DISPONIBILIDADE: campo do tipo varchar para armazenar a disponibilidade que o canal de atendimento ficará de prontidão. Exemplo: 24 horas, horário comercial, das 08 as 18, etc.

STATUS_CANAL: campo do tipo char para armazenar se o canal está ativo ou não. As opções são: "A" para ativo "I" para inativo.

7. TB_ATENDIMENTO

Essa é a principal tabela do nosso sistema de callcenter. Ela é responsável por armazenar os dados de cada atendimento que realizamos. Ela também possui uma referência para as tabelas de cliente, atendente, motivo, produto e canal. Todos esses campos são referenciados pelos seus códigos sequenciais, afim de não armazenarmos dados redundantes na tabela (como nome do cliente, nome do atendente) e garantir que a tabela tenha a melhor performance possível.

- Os dados que armazenaremos nessa tabela serão:

SEQATENDIMENTO: código sequencial único do tipo inteiro que identificará o atendimento realizado. É a chave primária da tabela.

DATA_HORA_INICIO_ATENDIMENTO: campo do tipo datetime para armazenar o dia e horário em que o atendimento iniciou.

DATA_HORA_FIM_ATENDIMENTO: campo do tipo datetime para armazenar o dia e horário em que o atendimento foi finalizado.

SEQCLIENTE: campo do tipo inteiro que serve para referenciar o cliente que foi atendido. É uma chave estrangeira, que se liga a tabela TB_CLIENTE através do campo SEQCLIENTE.

SEQATENDENTE: campo do tipo inteiro que serve para referenciar o atendente que realizou o serviço de atendimento. É uma chave estrangeira que se liga a tabela TB_ATENDENTE através do campo SEQATENDENTE.

SEQMOTIVO: campo do tipo inteiro que serve para referenciar o motivo do atendimento. É uma chave estrangeira que se liga a tabela TB_MOTIVO_ATENDIMENTO através do campo SEQMOTIVO.

PROTOCOLO: campo do tipo varchar responsável por armazenar o código de protocolo de atendimento que o cliente pode utilizar para referenciar o atendimento.

SEQPRODUTO: campo do tipo inteiro que serve para referenciar o produto no qual estamos prestando atendimento. É uma chave estrangeira que se liga a tabela TB_PRODUTO através do campo SEQPRODUTO.

SEQCANAL: campo do tipo inteiro que serve para referenciar por qual canal de atendimento o cliente entrou em contato com o callcenter. É uma chave estrangeira que se liga a tabela TB_CANALATENDIMENTO através do campo SEQCANAL.

ENPS: campo do tipo tinyint que serve para armazenar a nota que foi dada ao atendimento pelo cliente.

DETALHAMENTO DOS ÍNDICES

Os índices foram selecionados e criados todos com base no mesmo princípio: ser utilizado nos principais campos nos quais as tabelas farão conexão umas com as outras e serão utilizadas nas cláusulas where.

1. Tabela TB_ATENDIMENTO:

Na tabela de atendimento, foram escolhidos os campos SEQCLIENTE, SEQATENDENTE, SEQMOTIVO, SEQPRODUTO e SEQCANAL pois todos eles são as chaves estrangeiras da tabela, portanto, ao realizar um JOIN, esses campos serão primordiais para a melhorar a performance nas consultas.

```
CREATE INDEX IDX_ATENDIMENTO_SEQCLIENTE ON  
TB_ATENDIMENTO(SEQCLIENTE);
```

```
CREATE INDEX IDX_ATENDIMENTO_SEQATENDENTE ON  
TB_ATENDIMENTO(SEQATENDENTE);
```

```
CREATE INDEX IDX_ATENDIMENTO_SEQMOTIVO ON  
TB_ATENDIMENTO(SEQMOTIVO);
```

```
CREATE INDEX IDX_ATENDIMENTO_SEQPRODUTO ON  
TB_ATENDIMENTO(SEQPRODUTO);
```

```
CREATE INDEX IDX_ATENDIMENTO_SEQCANAL ON  
TB_ATENDIMENTO(SEQCANAL);
```

2. Tabela TB_CANALATENDIMENTO:

Na tabela de canal de atendimento, foi criado no campo SEQCANAL, devido ao mesmo ser utilizado para se conectar com a tabela de atendimento.

```
CREATE INDEX IDX_CANAL_SEQ ON TB_CANALATENDIMENTO  
(SEQCANAL);
```

3. Tabela TB_CLIENTE

Na tabela de Cliente, foram criados índices tanto no campo de SEQCLIENTE quando no campo de CPF, pois apesar do SEQCLIENTE ser o principal campo responsável por unir as informações do cliente a outras tabelas, o campo CPF será o mais utilizado para identificar um cliente específico em alguma consulta.

```
CREATE INDEX IDX_CLIENTE_CPF ON TB_CLIENTE(CPF);  
CREATE INDEX IDX_CLIENTE_SEQ ON TB_CLIENTE(SEQCLIENTE);
```

4. Tabela TB_ATENDENTE

Na tabela de atendente, foram criados índices tanto na chave primária, quanto na chave estrangeira.

```
CREATE INDEX IDX_ATENDENTE_SEQ ON  
TB_ATENDENTE(SEQATENDENTE);  
CREATE INDEX IDX_ATENDENTE_DPTO ON  
TB_ATENDENTE(SEQDEPARTAMENTO);
```

5. TB_MOTIVO_ATENDIMENTO

Na tabela de motivo atendimento, foram criados índices para a chave primária, juntamente com seu nome e para a chave estrangeira.

```
CREATE INDEX IDX_MOTIVO_SEQ ON  
TB_MOTIVO_ATENDIMENTO(SEQMOTIVO, NOME_MOTIVO);  
CREATE INDEX IDX_MOTIVO_DPTO ON  
TB_MOTIVO_ATENDIMENTO(DEPARTAMENTO_RESPONSAVEL);
```

6. Tabela TB_PRODUTO

Na tabela de produto, foi criado o índice para a chave primária, juntamente com a chave estrangeira de departamento.

```
CREATE INDEX IDX_PRODUTO_SEQ ON TB_PRODUTO(SEQPRODUTO,  
DEPARTAMENTO_RESPONSAVEL);
```

7. Tabela TB_DEPARTAMENTO

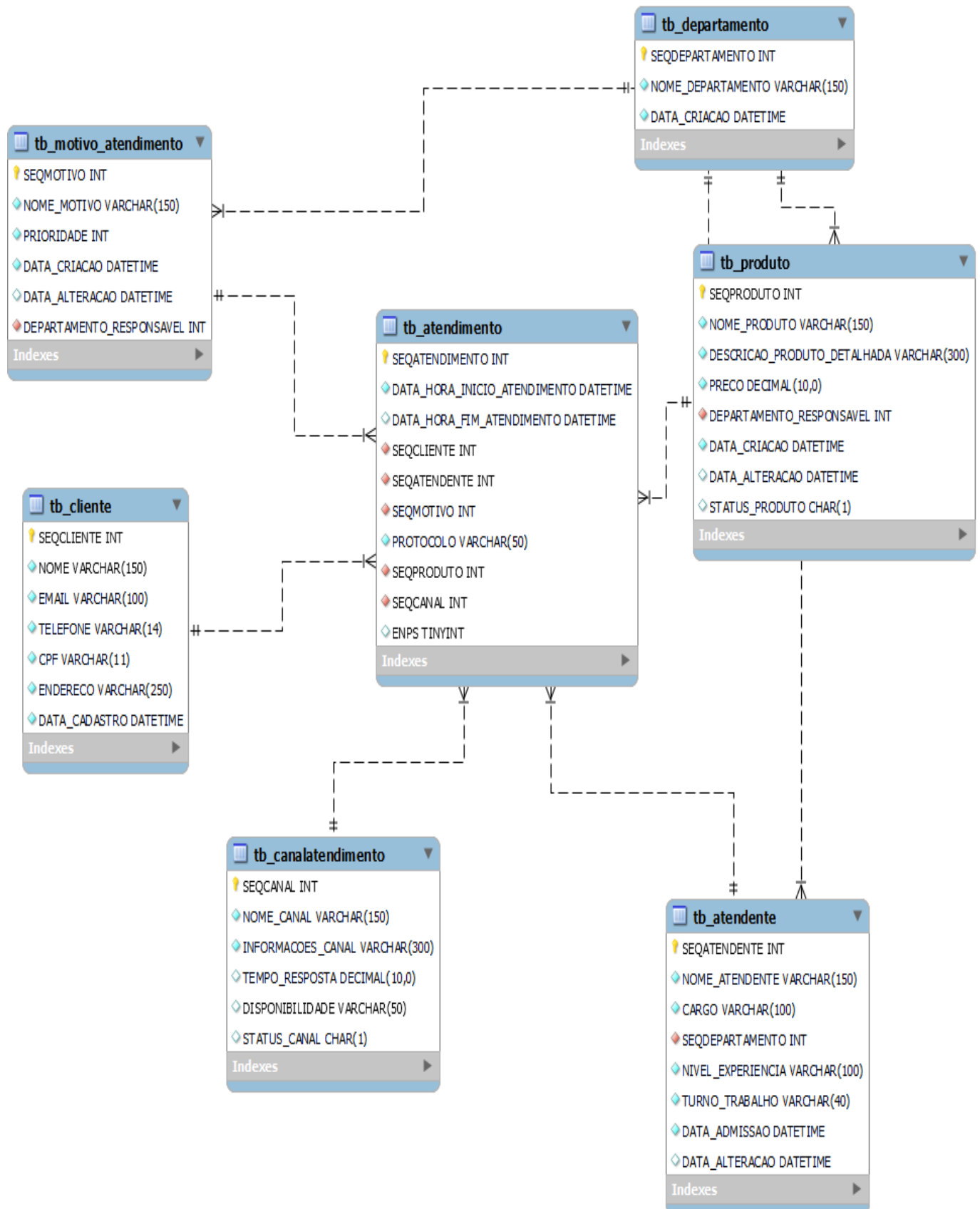
Na tabela de departamento, foi criado o índice tanto para a chave primária quanto para o campo do nome, que provavelmente será o mais utilizado para identificar o departamento numa consulta na tabela.

```
CREATE INDEX IDX_DPTO ON TB_DEPARTAMENTO(SEQDEPARTAMENTO,  
NOME_DEPARTAMENTO);
```

ESTRATÉGIA DETALHADA DE OTIMIZAÇÃO NO MODELO STAR SCHEMA

A estratégia de otimização implementada no modelo Star Schema, abrangendo a modelagem das tabelas, a escolha dos tipos de dados e a utilização de recursos de índices. O objetivo é garantir a eficiência do armazenamento, a precisão das informações e o bom desempenho do banco de dados, atendendo às necessidades de análise de dados complexas e oferecendo uma experiência de consulta rápida e confiável.

DIAGRAMA DE MODELAGEM



MODELAGEM ESCOLHIDA

- **Modelo Star Schema:**

- Adotado para otimizar consultas analíticas complexas, com tabelas dimensionais e de fatos bem estruturadas.
- Tabelas dimensionais armazenam atributos descritivos dos dados de fato, enquanto a tabela de fatos centraliza os dados quantitativos e as chaves primárias para referenciar as dimensões.
- Essa estrutura facilita a agregação e análise de dados de diferentes dimensões, otimizando o desempenho das consultas.

- **Normalização Estratégica:**

- Nível de normalização 3NF (Terceira Forma Normal) implementado para minimizar redundância de dados e garantir integridade referencial.
- Evita anomalias de atualização e inserção, além de otimizar o espaço de armazenamento.
- As tabelas dimensionais armazenam apenas atributos relevantes para as análises, enquanto a tabela de fatos contém os dados quantitativos e as chaves primárias para referenciar as dimensões.

- **Chaves Estrangeiras:**

- Utilizadas para conectar as tabelas dimensionais à tabela de fatos, garantindo a integridade referencial.
- Permitem a navegabilidade eficiente entre as tabelas e facilitam a recuperação de dados relacionados.
- As chaves estrangeiras são implementadas com índices para otimizar as junções entre as tabelas.

TIPOS DE DADOS

- **Seleção Estratégica:**

- A escolha dos tipos de dados foi feita com base na natureza dos dados, nas necessidades de armazenamento e nas operações que serão realizadas sobre eles.
- Priorizada a eficiência do armazenamento, a precisão das informações e o desempenho das consultas.

- **Tipos de Dados Utilizados:**

- **INT:**

- Utilizado para campos que representam valores numéricos inteiros e únicos, como identificadores de entidades (SEQCLIENTE, SEQDEPARTAMENTO, etc.).
 - Permite ordenação rápida, indexação eficiente e consultas otimizadas.

- **VARCHAR:**

- Utilizado para campos que armazenam strings com tamanhos variáveis, como nomes, descrições e informações textuais (NOME, NOME_DEPARTAMENTO, etc.).
 - Oferece flexibilidade para diferentes tamanhos de dados e otimiza o espaço de armazenamento.

- **DATETIME:**

- Utilizado para campos que armazenam datas e horas com precisão, como datas de cadastro, datas de atendimento e datas de criação (DATA_CADASTRO, DATA_HORA_INICIO_ATENDIMENTO, etc.).
 - Permite ordenação cronológica, filtragem por data e análise de tendências.

- **DECIMAL:**

- Utilizado para campos que representam valores monetários, como preços de produtos (PRECO).
 - Garante precisão nos cálculos financeiros e na análise de dados monetários.

- **TINYINT:**

- Utilizado para campos que armazenam valores numéricos pequenos com pouca variação, como notas de avaliação (ENPS).
 - Otimiza o espaço de armazenamento e permite consultas eficientes.
 -

- **Justificativa da Escolha:**

- A escolha dos tipos de dados foi feita com o objetivo de garantir:

- Eficiência no armazenamento: Os tipos de dados escolhidos otimizam o espaço de armazenamento sem comprometer a precisão das informações.
- Precisão das informações: Os tipos de dados garantem que os dados sejam armazenados e processados de forma precisa, atendendo às necessidades de análise.
- Desempenho das consultas: Os tipos de dados e os índices otimizam as consultas, permitindo que sejam realizadas de forma rápida e eficiente.

ÍNDICES

- **Criados em Campos Chave:**

- Índices criados nas chaves primárias de todas as tabelas para otimizar a recuperação de registros específicos.
- Permitem acesso rápido aos dados e garantem o bom desempenho das consultas que utilizam as chaves primárias como critério de busca.

- **Criados em Campos de Consulta Freqüente:**

- Índices criados em campos que são frequentemente utilizados como critérios de filtro ou ordenação nas consultas.

Otimizam a performance das consultas que utilizam esses campos, permitindo a filtragem e ordenação eficientes dos dados.

- **Justificativa da Utilização:**

- A utilização de índices melhora o desempenho das consultas ao fornecer um ponteiro rápido para os dados relevantes.
- Ao invés de ler toda a tabela, o banco de dados pode utilizar o índice para localizar rapidamente os registros que correspondem ao critério de busca.

A estratégia de otimização implementada no modelo Star Schema, aliada à modelagem de tabelas, à seleção de tipos de dados e à utilização de índices, garante a

eficiência do armazenamento, a precisão das informações e o bom desempenho geral do modelo. Essa abordagem permite realizar consultas analíticas complexas de forma rápida e confiável, atendendo às necessidades de análise de dados e oferecendo uma experiência satisfatória para os usuários do sistema.

DASHBOARD PARA ANÁLISE DETALHADA DO ATENDIMENTO AO CLIENTE

Foi elaborado um dashboard no Power BI para avaliar indicadores de desempenho crucial no atendimento ao cliente. A estratégia de otimização utilizada no modelo Star Schema garante a eficiência do armazenamento, a precisão das informações e o bom desempenho geral do modelo, permitindo a construção de visualizações poderosas e interativas.

VISUALIZAÇÕES ESTRATÉGICAS

O dashboard apresenta uma série de gráficos selecionados para fornecer uma visão abrangente do desempenho do atendimento ao cliente:

1. Tempo Médio de Atendimento por Nome do Produto:

- **Tipo de Gráfico:** Coluna ou Barra
- **Análise:** Identifica os produtos que exigem maior tempo de atendimento, permitindo ações direcionadas para otimizar processos e reduzir o tempo de resolução.

2. Tempo Médio de Atendimento por Nome do Canal:

- **Tipo de Gráfico:** Coluna ou Barra
- **Análise:** Avalia a eficiência de cada canal de atendimento, possibilitando identificar gargalos e direcionar investimentos para os canais que apresentam maior demanda.

3. Tempo Médio de Atendimento por Nome do Departamento:

- **Tipo de Gráfico:** Coluna ou Barra

- **Análise:** Mede a performance de cada departamento em relação à resolução de atendimentos, auxiliando na identificação de áreas que necessitam de treinamento ou otimização de processos.

4. Tempo Médio de Atendimento por Nome do Atendente:

- **Tipo de Gráfico:** Coluna ou Barra
- **Análise:** Monitora o desempenho individual dos atendentes, permitindo identificar talentos, oferecer treinamento direcionado e otimizar a alocação de recursos.

5. TMA por Motivo de Atendimento:

- **Tipo de Gráfico:** Coluna
- **Análise:** Compreende o tempo médio de atendimento para cada motivo, possibilitando identificar padrões, investigar causas raízes e implementar soluções para os motivos que geram maior tempo de resolução.

6. Total de Atendimentos e Mediana da Taxa de Atendimento por Mês:

- **Tipo de Gráfico:** Linha e coluna
- **Análise:** Visualiza o volume de atendimentos e a taxa de resolução ao longo do tempo, permitindo identificar tendências, sazonalidades e áreas que precisam de atenção.

7. Soma do Tempo de Resolução por Nome do Canal:

- **Tipo de Gráfico:** Coluna ou Barra
- **Análise:** Quantifica o tempo total de atendimento por canal, fornecendo insights sobre a carga de trabalho e a necessidade de otimização de recursos.

IMPORTAÇÃO E RELACIONAMENTO DE TABELAS:

A modelagem Star Schema otimizada foi importada para o Power BI através do método de importação do MySQL, garantindo a integridade e a confiabilidade dos dados. As tabelas foram relacionadas de acordo com a lógica dimensional, permitindo a criação de relacionamentos eficientes e a construção de visualizações precisas.

KPIS ESTRATÉGICOS:

Para complementar a análise, foram inseridos KPIs estratégicos de desenvolvimento, detalhados a seguir:

QUERY 1 - Quantidade_Atendimentos_Mês_Atendente:

- **Objetivo:** Quantificar o número de atendimentos realizados por cada atendente em um determinado mês.
- **Utilização:** Monitorar a produtividade individual, identificar gargalos e direcionar treinamentos.

KPI DE NOME ATENDENTE, MÊS, ANO E NÚMERO DE ATENDIMENTOS POR MÊS.

SELECT

AT.NOME_ATENDENTE,
MONTHNAME(A.DATA_HORA_INICIO_ATENDIMENTO) AS MES,
YEAR(A.DATA_HORA_INICIO_ATENDIMENTO) AS ANO,
COUNT(A.SEQATENDIMENTO) AS NUMERO_ATENDIMENTOS

FROM

TB_ATENDIMENTO A
INNER JOIN TB_ATENDENTE AT ON A.SEQATENDENTE =
AT.SEQATENDENTE

GROUP BY

AT.NOME_ATENDENTE,
MONTHNAME(A.DATA_HORA_INICIO_ATENDIMENTO),
YEAR(A.DATA_HORA_INICIO_ATENDIMENTO);

QUERY 2 - tempo_médio_resolução_motivo:

- **Objetivo:** Calcular o tempo médio de resolução para cada motivo de atendimento.
- **Utilização:** Identificar os motivos que geram maior tempo de resolução e implementar ações para otimizar os processos.

```

SELECT
    C.NOME_CANAL,
    M.NOME_MOTIVO,
    SEC_TO_TIME(ROUND(AVG(TIMESTAMPDIFF(SECOND,
    A.DATA_HORA_INICIO_ATENDIMENTO,
    A.DATA_HORA_FIM_ATENDIMENTO)) / 3600)) AS
    TEMPO_MEDIO_RESOLUCAO_HORAS
FROM
    TB_ATENDIMENTO A
    INNER JOIN TB_CANALATENDIMENTO C ON A.SEQCANAL =
    C.SEQCANAL
    INNER JOIN TB_MOTIVO_ATENDIMENTO M ON A.SEQMOTIVO =
    M.SEQMOTIVO
WHERE
    A.DATA_HORA_FIM_ATENDIMENTO IS NOT NULL
GROUP BY
    C.NOME_CANAL,
    M.NOME_MOTIVO
ORDER BY
    TEMPO_MEDIO_RESOLUCAO_HORAS DESC;

```

QUERY 3 - Taxa_Resolução_Atendimento:

- **Objetivo:** Determinar a taxa de resolução de cada atendimento, considerando se o problema do cliente foi solucionado na primeira interação.
- **Utilização:** Avaliar a eficiência do atendimento à primeira vista, identificar áreas de melhoria e direcionar treinamentos para os atendentes.

```

SELECT
    D.NOME_DEPARTAMENTO,
    MONTH(A.DATA_HORA_INICIO_ATENDIMENTO) AS MES,
    YEAR(A.DATA_HORA_INICIO_ATENDIMENTO) AS ANO,
    COUNT(A.SEQATENDIMENTO) AS TOTAL_ATENDIMENTOS,

```

```

COUNT(CASE WHEN A.DATA_HORA_FIM_ATENDIMENTO IS NOT
NULL THEN 1 END) AS ATENDIMENTOS_RESOLVIDOS,
ROUND(((COUNT(CASE WHEN A.DATA_HORA_FIM_ATENDIMENTO IS
NOT NULL THEN 1 END) / COUNT(A.SEQATENDIMENTO)) * 100, 2) AS
TAXA_RESOLUCAO
FROM
TB_ATENDIMENTO A
INNER JOIN TB_ATENDENTE AT ON A.SEQATENDENTE =
AT.SEQATENDENTE
INNER JOIN TB_DEPARTAMENTO D ON AT.SEQDEPARTAMENTO =
D.SEQDEPARTAMENTO
GROUP BY
D.NOME_DEPARTAMENTO,
MONTH(A.DATA_HORA_INICIO_ATENDIMENTO),
YEAR(A.DATA_HORA_INICIO_ATENDIMENTO);

```

QUERY 4 - Tempo_Medio_Resolução_2023:

- **Objetivo:** Calcular o tempo médio de resolução de atendimentos no ano de 2023.
- **Utilização:** Analisar a performance geral do atendimento ao cliente em 2023 e identificar tendências ao longo do ano.

```

SELECT
M.SEQMOTIVO,
M.NOME_MOTIVO,
SEC_TO_TIME(ROUND(AVG(TIME_TO_SEC(TEMPO_MEDIO_RESOLU
CAO_HORAS)))) AS MEDIA_TEMPO_MEDIO_RESOLUCAO_HORAS
FROM ( SELECT A.SEQMOTIVO,
SEC_TO_TIME(ROUND(AVG(TIMESTAMPDIFF(SECOND,
A.DATA_HORA_INICIO_ATENDIMENTO,
A.DATA_HORA_FIM_ATENDIMENTO)) / 3600)) AS
TEMPO_MEDIO_RESOLUCAO_HORAS
FROM

```

```

TB_ATENDIMENTO A
WHERE
  A.DATA_HORA_FIM_ATENDIMENTO IS NOT NULL
  AND YEAR(A.DATA_HORA_INICIO_ATENDIMENTO) = 2023
GROUP BY
  A.SEQMOTIVO) AS subquery INNER JOIN TB_MOTIVO_ATENDIMENTO
M ON subquery.SEQMOTIVO = M.SEQMOTIVO GROUP BY
M.SEQMOTIVO, M.NOME_MOTIVO;

```

QUERY 5 - Tempo_Medio_Resolução_2024:

- **Objetivo:** Calcular o tempo médio de resolução de atendimentos no ano de 2024 (até o momento da consulta).
- **Utilização:** Monitorar a performance do atendimento ao cliente em 20

```

SELECT
  M.SEQMOTIVO,
  M.NOME_MOTIVO,

  SEC_TO_TIME(ROUND(AVG(TIME_TO_SEC(TEMPO_MEDIO_RESOLUCAO_H
ORAS)))) AS MEDIA_TEMPO_MEDIO_RESOLUCAO_HORAS
FROM (
  SELECT
  A.SEQMOTIVO,
    SEC_TO_TIME(ROUND(AVG(TIMESTAMPDIFF(SECOND,
A.DATA_HORA_INICIO_ATENDIMENTO,
A.DATA_HORA_FIM_ATENDIMENTO)) / 3600)) AS
TEMPO_MEDIO_RESOLUCAO_HORAS
  FROM
    TB_ATENDIMENTO A
  WHERE
    A.DATA_HORA_FIM_ATENDIMENTO IS NOT NULL

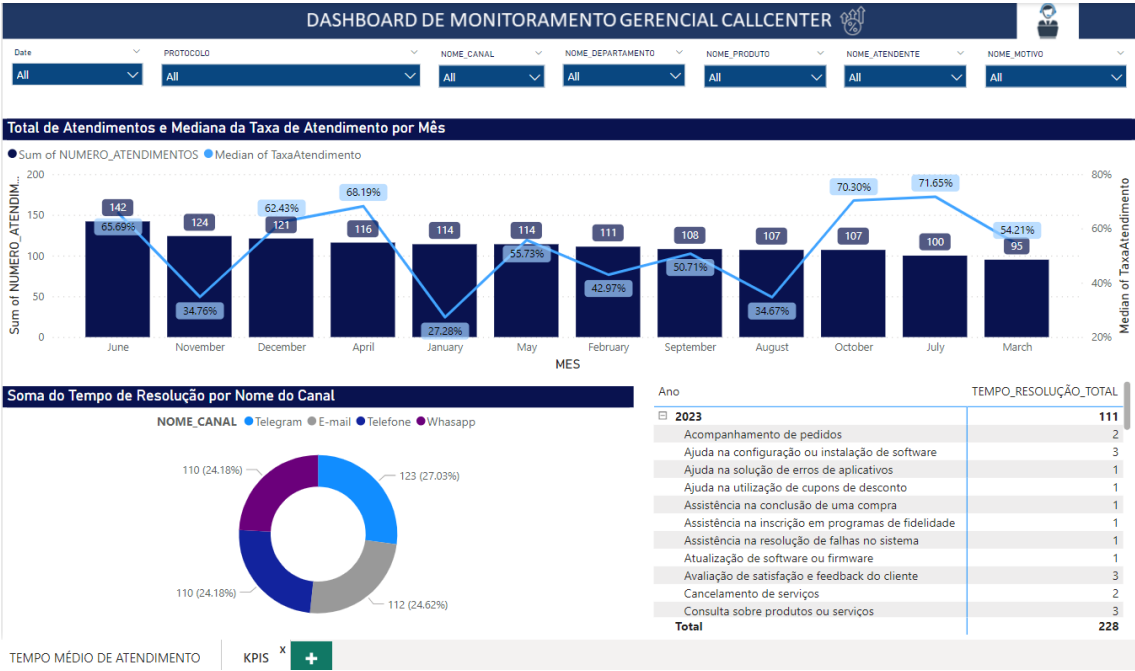
```



```
        AND YEAR(A.DATA_HORA_INICIO_ATENDIMENTO) = 2024
GROUP BY
A.SEQMOTIVO
) AS subquery
INNER JOIN TB_MOTIVO_ATENDIMENTO M ON subquery.SEQMOTIVO =
M.SEQMOTIVO
GROUP BY
M.SEQMOTIVO, M.NOME_MOTIVO;
```

IMAGENS DO DASHBOARD





REPOSITÓRIO GITHUB COM INFORMAÇÕES ADICIONAIS

Para facilitar o acesso à documentação completa do projeto, foi criado um repositório público no GitHub, disponível no seguinte link:

<https://github.com/jacksonwsup/bancodedados>

Conteúdo do Repositório:

O repositório contém os seguintes arquivos importantes para a avaliação do projeto:

Diagrama:

- DiagramaV2.png: Diagrama UML representando o modelo de dados Star Schema.

Dump de Dados:

- Dump20240504-1.rar: Arquivo compactado contendo o dump completo do banco de dados em 04 de maio de 2024.

Power BI:

- PROJETO DE OTIMIZAÇÃO DASHBOARD DE INDICADORES.pbix: Arquivo do projeto Power BI contendo o dashboard de indicadores de desempenho.

Scripts para Geração de Relatórios:

- PROJETO DE OTIMIZAÇÃO.sql: Script SQL para gerar os relatórios Kpis.

README.md: Arquivo de texto contendo informações sobre o projeto e instruções de uso.

Relatórios SQL:

- relatorio_kpi_Atendimentos.sql: Script SQL para gerar o relatório KPI Atendimentos.

- relatorio_kpi_canalatendimento_motivos.sql: Script SQL para gerar o relatório KPI Canal Atendimento x Motivos.
- relatorio_kpi_media_tempo_atendimentos_por_motivos.sql: Script SQL para gerar o relatório KPI Tempo Médio Atendimento por Motivo.