

# Relatório de Estágio

Juliana Carneiro da Motta

Curso Técnico Superior Profissional em  
Análise de Dados

set | 2023

GUARDA  
POLI  
TÉCNICO



**Escola Superior de Tecnologia e Gestão**

---

***BUSINESS INTELLIGENCE: PROJETO DE CRIAÇÃO DE  
DASHBOARDS PARA A EMPRESA MESTRECLIQUE***

---

RELATÓRIO DE ESTÁGIO PARA OBTENÇÃO DO DIPLOMA DE  
TÉCNICO(A) SUPERIOR PROFISSIONAL EM ANÁLISE DE DADOS

**Juliana Carneiro da Motta**

**Setembro / 2023**

# POLI TÉCNICO GUARDA

Escola Superior de Tecnologia e Gestão

---

***BUSINESS INTELLIGENCE: PROJETO DE CRIAÇÃO DE  
DASHBOARDS PARA A EMPRESA MESTRECLIQUE***

---

RELATÓRIO DE ESTÁGIO PARA OBTENÇÃO DO DIPLOMA DE TÉCNICO(A)  
SUPERIOR PROFISSIONAL EM ANÁLISE DE DADOS

Professor(a) Orientador(a): Paulo Alexandre Andrade Vieira

Juliana Carneiro da Motta

Setembro / 2023

# POLI TÉCNICO GUARDA

## Ficha Técnica

### **Estagiária**

Nome: Juliana Carneiro da Motta

Matrícula nº 1706988

*Linkedin:* [linkedin.com/in/jumotta](https://www.linkedin.com/in/jumotta)

Curso: Técnico Superior Profissional (CTeSP) em Análise de Dados

### **Estabelecimento de Ensino**

Instituto Politécnico da Guarda

Escola Superior de Tecnologia e Gestão

Av. Dr. Francisco Sá Carneiro, nº50, 6300-559, Guarda, Portugal

Contato Telefónico: 271220100

E-mail: [estg-geral@ipg.pt](mailto:estg-geral@ipg.pt)

### **Instituição de Acolhimento**

Empresa Mestrequique

Av. Monsenhor Mendes do Carmo, Nº12 A, 6300-586 - Guarda

Contato Telefónico: +351 271 227 354

E-mail: [geral@mestrequique.pt](mailto:geral@mestrequique.pt)

*Website:* <https://mestrequique.pt>

# POLI TÉCNICO GUARDA

## **Duração do Estágio**

Início: 13/02/2023

Término: 18/07/2023

## **Orientador na Escola Superior de Tecnologia e Gestão (ESTG)**

Nome: Paulo Alexandre Andrade Vieira

Grau Acadêmico: Doutor em Informática e Automação

## **Orientador na Instituição de Acolhimento**

Nome: Leonardo Lourenço

Grau Acadêmico: Licenciatura em Engenharia Informática

# POLI TÉCNICO GUARDA

## Resumo

O presente relatório surgiu no âmbito da unidade curricular denominada Estágio e está incluído no plano curricular do curso Técnico Profissional de Análise de Dados do Instituto Politécnico da Guarda, sendo realizado na empresa Mestreclique.

Este estágio teve início no dia 13 de fevereiro de 2023 e término no dia 18 de julho de 2023, com uma duração de 753 horas. O estágio teve como objetivo o estudo da plataforma de análise de dados pertencente à Microsoft, denominada *Power BI*, e a construção de dois *dashboards*, um direcionado à análise interna da Mestreclique e outro voltado aos seus clientes, capazes de fornecerem informações úteis à própria empresa e a seus clientes.

O relatório inicia-se com uma revisão da literatura sobre *Business Intelligence (BI)* e *Power BI*, e destaca a importância dessas ferramentas na transformação de dados brutos em insights acionáveis. É detalhado o funcionamento do *Power BI*, com ênfase no *Power Query* para a preparação de dados, e da Linguagem DAX para cálculos avançados e criação de medidas personalizadas. O relatório explora o processo desde a coleta e preparação dos dados até a publicação dos *dashboards*.

Os *dashboards* propostos satisfazem as necessidades sentidas, nestes são apresentados os indicadores definidos pela Mestreclique enquadrados com os objetivos propostos. Estes *dashboards* permitem que os utilizadores tenham acesso a um conjunto de mecanismos interativos e dinâmicos por meio de gráficos ilustrativos, o que faz com que tenham um conhecimento imediato e claro da atual situação e que façam uma análise das informações com um maior nível de detalhe. Também foi possível monitorar e analisar o desempenho, identificar eventuais falhas e tomar decisões mais seguras em conformidade com os objetivos da empresa. No final foi realizada a apresentação aos clientes da Mestreclique.

Palavras-passe: *Business Intelligence, Power BI, dados, linguagem DAX, dashboard*

# POLI TÉCNICO GUARDA

## Abstract

The present report emerged within the scope of the course unit called Internship and is included in the curriculum of the Professional Technical Course in Data Analysis at the Polytechnic Institute of Guarda, carried out at the Mestrelrique company.

The internship began on February 13, 2023, and concluded on July 18, 2023, lasting for 753 hours. The objective of the internship was to study the data analysis platform belonging to Microsoft, called Power BI, and to build two dashboards — one directed towards Mestrelrique's internal analysis and another geared towards its clients. These dashboards aimed to provide useful information to the company itself and its clients.

The report starts with a literature review on Business Intelligence (BI) and Power BI, emphasizing the importance of these tools in transforming raw data into actionable insights. The functioning of Power BI is detailed, with a focus on Power Query for data preparation and the DAX language for advanced calculations and custom measure creation. The report explores the process from data collection and preparation to dashboard publication.

The proposed dashboards meet the identified needs, presenting indicators defined by Mestrelrique aligned with the set objectives. These dashboards enable users to access a set of interactive and dynamic mechanisms through illustrative graphs, providing immediate and clear knowledge of the current situation and allowing for a detailed analysis of information. It was also possible to monitor and analyze performance, identify potential issues, and make more informed decisions in line with the company's objectives. In the end, a presentation of the dashboards was made to Mestrelrique's clients.

Keywords: Business Intelligence, Power BI, data, DAX language, dashboard

# POLI TÉCNICO GUARDA

## Índice

1	Introdução .....	1
1.1	Enquadramento e Motivação.....	1
1.2	Caracterização da Instituição de Acolhimento.....	1
1.3	Objetivos .....	2
1.4	Estrutura do Relatório .....	3
2	Estado da Arte .....	4
2.1	<i>Power BI</i> .....	<b>Erro! Marcador não definido.</b>
2.2	Outros <i>Softwares</i> de <i>Business Intelligence</i> .....	7
3	Revisão da Literatura.....	8
3.1	<i>Business Intelligence</i> .....	<b>Erro! Marcador não definido.</b>
3.2	<i>Power BI</i> .....	<b>Erro! Marcador não definido.</b>
3.2.1	Importação e Preparação de Dados.....	10
3.2.2	Criação de Visualizações .....	14
3.2.3	Interação e Publicação .....	16
4	Tecnologias e Ferramentas Utilizadas.....	17
5	O trabalho realizado: O Modelo de Dados.....	20
5.1	RECOLHA E SELEÇÃO DOS DADOS.....	22
5.2	TRANSFORMAÇÃO DOS DADOS .....	23
5.2.1	CRIAÇÃO DA TABELA CALENDÁRIO .....	25
5.2.2	DADOS DA MESTRECLIQUE .....	31
5.2.3	DADOS DOS CLIENTES DA MESTRECLIQUE.....	35
5.3	CARREGAMENTO DOS DADOS .....	37
5.4	RELACIONAMENTOS.....	38
6	O trabalho realizado: Implementação dos <i>Dashboards</i> .....	40
6.1	USO DA LINGUAGEM DAX .....	40
6.1.1	PROJETO DA MESTRECLIQUE .....	40
6.1.2	PROJETO DOS CLIENTES .....	44
7	O trabalho realizado: Os <i>Dashboards</i> .....	52
7.1	<i>DASHBOARD</i> DA MESTRECLIQUE .....	52
7.2	<i>DASHBOARD</i> DO CLIENTE .....	63
8	<i>Dashboards</i> em Serviço .....	69



# POLI TÉCNICO GUARDA

8.1	<i>GATEWAY</i> .....	<b>Erro! Marcador não definido.</b>
8.1.1	INSTALAÇÃO E CONFIGURAÇÃO DO <i>GATEWAY</i> .....	70
8.1.2	CONFIGURAÇÃO DA FONTE DE DADOS.....	70
8.2	ATUALIZAÇÃO DO <i>DASHBOARD</i> .....	71
8.3	PUBLICAÇÃO NA WEB .....	72
9	CONCLUSÃO .....	75
10	Bibliografia .....	76
11	ANEXOS .....	77
11.1	ANEXO A - Plano de estágio .....	78
11.2	ANEXO B – Transformação dos dados nas demais tabelas .....	79

# POLI TÉCNICO GUARDA

## Índice de Figuras

Figura 1 - Quadrante Mágico de Gartner.....	4
Figura 2 - Principais Componentes do <i>Power BI</i> .....	5
Figura 3 - Versões do <i>Power BI</i> .....	6
Figura 4 - Fluxo do processo de funcionamento do <i>Power BI</i> .....	10
Figura 5 - Processo ETL.....	11
Figura 6 - Interface do <i>Power Query</i> .....	13
Figura 7 - Ícones de Exibição no <i>Power BI</i> Desktop .....	14
Figura 8 - Modo Exibição Relatório.....	15
Figura 9 - Forma de acessar os códigos da Linguagem M .....	17
Figura 10 - Funcionalidade “Obter Dados” .....	22
Figura 11 – Preenchimento dos campos para ligar ao <i>SQL Server</i> .....	23
Figura 12 - Algumas Funcionalidades do <i>Power Query</i> .....	24
Figura 13 - Escolha de uma consulta em branco para criação da tabela calendário.....	25
Figura 14 - Editor avançado da consulta em branco .....	26
Figura 15 - Código para criação da tabela calendário .....	27
Figura 16 - Transformação da lista em tabela .....	28
Figura 17 - Alteração do nome e tipo da coluna.....	29
Figura 18 - Forma de criar as colunas auxiliares de datas.....	30
Figura 19 - Transformações realizadas, em linguagem M, na tabela <i>TICKETS</i> .....	31
Figura 20 - Função Adicionar Coluna Condicional.....	31
Figura 21 - Campos preenchidos na função Adicionar Coluna Condicional .....	32
Figura 22 - Opção Intercalar Consultas no <i>Power Query</i> .....	32
Figura 23 - Função Intercalar Consultas.....	33
Figura 24 - Escolha da coluna a utilizar .....	33
Figura 25 - Adicionada a coluna Tempo Resposta.....	34
Figura 26 - Transformações realizadas, em linguagem M, na tabela <i>INSCRICAO</i> .....	35
Figura 27 - Ícones das colunas .....	35
Figura 28 - Substituição de valores .....	36
Figura 29 - Botão Fechar e Aplicar .....	37
Figura 30 - Modelo Lógico   Projeto Mestreclique .....	38
Figura 31 - Modelo Lógico   Projeto cliente Mestreclique.....	39
Figura 32 - Utilização da fórmula <i>COUNT</i> .....	40
Figura 33 - Cálculo da medida "Total Tempo Resposta" .....	41
Figura 34 - Cálculo da medida "Total Tempo Resposta Hora" .....	41
Figura 35 - Cálculo da medida "Media Tempo Resposta" .....	41
Figura 36 - Sintaxe da função <i>CALCULATE</i> .....	42
Figura 37 - Fórmula da medida "TotalClientesAtivos" .....	42
Figura 38 - Cálculo da medida " <i>Tickets</i> Respondidos - Estados" .....	43
Figura 39 - Fórmula para criar o Percentual de <i>Tickets</i> Respondidos .....	43

# POLI TÉCNICO GUARDA

Figura 40 - Cálculo da quantidade de <i>tickets</i> abertos .....	44
Figura 41 - Criação de nova coluna calculada.....	44
Figura 42 - Barra de fórmulas para criação da coluna calculada.....	45
Figura 43 - Fórmula DAX para coluna calculada.....	45
Figura 44 - Classificações da faixa etária .....	46
Figura 45 - Erro na coluna da faixa etária .....	47
Figura 46 - Correção de erro na coluna .....	47
Figura 47 - Janela para introduzir o valor na correção do erro.....	48
Figura 48 - Tabela com as medidas - projeto dos clientes.....	48
Figura 49 - Uso da função AVERAGE com ROUND .....	49
Figura 50 - Cálculo do número de inscrições do ano anterior.....	49
Figura 51 - Cálculo do <i>ranking</i> .....	50
Figura 52 - Visualização do <i>ranking</i> em uma matriz .....	51
Figura 53 - Página 1 do <i>dashboard</i> da Mestreclique.....	52
Figura 54 - Início da construção do relatório .....	53
Figura 55 - Configuração do cartão .....	53
Figura 56 - Construção da dica de ferramenta.....	54
Figura 57 - Dica de ferramenta dos <i>tickets</i> em andamento.....	55
Figura 58 - Dica de ferramenta dos <i>tickets</i> respondidos.....	56
Figura 59 - Tempo médio de resposta ao <i>ticket</i> .....	56
Figura 60 - Total de <i>tickets</i> em cada mês .....	57
Figura 61 - Tempo médio de resposta aos <i>tickets</i> no decorrer do ano .....	57
Figura 62 - Página 2 do <i>dashboard</i> da Mestreclique.....	58
Figura 63 - Treemap dos clientes com mais <i>tickets</i> .....	58
Figura 64 - Grau de Satisfação dos Clientes.....	59
Figura 65 - Clientes com <i>tickets</i> em andamento.....	59
Figura 66 - Tempo de resposta a cada cliente .....	60
Figura 67 - Comparação entre <i>tickets</i> abertos e respondidos .....	61
Figura 68 - Previsão e anomalia da quantidade de <i>tickets</i> .....	61
Figura 69 - Ícone de análises .....	62
Figura 70 - Configurações da previsão.....	62
Figura 71 - Configuração da anomalia .....	63
Figura 72 - Página 1 do <i>dashboard</i> dos clientes.....	64
Figura 73 - Dica de ferramenta das inscrições de cada curso.....	65
Figura 74 - Gráfico de colunas agrupadas com total de inscrições .....	65
Figura 75 - Página 2 do <i>dashboard</i> dos clientes.....	66
Figura 76 - Gráfico de pizza .....	67
Figura 77 - Página 3 do <i>dashboard</i> dos clientes.....	68
Figura 78 - Publicação através do <i>Power BI</i> Desktop.....	69
Figura 79 - Gestão de <i>gateways</i> .....	70
Figura 80 - Botão para atualização manual .....	71
Figura 81 - Atualização automática.....	71

# POLI TÉCNICO GUARDA

Figura 82 - Transferência dos arquivos no FileZilla .....	72
Figura 83 - Publicar na web pelo <i>Power BI Service</i> .....	73
Figura 84 - Código gerado pelo <i>Power BI Service</i> .....	74

# POLI TÉCNICO GUARDA

## Acrónimos

BD: Base de Dados

BI: *Business Intelligence*

CTeSP: Curso Técnico Superior Profissional

DAX: *Data Analysis Expression*

ETL: *Extract, Transform, Load*

ESTG: Escola Superior de Tecnologia e Gestão

FTP: *File Transfer Protocol*

HTML: *HyperText Markup Language*

KPI: *Key Performance Indicator*

SaaS: *Software as a Service*

SFTP: *Secure File Transfer Protocol*

SGBD: Sistema de Gestão de Banco de dados

SQL: *Structured Query Language*

TI: Tecnologia da Informação

# POLI TÉCNICO GUARDA

## 1 Introdução

Este capítulo tem por objetivo realizar uma descrição introdutória do presente relatório. A estrutura e os objetivos do documento, bem como a breve descrição das tarefas elaboradas no decorrer do estágio, serão apresentadas.

### 1.1 Enquadramento e Motivação

O presente relatório descreve as atividades desenvolvidas e os resultados obtidos, na empresa Mestreclique, durante o período de realização do estágio curricular do Curso Técnico Superior Profissional (CTeSP) de Análise de Dados do Instituto Politécnico da Guarda, durante o período de 13 de fevereiro de 2023 a 18 de julho de 2023, totalizando 750 horas.

A realização deste estágio teve um papel crucial no desenvolvimento profissional, uma vez que foi possível adquirir uma maior experiência e aprimorar muitas competências, bem como foi possível colocar em prática diversos conhecimentos adquiridos no decorrer do curso.

### 1.2 Caracterização da Instituição de Acolhimento

A empresa Mestreclique – Sistemas de Informação, Lda, instituição acolhedora do estágio, foi constituída no ano de 2002 e, desde o primeiro momento, atuou na criação de *software* para gerir formação. Inicialmente possuía dois sócios, Rui Pereira e Ricardo Dias, e, com o conhecimento de ambos, surgiu a Plataforma Mestreclique.

Com o decorrer do tempo, a Mestreclique expandiu suas atividades para o mercado de TI e abriu um espaço comercial ao público, que hoje se localiza na Av. Monsenhor Mendes do Carmo, na Guarda, para a venda de *hardware*, *software* genérico, gestão comercial e demais serviços. O horário de funcionamento vai das 09h às 13h e das 14:30h às 18:30h, que era o mesmo horário cumprido no estágio.

Atualmente, a empresa consta de apenas um sócio, Rui Pereira, e gere formação com a Plataforma Mestreclique a dezenas de clientes em todo o território nacional e Angola.

A empresa possui os seguintes departamentos: Programação e *Web Design*, Comercial e Serviços *HelpDesk* e Pós-venda.

# POLI TÉCNICO GUARDA

A Mestreclique apresenta como missão fornecer ao mercado soluções de gestão baseadas em sistemas e TI. Ela apoia a capacitação dos seus clientes, com ferramentas e serviços inovadores, e estimula o potencial humano e tecnológico em um clima de parceria.

## 1.3 Objetivos

O principal objetivo deste estágio foi a criação de dois *dashboards*, um para a empresa Mestreclique e outro para os seus clientes, utilizando o *Power BI*, uma ferramenta de *Business Intelligence* (BI). A utilização do *Power BI* foi essencial para transformar grandes quantidades de dados em valiosas informações capazes de demonstrarem a atual situação organizacional. Como a empresa ainda não utilizava este tipo de análise, foi possível identificar algumas lacunas e tomar decisões mais objetivas e rápidas.

Inicialmente foi elaborado um plano de estágio, que se encontra no anexo A, contendo a designação do projeto, o resumo do trabalho a desenvolver, diversos objetivos e o resultado previsto.

Todas as atividades previstas foram realizadas e apresentadas no final do estágio. Os objetivos listados eram o estudo e análise do *software Power BI*, bem como a ligação do referido *software* à base de dados *Microsoft SQL* da gestão de *tickets* da empresa Mestreclique e às bases de dados dos clientes da empresa. Além destes, ainda era previsto a identificação dos conjuntos de dados, a criação da visualização desses dados e a transformação dos dados utilizados. Por fim, e não menos importante, a publicação dos *dashboards* no *Power BI* online e, posteriormente, nos sites da Mestreclique e dos seus respetivos clientes.

# POLI TÉCNICO GUARDA

## **1.4 Estrutura do Relatório**

Este relatório é composto de seis capítulos de forma a facilitar a organização estrutural e, conseqüentemente, a leitura.

Capítulo 1 – introdução do presente relatório de estágio, abrangendo o enquadramento e motivação, a caracterização da instituição de acolhimento, os objetivos e a estrutura do relatório.

Capítulo 2 – estado da arte

Capítulo 3 – revisão da literatura

Capítulo 4 – descrição das tecnologias e ferramentas utilizadas no decorrer do estágio

Capítulo 5, 6, 7 e 8 – exposição, implementação e considerações do trabalho realizado

Capítulo 9 – conclusão com as apreciações finais sobre o trabalho realizado.



# POLI TÉCNICO GUARDA

## 2 Estado da Arte

### 2.1 Power BI

Segundo a *Microsoft*, o *Power BI* é uma coleção de serviços de *software*, aplicativos e conectores que trabalham juntos para transformar as fontes de dados não relacionadas em *insights* coerentes, visualmente imersivos e interativos.

Pelo décimo sexto ano consecutivo, a *Microsoft* ficou posicionada como líder de mercado dentro do Quadrante Mágico de Gartner (figura 1), em 2023, para Plataformas de Análise e *Business Intelligence*, graças ao *Power BI*.



Figura 1 - Quadrante Mágico de Gartner

Fonte: Gartner

# POLI TÉCNICO GUARDA

O *Power BI* é composto por diversos componentes que são interligados. Os principais componentes são (figura 2):

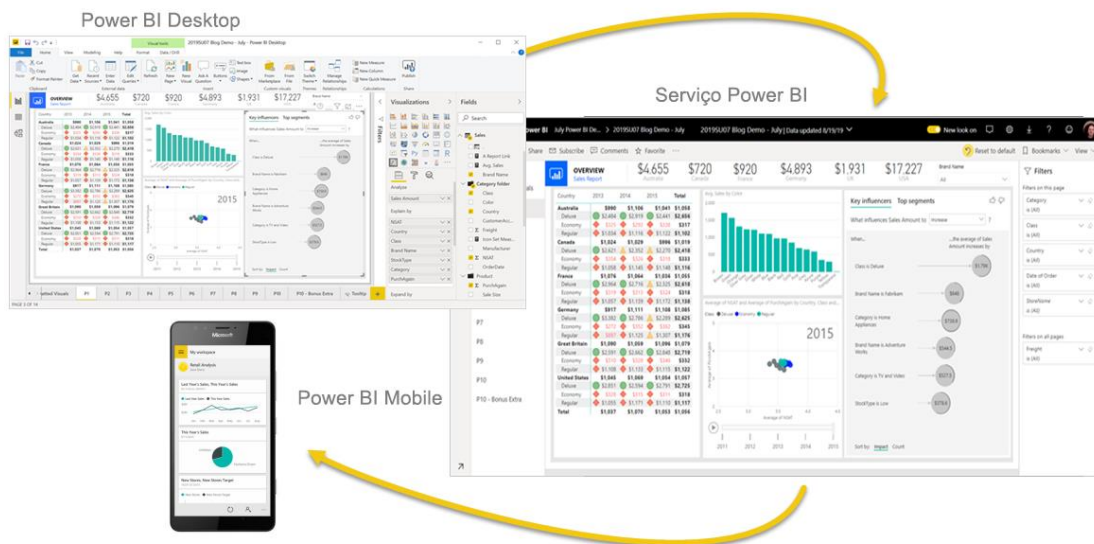


Figura 2 - Principais Componentes do *Power BI*

Fonte: Microsoft (2023)

**Power BI Desktop:** o *Power BI Desktop* é uma aplicação da área de trabalho do *Windows* onde os usuários podem criar, editar e publicar relatórios e *dashboards*. Ele tem uma interface intuitiva e diversificada em recursos para a importação de dados, transformação, modelagem e criação de visualizações. O *Power BI Desktop* para *Windows* pode ser baixado gratuitamente no site <https://www.microsoft.com/pt-pt/download/details.aspx?id=58494>.

**Serviço do Power BI:** o *Power BI Service* é uma plataforma online que permite compartilhar, colaborar e publicar os relatórios que foram criados no *Power BI Desktop*.

**Power BI Móvel:** o *Power BI Mobile* é um aplicativo que permite o acesso em dispositivos móveis, como *smartphones* e *tablets*, aos relatórios e *dashboards* criados no *Power BI Desktop* proporcionando, desta forma, uma mobilidade e acesso aos dados em tempo real e em qualquer lugar.

# POLI TÉCNICO GUARDA

Na busca pelo aprimoramento contínuo nas habilidades em *Power BI*, as plataformas educacionais disponibilizadas pela *Microsoft* foram amplamente exploradas. A página oficial da *Microsoft* oferece uma gama abrangente de recursos, que vai desde tutoriais com o passo a passo até cursos completos, dessa forma proporciona uma imersão profunda nas funcionalidades da ferramenta. Além disso, o *Power BI* apresenta a versão gratuita e as versões pagas, que apresentam mais diferenças na figura 3. A versão gratuita oferece robustas funcionalidade, permite a criação de relatórios e *dashboards* interativos e é adequado para pequenos projetos. As versões pagas desbloqueiam funcionalidades avançadas e a capacidade de uma maior colaboração em nível empresarial. A flexibilidade proporcionada por essas opções atende às diversas necessidades, desde usuários individuais até altas organizações. Na Mestreclique foi utilizada a versão *free*.

Power BI Licence	Free	Pro	Premium
Custo	Grátis	USD 7.99 /usuário /mês	> USD 4.100 /Licença por Capacidade
Limite de Dados	10GB	10GB	100TB
Tamanho máximo modelo de dados	1GB	1GB	12GB
Criar e Visualizar Dashboards e Reports	•	•	Apenas Visualizar
Compartilhar Dashboards e Reports com outros usuários		•	
Power BI Desktop	•	•	
Acesso via App Smartphone	•	•	
Consumir dados de Terceiros (SalesForce, Google Analytics)	•	•	
Importar dados do Excel, .csv e arquivos do Power BI	•	•	
Publicar Dashboards e Relatórios na Web	•	•	
Exportar dados para o Powerpoint e .csv		•	
Publicar no SharePoint		•	
Publicar no Premium		•	
Power BI Report Server (On Premise Server)		Premium	Premium
Atualização Automática	8 por dia	8 por dia	48 por dia
Tempo mínimo entre atualizações	30min	30min	1 min
Streaming de Dados (por hora)	1M	1M	
Fontes de dados Live		•	
Acessos de dados on premise via Gateways	•	•	
Utilizar Grupos do Office 365		•	
Criar e Publicar Apps		•	
Consumir Apps da Loja		•	Premium
Gerenciar acesso		•	

Figura 3 - Versões do *Power BI*

# POLI TÉCNICO GUARDA

## 2.2 Outros Softwares de *Business Intelligence*

Os outros dois *softwares* que estão próximos ao *Power BI* dentro do quadrante de Gartner são o *Tableau* e o *Qlik*, também consideradas como líderes.

O *Tableau* é uma plataforma que foi fundada em 2003 na Califórnia tendo sido adquirida pela *Salesforce* em 2019. A ferramenta é conhecida por sua facilidade de uso e por fornecer visualizações interativas de forma instantânea. O *Tableau* também oferece conectividade a diversas fontes de dados, bem como suporta análises avançadas de um conjunto de dados.

O *Qlik*, que está situado logo abaixo do *Tableau* no Quadrante Mágico de Gartner, destaca-se pela sua capacidade de análise associativa. Ele permite a exploração de dados de forma livre, sem a necessidade de criar previamente modelos de dados rígidos. O *Qlik* oferece recursos de arrastar e soltar e oferece uma experiência de usuário altamente interativa. É a plataforma que tem uma menor comunidade no mercado quando comparada aos outros líderes.

Tanto o *Power BI* quanto o *Tableau* e *Qlik* são ferramentas líderes de *BI* e possuem seus pontos fortes e fracos. A escolha dependerá da necessidade, estratégia e características específicas de cada empresa.

# POLI TÉCNICO GUARDA

## 3 Revisão da Literatura

Antes de abordar os detalhes dos *dashboards*, é fundamental compreender os principais conceitos de *Business Intelligence* e do *Power BI* para um melhor entendimento teórico sobre o que será explanado posteriormente.

### 3.1 *Business Intelligence*

O termo *Business Intelligence*, também conhecido como *BI*, foi criado na década de 80. De uma maneira prática, consiste na tomada de decisões empresarias baseadas em dados. O *BI* é frequentemente referido como técnicas, tecnologias, sistemas, práticas, metodologias e aplicações que analisam dados críticos de negócios para ajudar a empresa a compreender melhor os seus negócios e mercados e tomar decisões de forma oportuna (Chen, Hsinchun & Chiang, Roger & Storey, Veda. (2012)).

A análise de dados desempenha um papel vital no *BI* pois permite a descoberta de padrões, tendências e, por vezes, informações ocultas nos dados. O grande objetivo do *BI* é ter uma fácil interpretação de grandes volumes de dados.

O *Business Intelligence* envolve, na verdade, muito pouco do novo, pois resolve problemas antigos com os quais os gestores sempre estiveram ocupados. Representa uma tarefa de gestão básica – analisar um ambiente de negócios complexos para tomar as melhores decisões possíveis. No entanto, uma verdadeira novidade da inteligência empresarial é a sua capacidade de apresentar informações empresariais de uma forma rápida, simples e eficiente, para que os utilizadores possam compreender a lógica e o significado das informações empresariais, empregando uma ampla gama de possibilidades analíticas (Jaklič, Jurij. (2008)).

Dessa forma, a tomada de decisões baseia-se em dados reais e não apenas em intuições e/ou suposições. O *BI* é uma ferramenta que apresenta a possibilidade de monitorar o desempenho, identificar as oportunidades e detetar os problemas fornecendo rápidas respostas às mudanças nas condições de negócios.

# POLI TÉCNICO GUARDA

Os sistemas de *BI* são usados para melhorar a qualidade das decisões tomadas e fornecer soluções oportunas para diversos problemas, desde problemas altamente estruturados até os altamente não estruturados. A intenção é que a experiência de gestão melhore com as ferramentas de *BI* com o objetivo de se tomarem melhores decisões.

## **3.2 Power BI**

O *Power BI* é a ferramenta de *Business Intelligence* da *Microsoft*. É uma ferramenta poderosa e versátil que possibilita a criação de relatórios interativos, painéis dinâmicos e avançadas análises de dados. Se destaca pela capacidade de transformar dados complexos em claras e informativas visualizações.

*Dashboard* pode ser definido como uma ferramenta de gestão de desempenho visual e interativo que exibe em uma única tela as informações mais importantes necessárias para atingir um ou vários objetivos individuais e/ou organizacionais, permitindo ao usuário identificar, explorar e comunicar áreas problemáticas que necessitam de ação corretiva (Velcu-Laitinen, Oana & Yigitbasioglu, Ogan. (2012)).

O *Power BI* permite que o usuário se conecte a diferentes fontes de dados (txt, Excel, CSV, banco de dados, dados de internet, etc) para extraí-los, tratá-los, gerar indicadores de desempenho e criar *dashboards*. Tudo isso fez com que esse *software* se tornasse o líder do mercado, segundo a Gartner<sup>1</sup>, e contribuísse fortemente para a democratização e popularização das tecnologias de *Business Intelligence*.

---

<sup>1</sup> A Gartner é uma empresa líder mundial focada em pesquisa e consultoria em diversas áreas da tecnologia da informação.

# POLI TÉCNICO GUARDA

Na figura 4, pode-se observar o fluxo de funcionamento do *software*, onde os dados, representados por ícones de várias fontes possíveis, são integrados e transformados em informação.

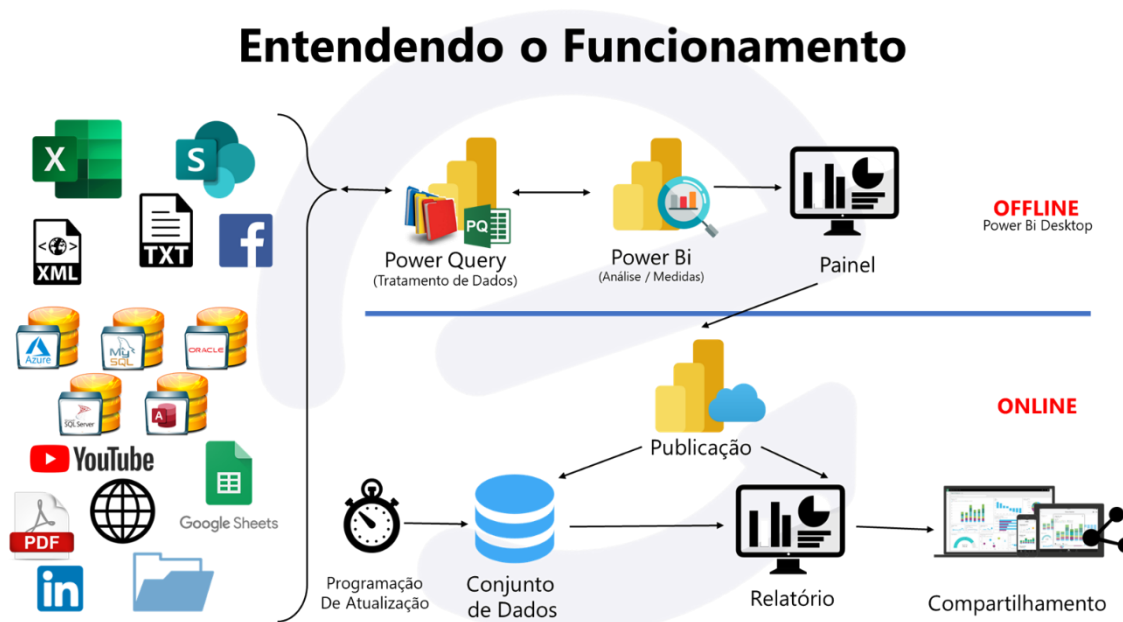


Figura 4 - Fluxo do processo de funcionamento do *Power BI*

Fonte: [https://www.consultoriaexcelencia.com.br/detalhes\\_pbi.php](https://www.consultoriaexcelencia.com.br/detalhes_pbi.php)

## 3.2.1 Importação e Preparação de Dados

Esta parte do relatório tem como objetivo fornecer uma compreensão abrangente do funcionamento do *Power BI*, destacando as principais características e capacidades.

Uma das funcionalidades fundamentais do *Power BI* é a capacidade de importar, transformar e preparar os dados para a devida análise. Os usuários podem se conectar a uma grande variedade de fontes de dados, como, por exemplo, banco de dados, arquivos locais, serviços *online* e outras.

# POLI TÉCNICO GUARDA

Através do *Power Query* é possível realizar a limpeza, transformação e modelagem dos dados para atender às necessidades específicas de cada projeto. O *Power Query* é uma ferramenta de ETL (*Extract, Transform, and Load*) de preparação e transformação de dados que permite conectar-se a múltiplas fontes de dados e é incorporada ao *Power BI* (figura 5).

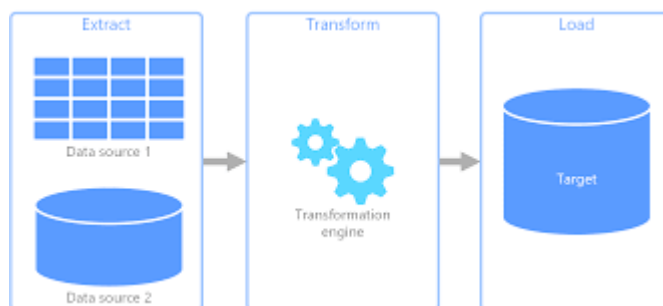


Figura 5 - Processo ETL

Fonte: <https://learn.microsoft.com/pt-pt/azure/architecture/data-guide/relational-data/etl>

Principais recursos e funcionalidade do *Power Query*:

1. Conexão a várias fontes de dados:

O *Power Query* suporta uma ampla variedade de fontes de dados, desde bancos de dados relacionais, planilhas e arquivos de texto até serviços web, fontes de dados em nuvem e muito mais. É possível a importação de dados de diversas fontes em um único local.

2. Transformação de dados:

Os usuários podem aplicar uma série de transformações aos dados importados. Essa é uma das principais funções do *Power Query*. Remover dados duplicados, renomear colunas, filtrar dados, combinar tabelas, criar colunas calculadas, corrigir a formatação e realizar outras manipulações garantem que os dados estejam prontos para análise. Nenhuma das alterações realizadas no *Power Query* serão aplicadas na base de dados. A BD continua intacta e não é afetada pelas alterações realizadas no *Power Query*.



# POLI TÉCNICO GUARDA

## 3. Interface visual e passos aplicados:

A interface visual oferecida pelo *Power Query* é amigável e permite que os usuários vejam todas as etapas de transformação que foram aplicadas aos dados. Dessa forma é facilitada a depuração e o rastreamento das transformações podendo reutilizar as etapas em diferentes conjuntos de dados.

## 4. Linguagem M:

O *Power Query* utiliza a linguagem M que é uma linguagem de programação funcional e utilizada para definir as transformações dos dados. Mesmo que a interface gráfica seja amigável e bastante utilizada, nada impede que os usuários avançados escrevam expressões personalizadas com a finalidade de realizar tarefas complexas na preparação dos dados.

## 5. Atualizações automáticas:

Os dados podem ser atualizados constantemente, em tempo real, a partir de fontes de dados conectadas, garantindo, dessa forma, que os relatórios e *dashboards* estejam sempre com as informações mais recentes.

## 6. Conexões a fontes de dados na nuvem:

Além de fontes de dados locais, o *Power Query* pode se conectar a serviços em nuvem, como *Azure Data Lake Storage*, *SharePoint Online*, *Salesforce*, entre outros, tornando-o uma ferramenta versátil para empresas que utilizam soluções em nuvem.

# POLI TÉCNICO GUARDA

A interface do *Power Query* está dividida em cinco componentes distintos, conforme pode ser visualizado na figura 6.

1. Menu de Opções ou Friso: é composto de vários separadores que possuem várias funções para realizar várias tarefas nos dados seleccionados;
2. Painel de Consultas: local onde pode ser encontrada a lista de consultas disponíveis para utilização;
3. Modo de Exibição Atual: local que exhibe o conteúdo das consultas;
4. Configurações de Consulta: exposição das propriedades e de todas as modificações feitas, passo a passo, de uma consulta;
5. Barra de Status: barra que exhibe informações importantes sobre a consulta como o total de linhas e colunas.

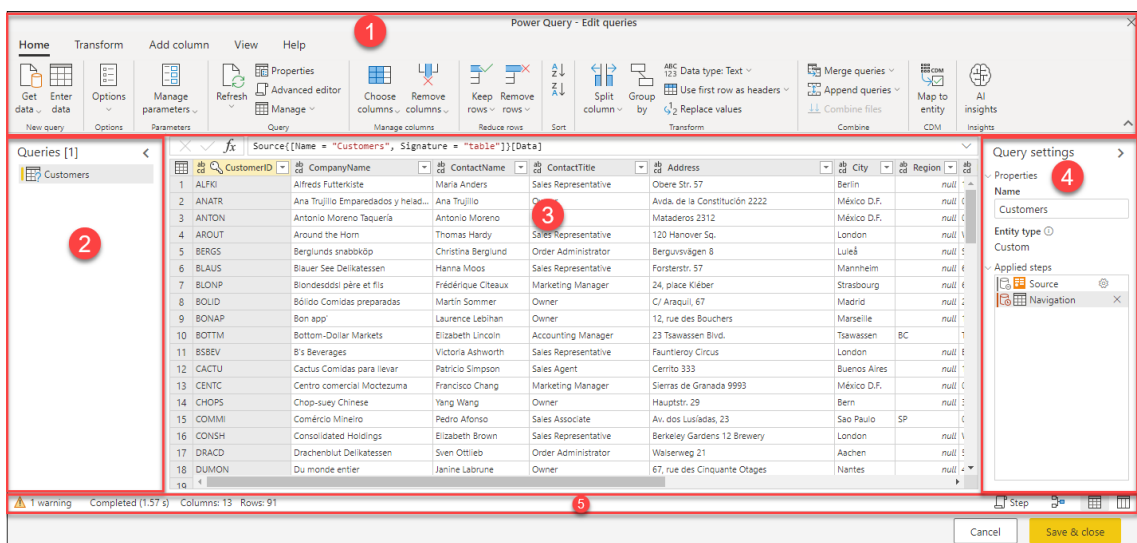


Figura 6 - Interface do *Power Query*

Fonte: <https://learn.microsoft.com/pt-pt/Power-query/Power-query-ui>

# POLI TÉCNICO GUARDA

## 3.2.2 Criação de Visualizações

O *Power BI* oferece uma extensa biblioteca de visualizações gráficas, como gráficos de pizza, gráficos de barras, mapas geográficos, cada qual com sua especificidade. É possível arrastar e soltar os elementos visuais nos relatórios e personalizá-los da maneira que melhor convir para transmitir informações de maneira clara e eficaz. Os elementos de visualização no *Power BI* variam desde a exibição de um único número, através de um cartão, até um mapa colorido.

A parte esquerda do *Power BI Desktop* possui três ícones de exibição: Relatório, Dados e Modelo, conforme pode ser visualizado na figura 7. Uma barra verde, localizada à esquerda do ícone, indica a exibição atual.

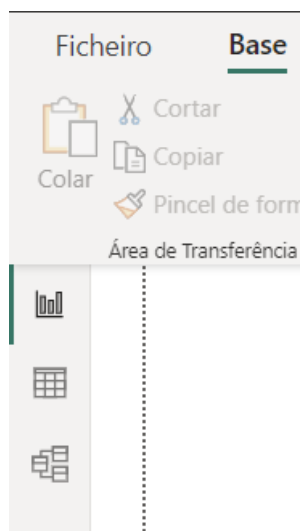


Figura 7 - Ícones de Exibição no *Power BI Desktop*

# POLI TÉCNICO GUARDA

É possível criar os relatórios na exibição de Relatório do *Power BI*. Esta exibição possui sete áreas principais (figura 8):

1. Faixa de Opções que fica na parte superior e exibe tarefas comuns;
2. Área da tela onde podem ser criadas e organizadas as visualizações;
3. Guia de páginas que permite adicionar páginas de relatório, bem como selecionar a página desejada;
4. Painel de Filtros em que é possível adicionar filtros às visualizações dos dados;
5. Painel de Visualizações no qual é possível adicionar, alterar e personalizar as visualizações e, ainda, aplicar detalhamento;
6. Painel Formatar onde se formata o relatório;
7. Painel Campos que mostra os campos disponíveis nas consultas.

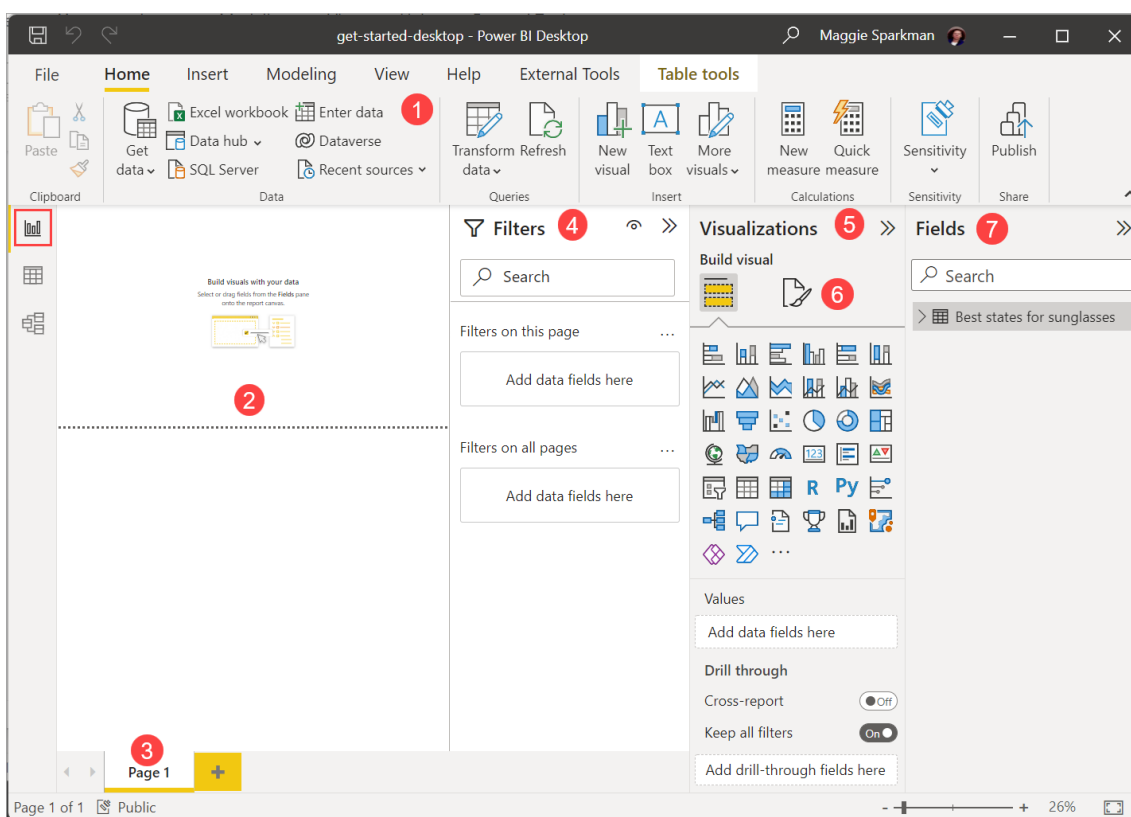


Figura 8 - Modo Exibição Relatório

# POLI TÉCNICO GUARDA

Além da Linguagem M, que pode ser utilizada no *Power Query*, tem-se a segunda linguagem chamada DAX (*Data Analysis Expressions*), utilizada pelo *Power BI*, e que também possibilita realizar análises com um maior critério. A linguagem DAX pode ser considerada como um conjunto de funções, variáveis, constantes e operadores e que podem ser utilizados em fórmulas com a finalidade de calcular e obter determinado valor.

### **3.2.3 Interação e Publicação**

Os relatórios e *dashboards* criados no *Power BI*, na grande maioria das vezes, são interativos e permitem que os usuários explorem os dados, apliquem filtros e obtenham *insights* dinâmicos. Além disso, os projetos podem ser publicados no *Power BI Service* podendo ser compartilhados com outras pessoas, editados colaborativamente e, ainda, acessados por meio de navegadores *web* e aplicativos móveis.

# POLI TÉCNICO GUARDA

## 4 Tecnologias e Ferramentas Utilizadas

### Linguagem DAX

A linguagem DAX, ou *Data Analysis Expressions*, é uma linguagem de expressão de fórmula utilizada no *Power BI*, bem como em outras ferramentas da *Microsoft*, como o *Analysis Services* e o *Power Pivot* do Excel. Sua principal finalidade é realizar cálculos e agregações em conjuntos de dados. Dessa forma, é possível criar medidas personalizadas e colunas e tabelas calculadas para que se faça uma análise da dados mais avançada. Uma fórmula DAX sempre começa com o sinal de igual (=). As funções DAX ajudam a encontrar informações detalhadas nos dados brutos das organizações.

### Linguagem M

A linguagem M, ou *Power Query Formula Language*, também é uma linguagem usada no *Power BI* para a transformação dos dados. Ela desempenha um papel fundamental na etapa de preparação dos dados e permite que os usuários importem, estruturem, limpem e transformem conjuntos de dados brutos em formatos mais adequados para análise. É esta linguagem que fica registada nas etapas do editor de consulta do *Power Query* e seus códigos podem ser acessados através do botão “Editor Avançado” (figura 9).

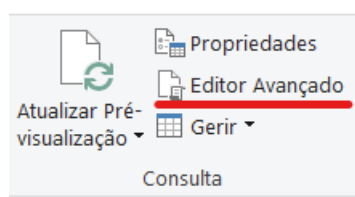


Figura 9 - Forma de acessar os códigos da Linguagem M

# POLI TÉCNICO GUARDA

## **Linguagem SQL**

*Structured Query Language (SQL)* é uma linguagem de programação, utilizada para modelos relacionais, que possui alguns objetivos como a consulta, manipulação, transação e controle dos dados. Ela é a interface entre o utilizador e o sistema que gere a base de dados, tal como o *SQL Server*.

## ***Power BI Desktop***

O *Power BI Desktop* é um aplicativo gratuito que oferece recursos de criação de relatórios e modelagem de dados. Ele permite que os usuários importem dados de diversas fontes, projetem visualizações personalizadas e criem medidas calculadas fazendo com que seja amplamente utilizado para tomar decisões baseadas na análise dos dados.

## ***Power BI Service***

O *Power BI Service*, também conhecido como *Power BI Online*, é uma plataforma SaaS (*Software as a Service*) que funciona diretamente nos navegadores *web*, com *login* e senha. Nesta plataforma também é possível criar, editar e publicar relatórios. Outra possibilidade é criar espaços de trabalho, ou *workspaces*, para todos os colaboradores. O *Power BI Online* também permite que os usuários programem as atualizações automáticas para os dados do *dashboard*.

## ***SQL Server***

*SQL Server* é um sistema gerenciador de banco de dados (SGBD) que foi desenvolvido pela *Microsoft*. Ele permite que seus usuários armazenem, acessem e manipulem dados de forma eficiente. O *SQL Server* utiliza a linguagem SQL para executar as operações no banco de dados, como inserção, consulta, atualização e exclusão. Com este *software* pode-se criar e gerenciar bancos de dados relacionais, estabelecer os relacionamentos entre as tabelas, entre outras funções.

# POLI TÉCNICO GUARDA

## *FileZilla*

O *FileZilla* é um popular *software* de código aberto (*Open Source*) que é amplamente utilizado para a transferência de arquivos maiores com segurança e rapidez. Ele envia arquivos para um servidor através do protocolo FTP (*File Transfer Protocol*). Possui uma interface amigável que torna mais fácil realizar o upload de arquivos para o servidor. O *FileZilla* também suporta SFTP (*Secure File Transfer Protocol*), protocolo muito seguro e que permite vários comandos que normalmente o FTP não aceita.



# POLI TÉCNICO GUARDA

## 5 O trabalho realizado: O Modelo de Dados

Durante o meu estágio na empresa Mestreclique, tive a oportunidade de aplicar meus conhecimentos em Análise de Dados e *BI* para desenvolver dois *dashboards* no *Power BI*. O primeiro *dashboard* foi projetado para atender às necessidades internas da empresa Mestreclique, servindo de suporte ao controlo de gestão, incluindo relatórios que abrangiam os seus funcionários e seus clientes, englobando os *tickets* que são abertos pelos clientes. O segundo *dashboard* foi criado especialmente e exclusivamente para os clientes da Mestreclique, apenas na componente “Inscrição dos Formandos” proporcionando-lhes uma experiência personalizada e informativa. A ideia desse segundo projeto era demonstrar aos clientes, de forma gratuita, um pouco do funcionamento dessa ferramenta poderosa que é o *Power BI* para, então, incluir esse tipo de análise como mais um serviço oferecido pela Mestreclique.

Neste relatório, descrevo em detalhes o processo de criação desses *dashboards*, desde a coleta e preparação dos dados até a escolha das visualizações e análises realizadas.

Os indicadores são essenciais ao planejamento e controle dos processos organizacionais, pois constituem a base do planejamento. Estabelecem medidas verificadoras do cumprimento de metas e objetivos e sinalizam o rumo que a organização está seguindo. Assim, facilita a ação da gerência, proporcionando maior respaldo na tomada de decisão (Fernandes, D. R. (2017)).

Inicialmente, o pedido feito pela Mestreclique foi a confecção de um *dashboard* que contivesse os seguintes indicadores:

1. Número de clientes ativos;
2. Número de *tickets* respondidos;
3. Número de *tickets* respondidos por técnico;
4. Tempo gasto na resolução dos *tickets*;
5. Grau de satisfação dos clientes;
6. Tempo mensal gasto por cliente.

# POLI TÉCNICO GUARDA

Já com relação ao *dashboard* dos clientes, como era apenas uma amostra, os indicadores foram escolhidos com base na componente “inscrição dos formandos”. Seguem os indicadores iniciais:

1. Inscrição por faixa etária;
2. Formas de conhecimento;
3. Tipo de interesse;
4. Perfil dos candidatos a formações que se inscrevem pela web (faixa etária, concelhos de residência, habilitações, área de base)
5. Perfil dos formandos que frequentam cursos de formação não financiada (faixa etária, concelhos de residência, habilitações, área de base);
6. Entidades empregadoras que mais formandos tiveram participantes em ações de formação.

Após a escolha dos indicadores, procedeu-se à fase ETL onde todos os dados previamente escolhidos foram extraídos, transformados e carregados.

# POLI TÉCNICO GUARDA

## 5.1 RECOLHA E SELEÇÃO DOS DADOS

Na primeira fase do processo ETL, foi realizada a ligação à base de dados da Mestreclique gerenciada pelo *SQL Server* utilizando o conector oficial do *Power BI*. Para esta etapa, foi usada a opção “Obter Dados” do *Power BI* (figura 10).

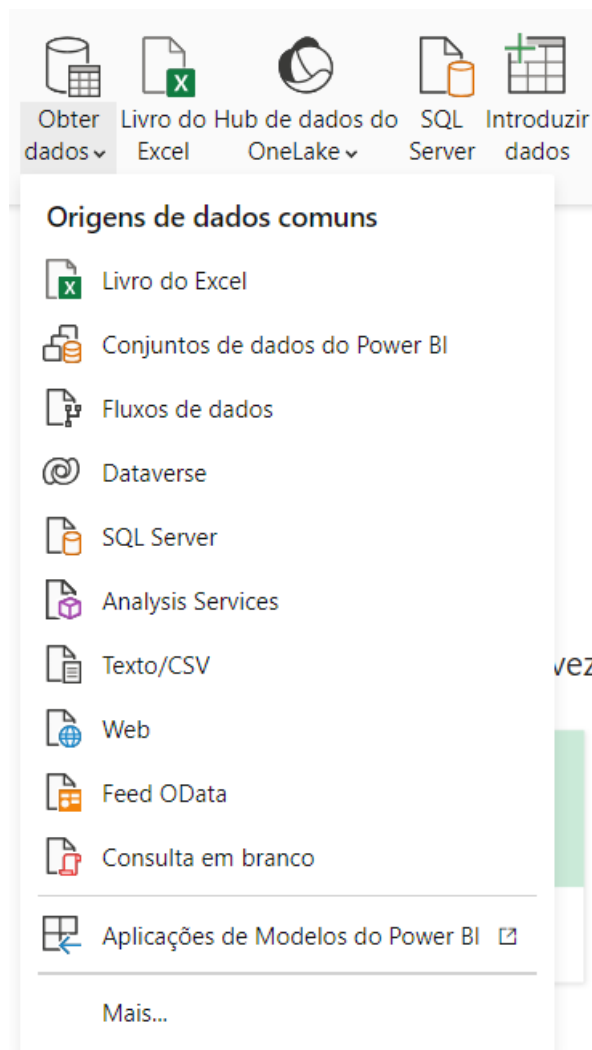
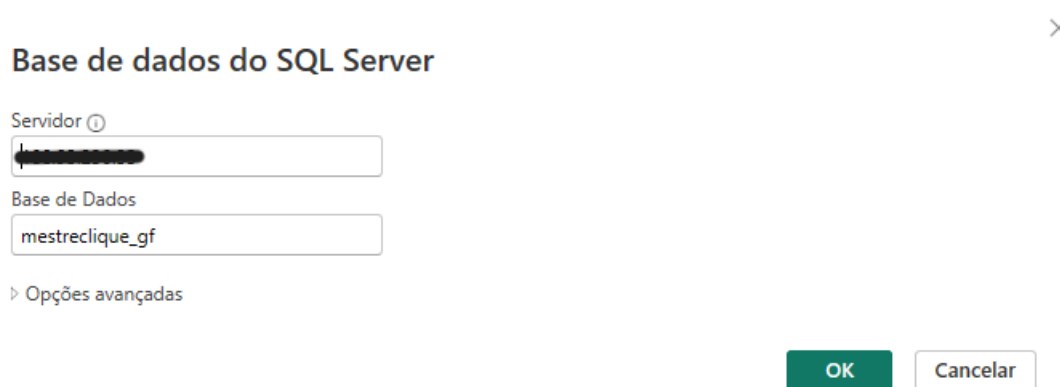


Figura 10 - Funcionalidade “Obter Dados”

# POLI TÉCNICO GUARDA

Em seguida, foi realizada a extração dos dados através de uma ligação direta ao *SQL Server*, após o preenchimento dos campos solicitados, conforme ilustrado na figura 11.



The image shows a dialog box titled "Base de dados do SQL Server" with a close button (X) in the top right corner. It contains two input fields: "Servidor" with a redacted value and "Base de Dados" with the value "mestreclique\_gf". Below the fields is a link "Opções avançadas" and two buttons: "OK" and "Cancelar".

Figura 11 – Preenchimento dos campos para ligar ao *SQL Server*

Ao clicar em “OK” as tabelas são mostradas e, de acordo com os indicadores sugeridos, as tabelas devidas foram selecionadas.

Esta etapa é realizada da mesma forma em ambos os projetos.

## 5.2 TRANSFORMAÇÃO DOS DADOS

Na segunda fase do processo ETL é preciso efetuar a transformação dos dados. Essa transformação refere-se à limpeza e agregação dos dados para que estes fiquem corretamente preparados para análise. As possibilidades de alteração dos dados e manipulação são imensas, tais como:

- Limpeza dos dados:
  - Remoção de valores nulos
  - Remoção de linhas duplicadas
  - Conversão dos tipos de dados (ex.: texto para números)
- Enriquecimento dos dados:
  - Agregação dos dados
  - Complementação dos dados
  - Divisão das informações de um campo



# POLI TÉCNICO GUARDA

## 5.2.1 CRIAÇÃO DA TABELA CALENDÁRIO

Uma das transformações necessárias, utilizando o *Power Query*, foi a criação da tabela Calendário, onde constam apenas datas, sem repetições, e seus atributos. Ela é utilizada para desenvolver funções de inteligência de tempo e ajudar nas análises, através dos relacionamentos, que utilizarão comparações entre datas. Essa tabela deve ser criada de uma forma dinâmica, que atualize as datas de acordo com as datas existentes na base de dados.

Para criar essa nova tabela, foi necessário clicar no botão “Nova Origem” e escolher a opção “Consulta em Branco” (figura 13).

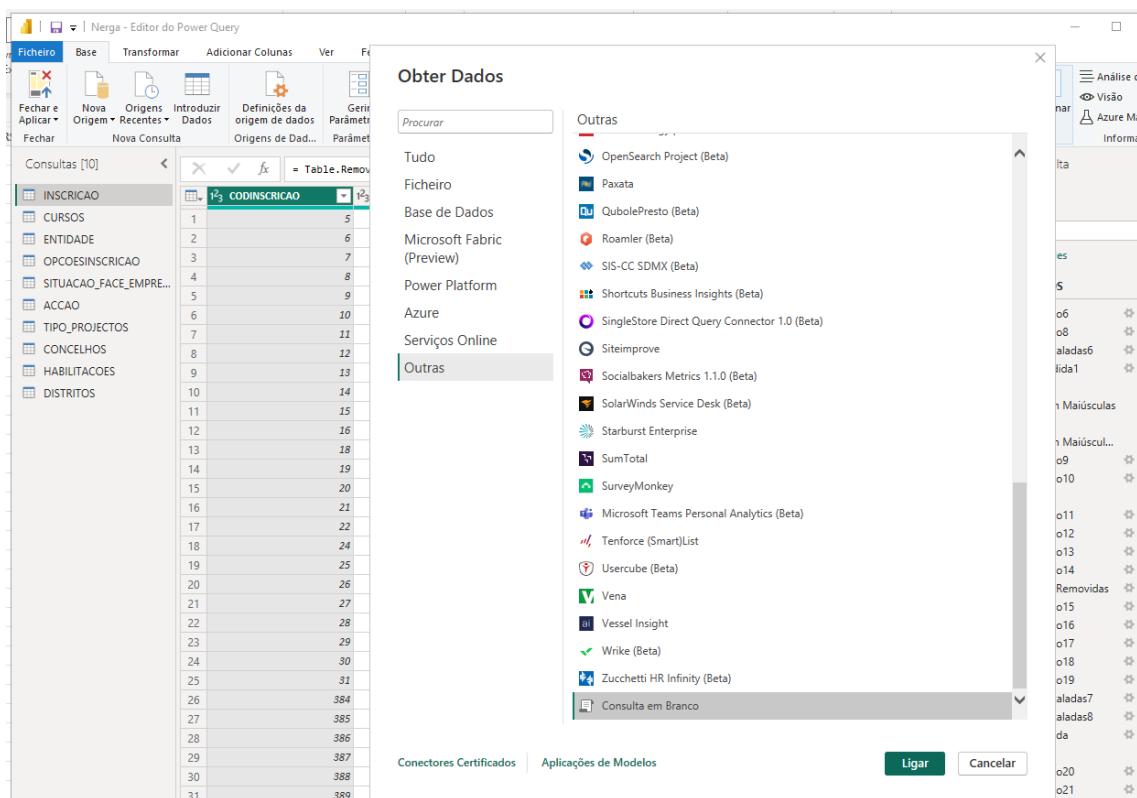


Figura 13 - Escolha de uma consulta em branco para criação da tabela calendário

# POLI TÉCNICO GUARDA

Feito isso, clicou-se no botão “Editor Avançado” para abrir a janela da “Consulta em Branco” que foi criada (figura 14).

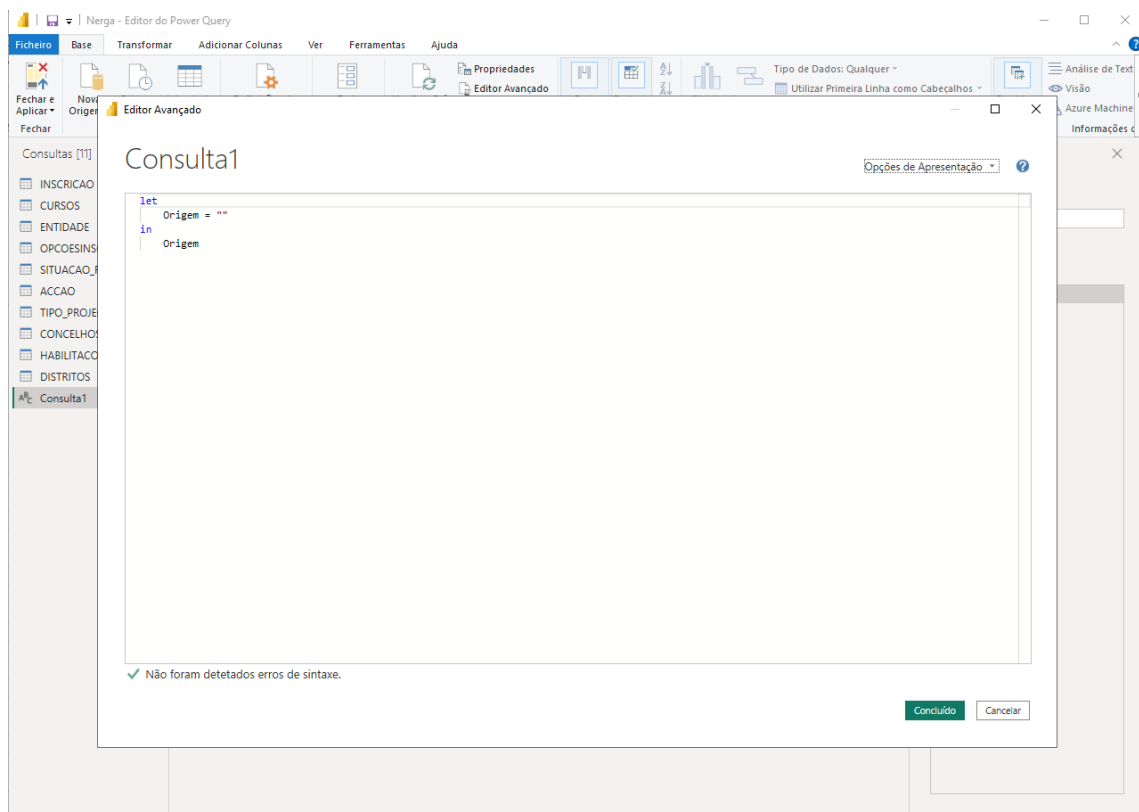
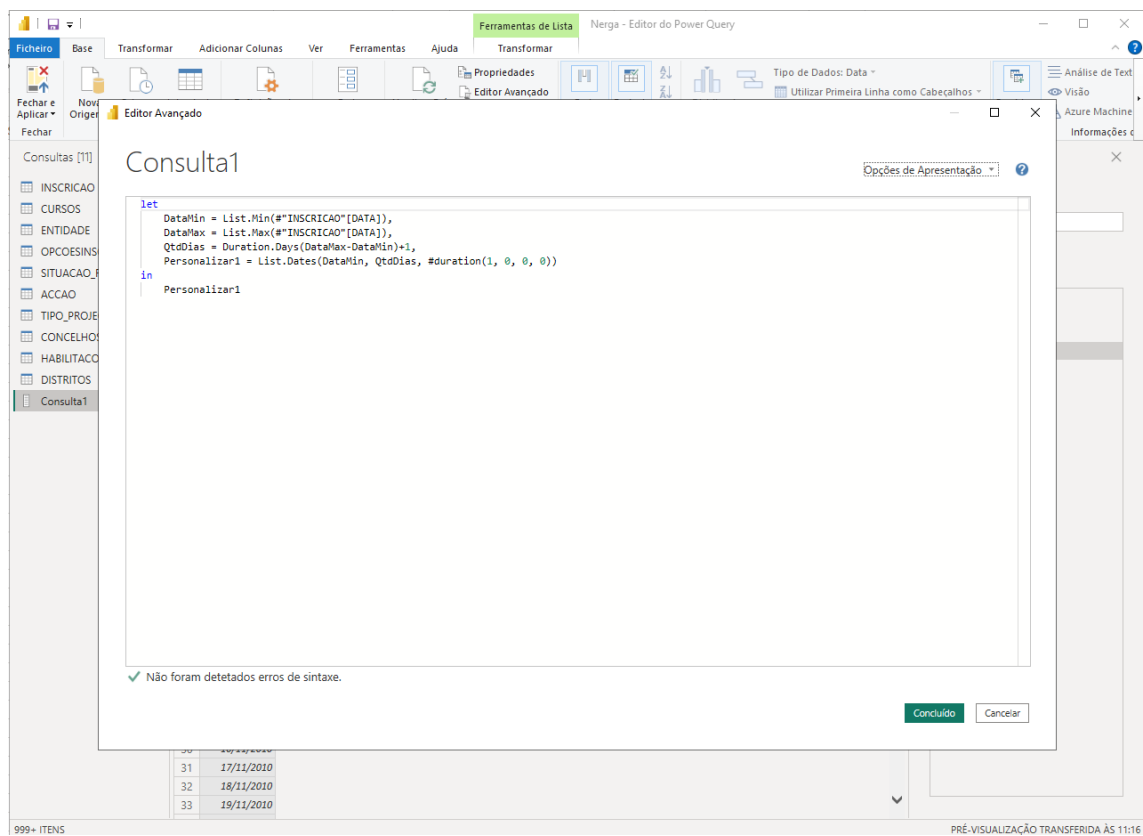


Figura 14 - Editor avançado da consulta em branco

# POLI TÉCNICO GUARDA

A lista de datas deve começar no primeiro dia em que ocorreu uma inscrição e terminar no último dia que houve alguma inscrição. Como pode ser visualizado na figura 15, deve-se descobrir a menor data (DataMin), a maior data (DataMax), a quantidade de dias entre as duas datas (QtuDias) e, finalmente, retornar todas as datas com a utilização da fórmula *List.Dates*, ocorrendo a cada dia (de 1 em 1 dia).



The screenshot shows the Power Query Editor interface. The main area displays a query named 'Consulta1' with the following M code:

```
let
    DataMin = List.Min(*"INSCRICAO"[DATA]),
    DataMax = List.Max(*"INSCRICAO"[DATA]),
    QtuDias = Duration.Days(DataMax - DataMin) + 1,
    Personalizar1 = List.Dates(DataMin, QtuDias, #duration(1, 0, 0, 0))
in
    Personalizar1
```

Below the code editor, a preview of the resulting data is visible, showing a list of dates:

31	17/11/2010
32	18/11/2010
33	19/11/2010

The status bar at the bottom indicates '999+ ITENS' and 'PRÉ-VISUALIZAÇÃO TRANSFERIDA ÀS 11:16'.

Figura 15 - Código para criação da tabela calendário



# POLI TÉCNICO GUARDA

Após clicar em concluído, foi criada uma lista de datas. Para converter a lista em tabela, clicou-se na opção “Transformar” e no botão “Para a Tabela” (figura 16).

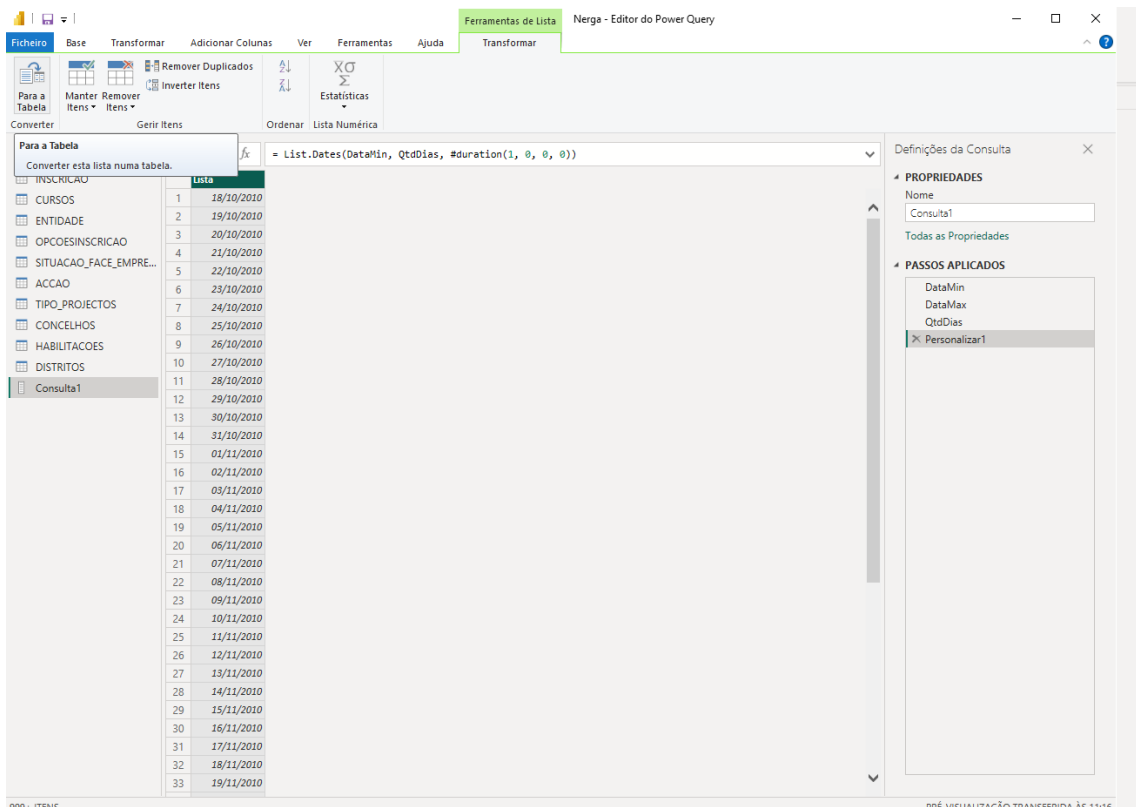


Figura 16 - Transformação da lista em tabela

# POLI TÉCNICO GUARDA

O nome da coluna foi alterado para “Data” e o tipo foi alterado de texto (ABC) para data, fazendo com que surgisse um símbolo de calendário ao lado do nome da coluna (figura 17).

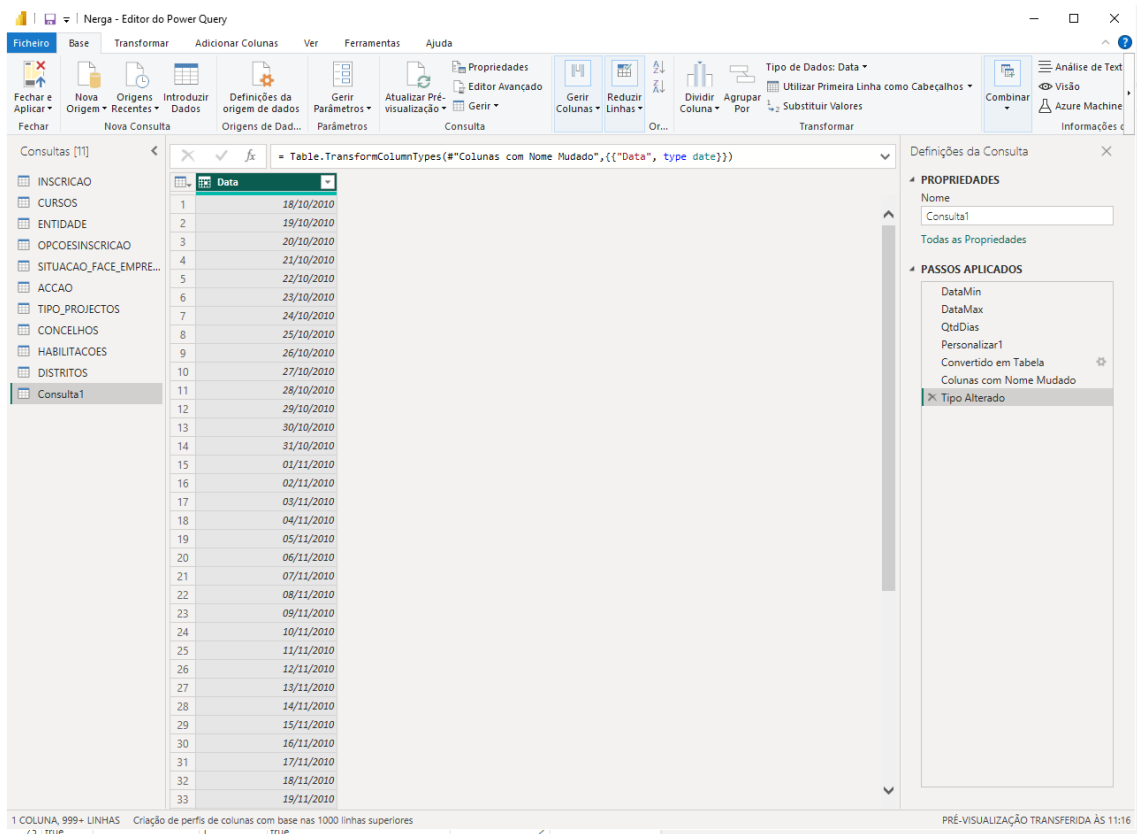


Figura 17 - Alteração do nome e tipo da coluna

# POLI TÉCNICO GUARDA

Por fim, para criar as colunas auxiliares de datas clicou-se no menu “Adicionar Colunas” e, em seguida, na opção “Data”, conforme pode ser visualizado na figura 18. Além de ter sido renomeado o nome da nova tabela para “Calendário”.

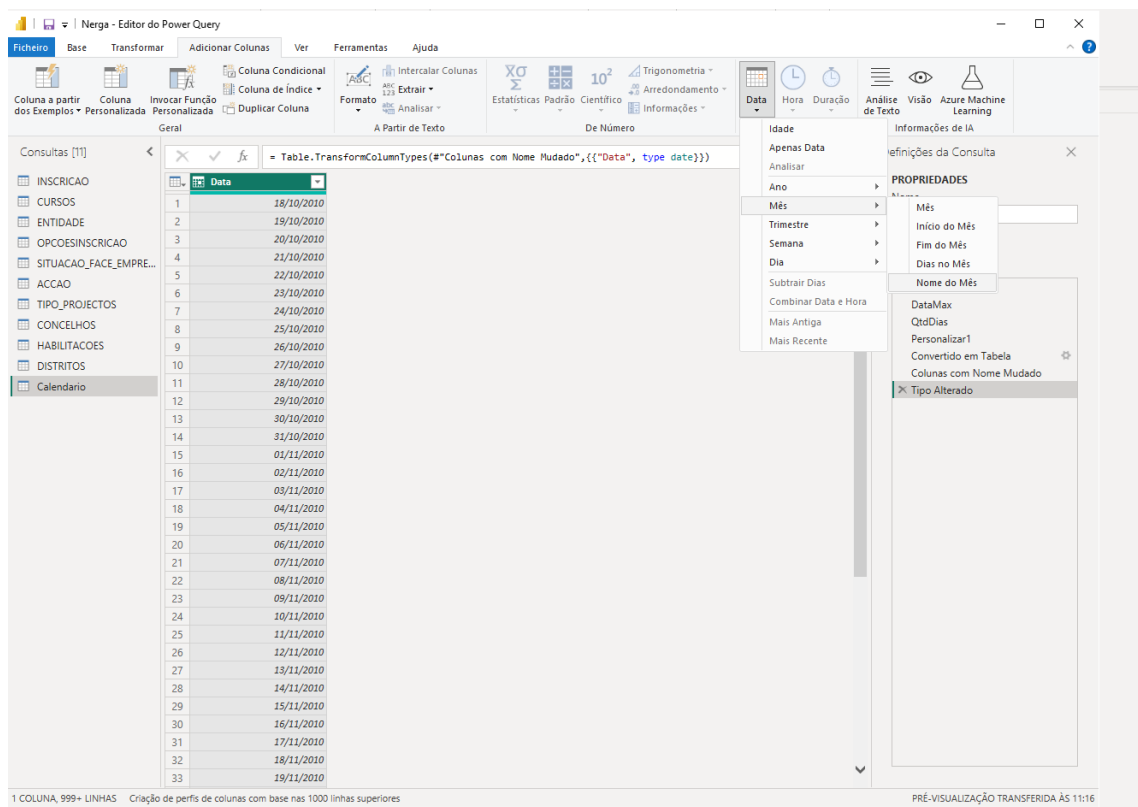



Figura 18 - Forma de criar as colunas auxiliares de datas

# POLI TÉCNICO GUARDA

## 5.2.2 DADOS DA MESTRECLIQUE

As primeiras transformações foram realizadas na tabela dimensão “*TICKETS*”, conforme pode ser visto na figura 19, onde as alterações estão em linguagem M.



```
let
    Origem = Sql.Database(██████████, "mestreclique_gf"),
    dbo_T_TICKETS = Origem([Schema="dbo",Item="T_TICKETS"])[Data],
    #"Colunas Removidas" = Table.RemoveColumns(dbo_T_TICKETS,{"DESCRICAO", "ASSUNTO", "RESPOSTA", "FICHEIRO", "PRINTSCREEN", "USERINSERT", "NOMECLIENTE", "OBSERVACOES"},
    #"Coluna Condicional Adicionada" = Table.AddColumn(#"Colunas Removidas", "EstadoTicket", each if [CODESTADO] = 1 then "Aberto" else if [CODESTADO] = 2 then "Respo",
    #"Consultas Intercaladas" = Table.NestedJoin(#"Coluna Condicional Adicionada", {"CODTICKET"}, TICKETS_ESTADOS, {"CODTICKET"}, "T_TICKETS_ESTADOS.1", JoinKind.Left",
    #"Adicionado Tempo Resposta" = Table.AggregateTableColumn(#"Consultas Intercaladas", "T_TICKETS_ESTADOS.1", {"TEMPO", List.Sum, "Tempo Resposta"}),
    #"Adicionada a Entidade" = Table.ExpandRecordColumn(#"Adicionado Tempo Resposta", "ENTIDADE(CODCLIENTE)", {"NOME"}, {"Nome da Entidade"}),
    #"Tipo Alterado" = Table.TransformColumnTypes(#"Adicionada a Entidade",{{"Tempo Resposta", Int64.Type}},
    #"Consultas Intercaladas1" = Table.NestedJoin(#"Tipo Alterado", {"CODTECNICO"}, ENTIDADE, {"CODENTIDADE"}, "ENTIDADE", JoinKind.LeftOuter",
    #"ENTIDADE Expandida" = Table.ExpandTableColumn(#"Consultas Intercaladas1", "ENTIDADE", {"NOME"}, {"ENTIDADE.NOME"}),
    #"Linhas Filtradas" = Table.SelectRows(#"ENTIDADE Expandida", each true),
    #"Tipo Alterado1" = Table.TransformColumnTypes(#"Linhas Filtradas",{{"DATAINSERT", type date}, {"DATAFIN", type date}})
in
    #"Tipo Alterado1"
```

✓ Não foram detetados erros de sintaxe.

Concluído Cancelar

Figura 19 - Transformações realizadas, em linguagem M, na tabela *TICKETS*

Uma vez que existiam diversas tabelas e uma grande quantidade de dados, as colunas irrelevantes foram removidas, ficando apenas as colunas necessárias para a devida análise dos dados. Este passo foi repetido nas demais tabelas.

Para mostrar um maior nível de detalhe e facilitar a compreensão, foi adicionada uma nova coluna “*EstadoTicket*” através de uma formatação condicional, no qual o estado do *ticket* será identificado de acordo com o código que consta na coluna “*CODESTADO*”. Essa etapa foi feita através da opção “Coluna Condicional” no *Power Query* (figura 20).

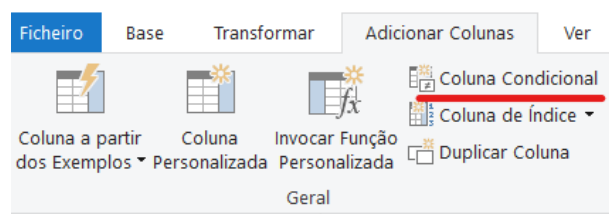


Figura 20 - Função Adicionar Coluna Condicional

# POLI TÉCNICO GUARDA

Na janela que se abriu foi escolhido o nome da coluna e os devidos campos foram preenchidos (figura 21).

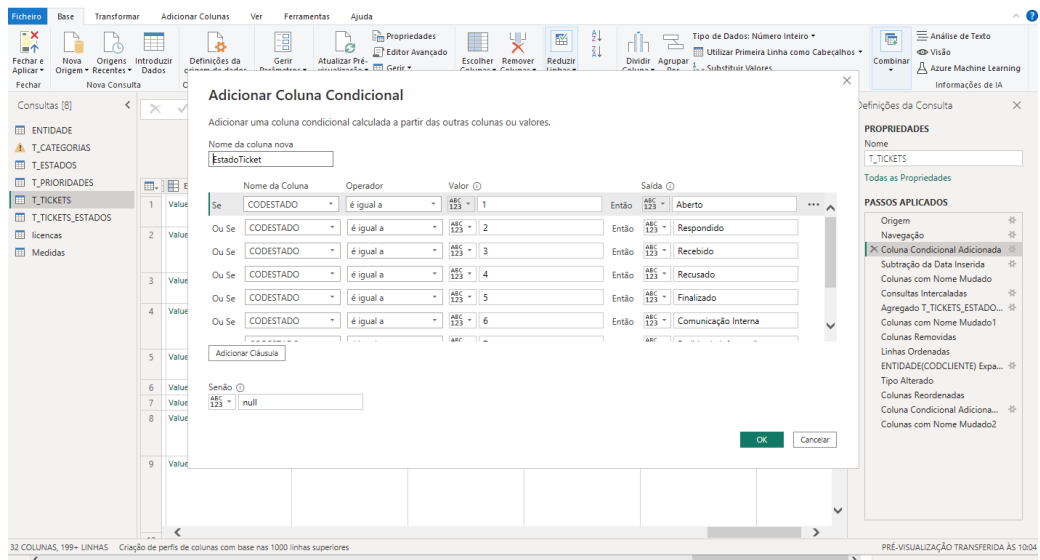


Figura 21 - Campos preenchidos na função Adicionar Coluna Condicional

Foi necessário associar as tabelas “*TICKETS*” e “*TICKETS\_ESTADOS*”. Este passo foi feito utilizando a opção “Intercalar Consultas” no menu visual do *Power Query* (figura 22).

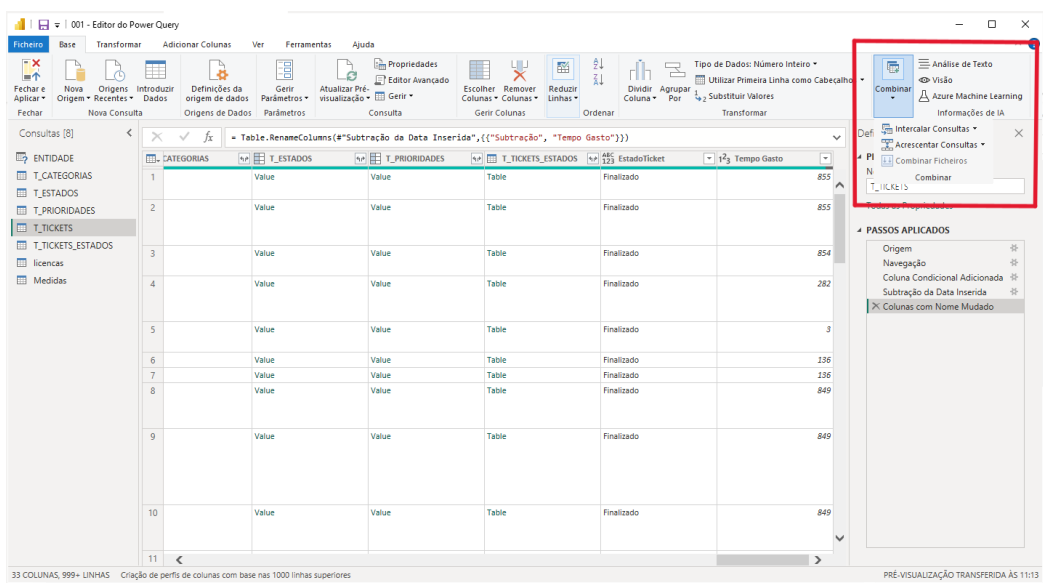


Figura 22 - Opção Intercalar Consultas no *Power Query*

# POLI TÉCNICO GUARDA

A intercalação foi realizada através da coluna “CODTICKET” da tabela *TICKETS\_ESTADOS* para a tabela *TICKETS*, conforme visualizado na figura 23.

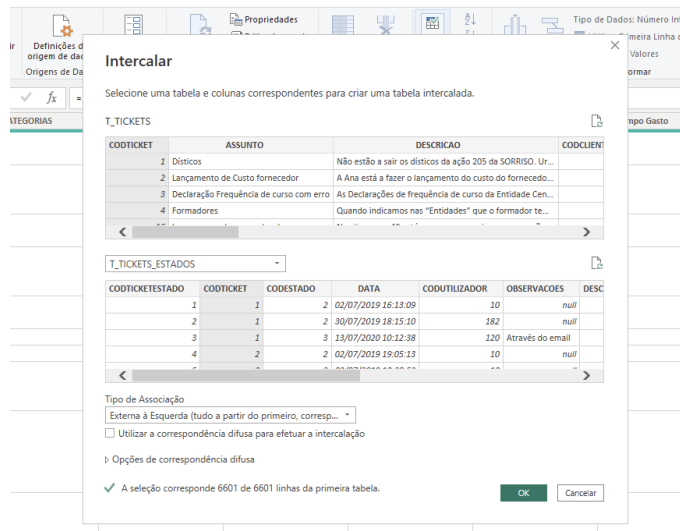


Figura 23 - Função Intercalar Consultas

Desta forma foi possível adicionar uma nova coluna e obter o Tempo de Resposta, em minutos, de cada *ticket* que estava na tabela *TICKETS\_ESTADOS* (figuras 24 e 25). Ao clicar no ícone com duas setas viradas para fora, é aberta uma janela. Pode-se optar por “Expandir” as colunas da outra tabela ou “Agregar” os dados através de cálculo. Foi escolhida a opção “Agregar” e calculada a soma do tempo de resposta de cada *ticket*.

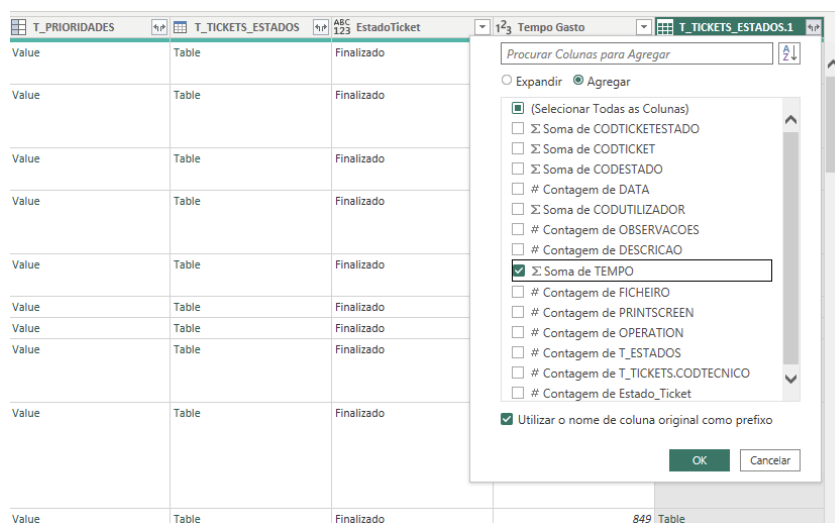
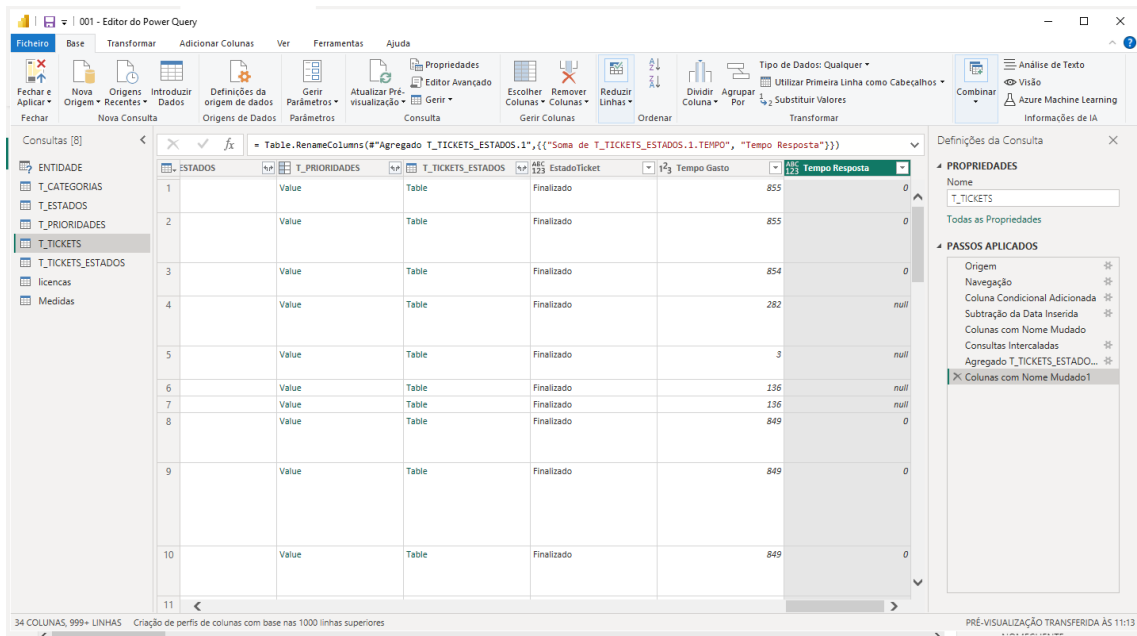


Figura 24 - Escolha da coluna a utilizar

# POLI TÉCNICO GUARDA



The screenshot shows the Power Query Editor interface. The main area displays a table with 11 rows and 7 columns. The columns are: ESTADOS, T\_PRIORIDADES, T\_TICKETS\_ESTADOS, EstadoTicket, Tempo Gasto, and Tempo Resposta. The 'Tempo Resposta' column is highlighted in green. The table data is as follows:

ESTADOS	T_PRIORIDADES	T_TICKETS_ESTADOS	EstadoTicket	Tempo Gasto	Tempo Resposta
1	Value	Table	Finalizado	855	0
2	Value	Table	Finalizado	855	0
3	Value	Table	Finalizado	854	0
4	Value	Table	Finalizado	282	null
5	Value	Table	Finalizado	3	null
6	Value	Table	Finalizado	136	null
7	Value	Table	Finalizado	136	null
8	Value	Table	Finalizado	849	0
9	Value	Table	Finalizado	849	0
10	Value	Table	Finalizado	849	0
11					

Figura 25 - Adicionada a coluna Tempo Resposta

Esses mesmos passos de intercalação foram utilizados para a obtenção dos nomes dos técnicos responsáveis por cada *ticket* gerado. A intercalação foi realizada através da coluna “CODENTIDADE” da tabela “ENTIDADE” para a tabela “TICKETS”. Na janela que apareceu depois de clicar nas duas setas para fora, igual ao passo anterior, escolheu-se a opção “Expandir” e foi selecionado o nome do técnico. Dessa forma também houve um maior detalhamento dos dados com o objetivo de facilitar a compreensão. É possível verificar as demais transformações dos dados, realizadas em outras tabelas, nos anexos.

# POLI TÉCNICO GUARDA

## 5.2.3 DADOS DOS CLIENTES DA MESTRECLIQUE

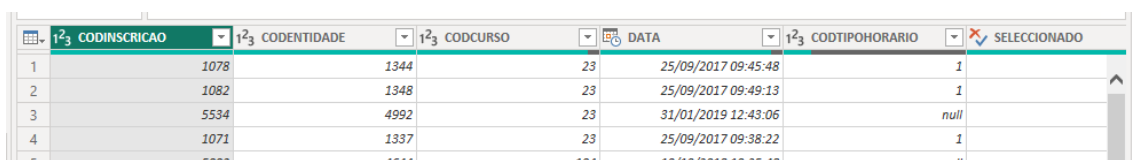
As transformações mais significativas realizadas nos dados pertencentes aos clientes da Mestreclique começaram pela tabela “INSCRICAO” (figura 26).



```
let
Origem = Sql.Database("188.93.230.33", "soprofor_nerga"),
dbo_INSCRICAO = Origem([Schema="dbo",Item="INSCRICAO"])[Data],
#"Tipo Alterado" = Table.TransformColumnTypes(dbo_INSCRICAO,{{"DATA", type date}, {"SELECCIONADO", type text}}),
#"Valor Substituído" = Table.ReplaceValue("#"Tipo Alterado", "true", "Sim", Replacer.ReplaceText, {"SELECCIONADO"}),
#"Valor Substituído1" = Table.ReplaceValue("#"Valor Substituído", "false", "Não", Replacer.ReplaceText, {"SELECCIONADO"}),
#"ENTIDADE Expandida" = Table.ExpandRecordColumn("#"Valor Substituído1", "ENTIDADE", {"CODENTIDADEEEE"}, {"ENTIDADE.CODENTIDADEEEE}),
#"Consultas Intercaladas" = Table.NestedJoin("#ENTIDADE Expandida", {"ENTIDADE.CODENTIDADEEEE"}, ENTIDADE, {"CODENTIDADE"}, {"ENTIDADE"}, JoinKind.LeftOuter),
#"ENTIDADE Expandida1" = Table.ExpandTableColumn("#Consultas Intercaladas", "ENTIDADE", {"NOME"}, {"ENTIDADE.NOME"}),
#"Tipo Alterado1" = Table.TransformColumnTypes("#ENTIDADE Expandida1", {{"WEB", type text}}),
#"Valor Substituído2" = Table.ReplaceValue("#"Tipo Alterado1", "true", "Sim", Replacer.ReplaceText, {"WEB"}),
#"Valor Substituído3" = Table.ReplaceValue("#"Valor Substituído2", "false", "Não", Replacer.ReplaceText, {"WEB"}),
#"Valor Substituído4" = Table.ReplaceValue("#"Valor Substituído3", null, "Não", Replacer.ReplaceText, {"WEB"}),
#"Consultas Intercaladas1" = Table.NestedJoin("#"Valor Substituído4", {"FORMACONHECIMENTO"}, OPCOESINSCRICAO, {"CODLISTA"}, {"OPCOESINSCRICAO", JoinKind.LeftOuter}),
#"OPCOESINSCRICAO Expandida" = Table.ExpandTableColumn("#Consultas Intercaladas1", "OPCOESINSCRICAO", {"NOME"}, {"Forma Conhecimento"}),
#"Valor Substituídos" = Table.ReplaceValue("#OPCOESINSCRICAO Expandida", null, "NÃO Informado", Replacer.ReplaceText, {"Forma Conhecimento"}),
#"Consultas Intercaladas2" = Table.NestedJoin("#"Valor Substituídos", {"INTERESSE"}, OPCOESINSCRICAO, {"CODLISTA"}, {"OPCOESINSCRICAO", JoinKind.LeftOuter}),
#"OPCOESINSCRICAO Expandida1" = Table.ExpandTableColumn("#Consultas Intercaladas2", "OPCOESINSCRICAO", {"NOME"}, {"Interesses"}),
#"Valor Substituído6" = Table.ReplaceValue("#OPCOESINSCRICAO Expandida1", null, "NÃO Informado", Replacer.ReplaceText, {"Interesses"});
```

Figura 26 - Transformações realizadas, em linguagem M, na tabela INSCRICAO

Assim que importamos as tabelas, a *Power Query* tenta identificar o tipo de cada coluna e coloca o respetivo ícone para aquele tipo de dado: se é um texto, aparece um ABC; para coluna de números inteiros, aparece o ícone 123; para coluna de números decimais, o ícone 1.2, e assim por diante. Para alterar o tipo do dado da coluna basta clicar em cima do ícone e selecionar a opção correta. Dessa forma, alterou-se o tipo da coluna “DATA”, que foi importada como texto e alterada para data, e da coluna “SELECCIONADO” que foi importada como Verdadeiro/Falso e deveria aparecer como Texto (figura 27).



	123 CODINSCRICAO	123 CODENTIDADE	123 CODCURSO	1.2 DATA	123 CODTIPOHORARIO	ABC SELECCIONADO
1	1078	1344	23	25/09/2017 09:45:48	1	
2	1082	1348	23	25/09/2017 09:49:13	1	
3	5534	4992	23	31/01/2019 12:43:06		null
4	1071	1337	23	25/09/2017 09:38:22	1	
5	5000	4244	104	10/10/2018 10:35:40		null

Figura 27 - Ícones das colunas



# POLI TÉCNICO GUARDA

Em algumas colunas foram encontrados valores como “*true*” e “*false*”. Foi resolvido que, ao invés desses valores, seriam utilizados, no *dashboard*, os valores “sim” e “não”. Esse tipo de alteração é rápido e fácil. Deve-se clicar com o botão direito do rato em cima do cabeçalho da coluna que deseja realizar a alteração. Será aberta uma janela (figura 28) e deverá clicar em “Substituir Valores”.

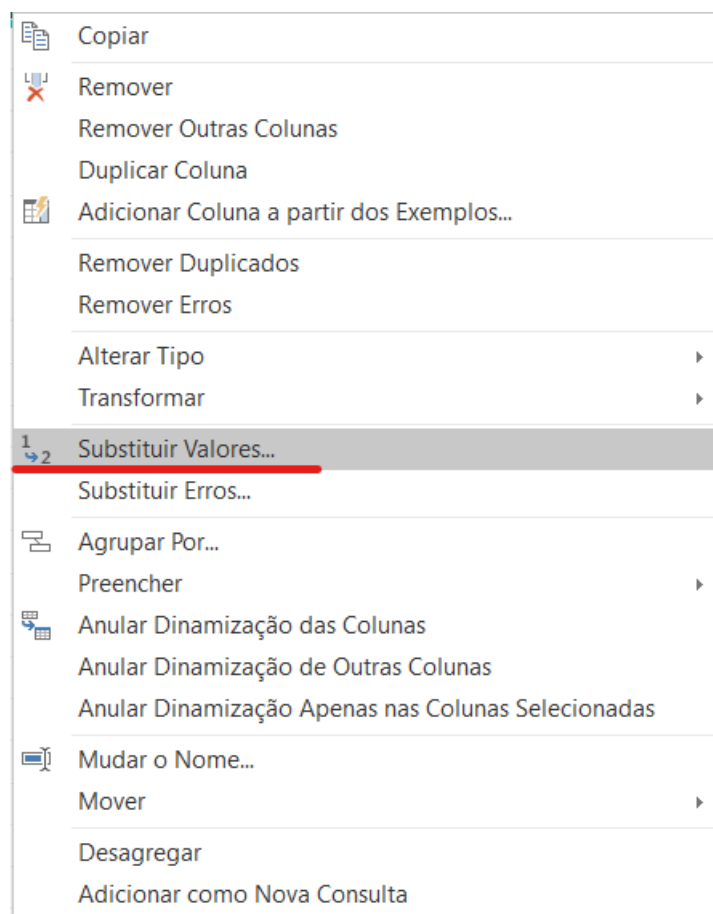


Figura 28 - Substituição de valores

Como no *dashboard* da Mestreclique, as colunas irrelevantes foram excluídas e também houve intercalação com colunas de outras tabelas.

# POLI TÉCNICO GUARDA

## 5.3 CARREGAMENTO DOS DADOS

Após o término da etapa de transformação e limpeza dos dados, foi realizado o carregamento (Load) dos dados para o *Power BI*, clicando no botão do menu “Fechar e Aplicar” (figura 29) e foi possível seguir para a confecção da criação do modelo multidimensional.

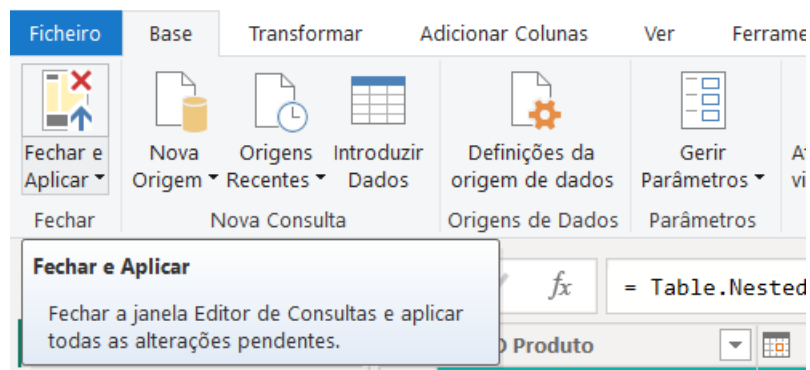


Figura 29 - Botão Fechar e Aplicar

Com isso, o processo ETL foi terminado e, em seguida, foi necessário definir as relações entre as tabelas.

# POLI TÉCNICO GUARDA

## 5.4 RELACIONAMENTOS

No *Power BI*, através da guia “Modelo”, pode-se criar e gerir os relacionamentos entre as tabelas importadas. Na figura 30 é possível observar o modelo lógico construído no projeto da Mestreclique. A tabela “Medidas” apenas ajuda na organização das medidas criadas. É possível visualizar que da tabela “*TICKETS*” para a tabela “*Calendario*” existem duas relações e uma delas se encontra tracejada. Uma relação é com a data de inserção do *ticket* e a outra com a data de finalização. A linha tracejada significa que a relação está inativa pois não é possível criar dois relacionamentos entre duas tabelas ao mesmo tempo.

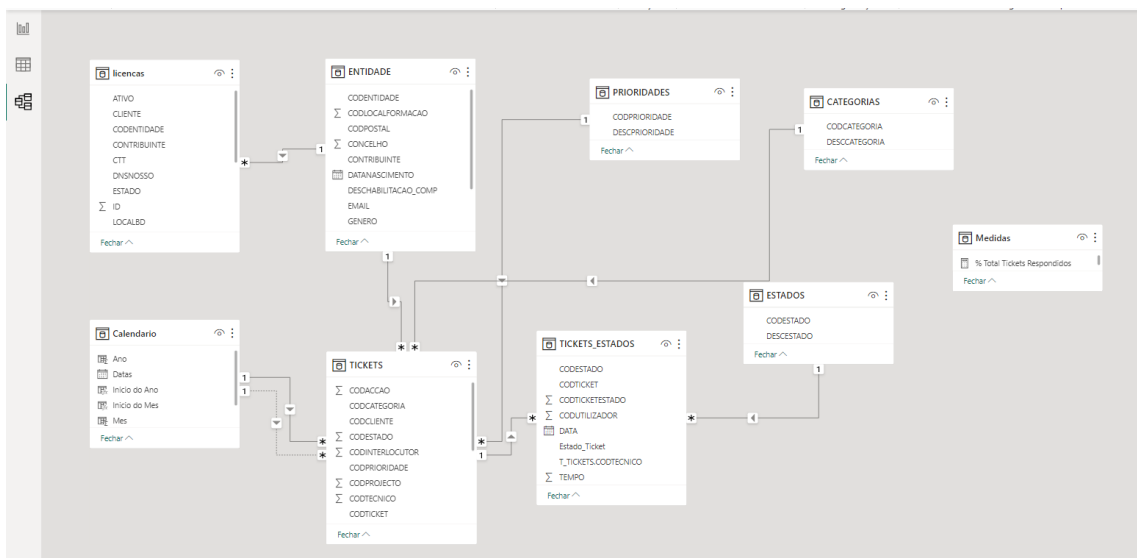


Figura 30 - Modelo Lógico | Projeto Mestreclique

# POLI TÉCNICO GUARDA

A figura 31 mostra os relacionamentos das tabelas que contém os dados dos clientes da Mestreclique.

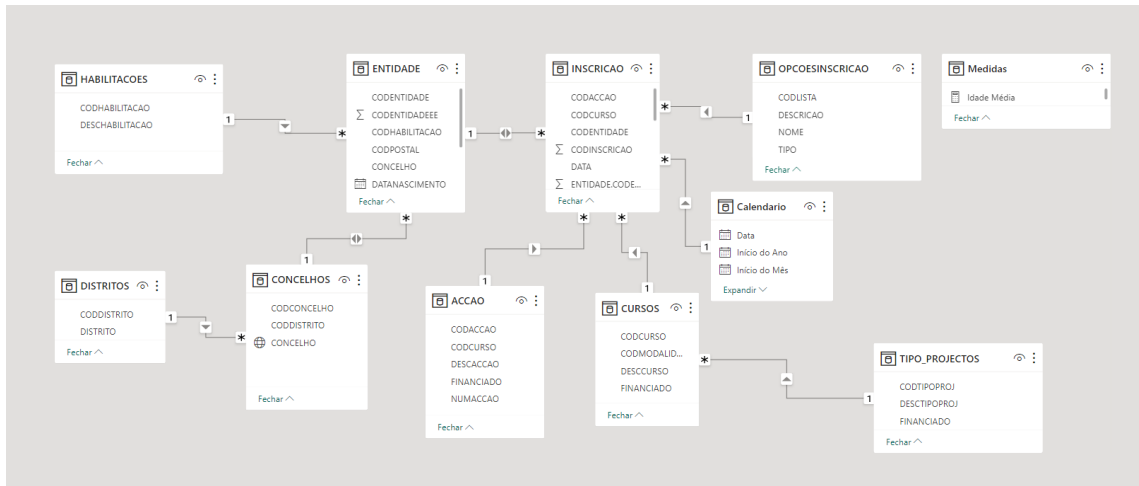


Figura 31 - Modelo Lógico | Projeto cliente Mestreclique

# POLI TÉCNICO GUARDA

## 6 O trabalho realizado: Implementação dos *Dashboards*

Após a finalização do modelo de dados foi dado início à construção dos *dashboards* que foram criados e ajustados em conformidade ao que foi solicitado pela Mestreclique, sempre havendo um feedback contínuo. Em todos os *dashboards* foi utilizada a paleta de cores com base no logotipo das empresas.

Os *dashboards* permitem obter uma visão do estado da empresa com um maior nível de detalhe facilitando o processo de tomada de decisão.

### 6.1 USO DA LINGUAGEM DAX

Os indicadores foram determinados pela Mestreclique e para refletir exatamente cada um deles foram utilizadas funções DAX com a finalidade de gerar medidas ou colunas extras nas tabelas de dados. Pode-se fazer isso de duas formas: com medidas que fornecem o resultado como um valor único ou com colunas calculadas que permitem aceder a valores de linhas específicas. As colunas calculadas, diferente das medidas, ocupam espaço de armazenamento e influenciam na performance dos processos envolvidos. Por uma questão de organização das medidas geradas, a tabela “medidas” foi criada em ambos os projetos.

#### 6.1.1 PROJETO DA MESTRECLIQUE

A primeira medida desenvolvida foi o número total de *tickets* a qual utilizou a função *COUNT* que conta a quantidade de células em uma coluna que contém números (figura 32).

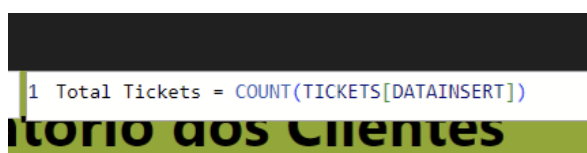


Figura 32 - Utilização da fórmula COUNT

# POLI TÉCNICO GUARDA

O início deu-se com fórmulas mais básicas, como a *SUM*, que soma os valores de uma coluna. Na exibição de Dados, do *Power BI*, foi criada uma coluna adicional, através da coluna tempo resposta (que está em minutos), na tabela *tickets*, que mostra o tempo de resposta em horas. Após, foi calculado o total do tempo de resposta, tanto em minutos quanto em horas, que o técnico leva para responder o *ticket* (figuras 33 e 34).



Figura 33 - Cálculo da medida "Total Tempo Resposta"

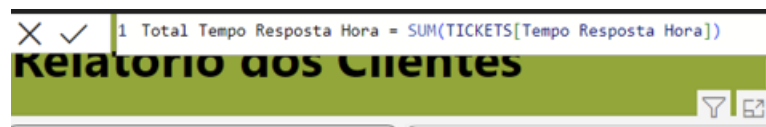


Figura 34 - Cálculo da medida "Total Tempo Resposta Hora"

Seguiu-se para o cálculo da média do tempo de resposta dos *tickets*, utilizando-se da fórmula *AVERAGE* que é responsável por retornar a média aritmética dos valores de uma coluna, como pode ser visto na figura 35.



Figura 35 - Cálculo da medida "Media Tempo Resposta"

# POLI TÉCNICO GUARDA

A fórmula *COUNTROWS* conta a quantidade de linhas em uma tabela. Já a fórmula *DISTINCTCOUNT* conta a quantidade de valores únicos em uma coluna (sem as duplicatas).

A função *CALCULATE* é responsável por calcular uma expressão ou fórmula a partir de filtros indicados. Sua sintaxe pode ser visualizada da figura 36.

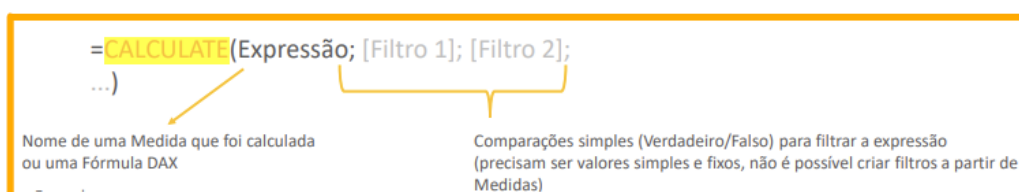


Figura 36 - Sintaxe da função CALCULATE

Para calcular a quantidade de clientes ativos da Mestreclique, foi utilizada a fórmula *COUNTROWS* juntamente com a *CALCULATE*, na qual foi contado apenas a quantidade de linhas que continha o valor “S” na coluna “ATIVO” da tabela “licenças” (figura 37). Essa tabela licenças foi importada apenas para gerar essa medida e foi desativada no *Power Query*, motivo pelo qual ela não aparece no modelo lógico.

```
1 TotalClientesAtivos = CALCULATE(
2   COUNTROWS(licenças);
3   licenças[ATIVO] = "S")
```

Figura 37 - Fórmula da medida "TotalClientesAtivos"

# POLI TÉCNICO GUARDA

Também foi utilizada a fórmula *CALCULATE* para calcular o total de *tickets* respondidos. Esta fórmula foi usada juntamente com a fórmula *DISTINCTCOUNT* (figura 38). Ou seja, dentre todas as linhas de valores únicos na coluna com os códigos dos *tickets* (“*CODTICKET*”), foi calculada a quantidade dos *tickets* que apresentava a palavra “Respondido”, na coluna “*Estado\_Ticket*” da tabela “*TICKETS\_ESTADOS*”.

```
1 Tickets Respondidos - Estados = CALCULATE(  
2 |   DISTINCTCOUNT(TICKETS_ESTADOS[CODTICKET]);  
3 |   TICKETS_ESTADOS[Estado_Ticket] = "Respondido")
```

Figura 38 - Cálculo da medida "Tickets Respondidos - Estados"

Para calcular o percentual de *tickets* respondidos em relação ao total de *tickets*, primeiro foi criada uma variável chamada “*total\_tickets\_fixo*”, conforme pode ser visualizado na figura 39, para auxiliar o cálculo intermediário da medida. Para a criação dessa variável foi utilizada a fórmula *CALCULATE* que pede, no mínimo, dois argumentos. No primeiro argumento foi contado os *tickets* de forma distinta, utilizando a função *DISTINCTCOUNT*. No segundo argumento foi adicionada a função *ALL* que é responsável por cancelar qualquer tipo de filtro que exista. Por exemplo, a função *ALL* desconsidera qualquer contexto de filtro quando utilizada dentro de uma matriz e retorna o valor total (absoluto) para todas as categorias. Após a criação da variável, foi utilizado o *RETURN* para validar a variável e retornar a medida calculada. Em seguida, foi dividida a medida “*Tickets Respondidos – Estados*”, criada acima, pelo *total\_tickets\_fixo*. Dessa forma, foi obtido o percentual dos *tickets* respondidos em relação ao todo.

```
1 % Total Tickets Respondidos =  
2 var total_tickets_fixo = CALCULATE(  
3 |   DISTINCTCOUNT(TICKETS_ESTADOS[CODTICKET]);  
4 |   ALL(TICKETS_ESTADOS)  
5 | )  
6 return  
7 DIVIDE(  
8 |   [Tickets Respondidos - Estados];  
9 |   total_tickets_fixo  
10 | )
```

Figura 39 - Fórmula para criar o Percentual de Tickets Respondidos



# POLI TÉCNICO GUARDA

E por último, foi criada a medida para calcular a quantidade de *tickets* abertos. A Mestreclique contabiliza como *tickets* ainda não finalizados e que possuem alguma pendência, os que apresentam em seu estado os resultados: “Aberto”, “Comunicação Interna”, “Informação Recebida” e “Recusado”. Para a fórmula, foi utilizada a função *CALCULATE* com o *IN* nos filtros para considerar qualquer valor correspondente ao *ticket* que ainda apresenta alguma pendência (figura 40).

```
1 Tickets abertos = CALCULATE([Total Tickets];  
2 | TICKETS; TICKETS[EstadoTicket] in {"Aberto"; "Comunicação Interna"; "Informação Recebida"; "Recusado"})
```

Figura 40 - Cálculo da quantidade de *tickets* abertos

## 6.1.2 PROJETO DOS CLIENTES

O primeiro passo no projeto dos clientes, foi a criação de uma coluna calculada, utilizando a linguagem DAX. No ícone de exibição “Dados”, que é onde pode-se visualizar todas as tabelas que são usadas, foi escolhida a opção “Nova coluna”, localizada na aba “Ferramentas de tabela” (figura 41).

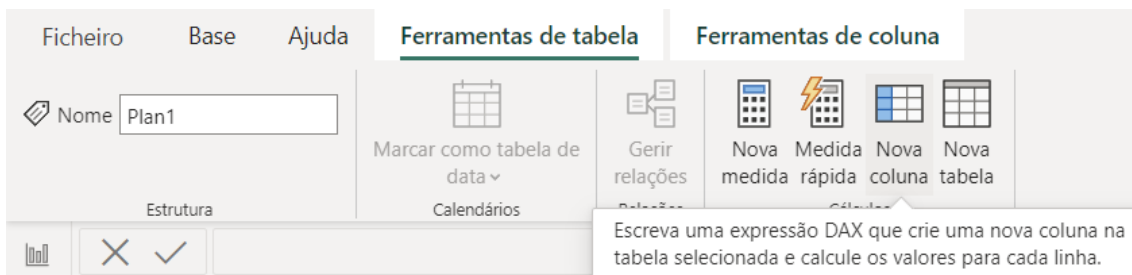


Figura 41 - Criação de nova coluna calculada

# POLI TÉCNICO GUARDA

Ao clicar em “Nova coluna”, automaticamente é aberta a barra de fórmulas, inicialmente com o texto “Coluna =”. É nessa barra de fórmulas que é criada a fórmula para preencher os valores da coluna calculada (figura 42).



Figura 42 - Barra de fórmulas para criação da coluna calculada

A princípio foi utilizada a fórmula que se encontra na Figura 43. A função *YEARFRAC* é responsável por calcular a fração do ano representada pelo número de dias inteiros existentes entre duas datas. Neste caso, entre a data de nascimento da entidade e a data da realização da inscrição, com a finalidade de calcular a idade da pessoa no ato da inscrição. Para que o número apareça inteiro, foi utilizada a função *INT* que arredonda para o número inteiro igual ou inferior mais próximo.

The image shows a Power BI table with a DAX formula in the formula bar. The formula is: `1 Idade na Inscricao = INT(YEARFRAC(INSCRICAO[ENTIDADE.DATANASCIMENTO]; INSCRICAO[DATA]))`. The table has the following columns: *ENTIDADE.DATANASCIMENTO*, *Idade Atual*, *FORMACONHECIMENTO\_NOME*, *OPCOESINSCRICAO.NOME*, *Tipos de Interesse*, and *Idade na Inscricao*.

ENTIDADE.DATANASCIMENTO	Idade Atual	FORMACONHECIMENTO_NOME	OPCOESINSCRICAO.NOME	Tipos de Interesse	Idade na Inscricao
28/07/1950 00:00:00	72	Site da Castelform	Aprofundar conhecimentos / competências	Aprofundar conhecimentos/competências	69
28/07/1950 00:00:00	72	Site da Castelform	Aprofundar conhecimentos / competências	Aprofundar conhecimentos/competências	69
28/07/1950 00:00:00	72	Site da Castelform	Aprofundar conhecimentos / competências	Aprofundar conhecimentos/competências	69
18/08/1955 00:00:00	67	Site da Castelform	Aprofundar conhecimentos / competências	Aprofundar conhecimentos/competências	64
02/02/1958 00:00:00	65	Site da Castelform	Aprofundar conhecimentos / competências	Aprofundar conhecimentos/competências	61

Figura 43 - Fórmula DAX para coluna calculada

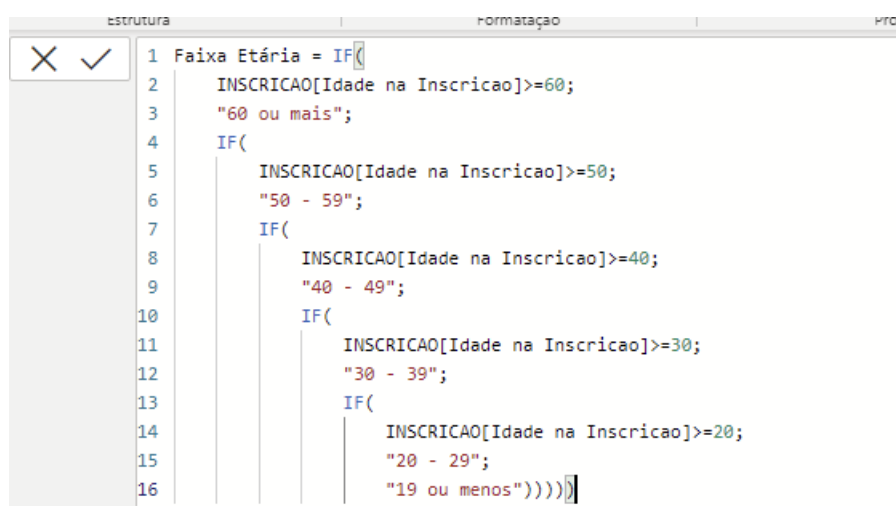
# POLI TÉCNICO GUARDA

Posteriormente, a fórmula foi alterada pois, nas entidades que não apresentavam a data de nascimento, o *Power Query* entendia que era a primeira data disponível no sistema e atribuía-lhes o ano de 1900. Isso fez com que algumas entidades apresentassem quase 200 anos. A fórmula acima foi incluída na nova fórmula. Foi utilizado o “*IF*” para que nas situações que a data de nascimento estivesse em branco, consequentemente a idade também apareceria em branco, caso contrário, seria utilizada a fórmula que foi escrita inicialmente. Para que esta situação não mais ocorresse, a fórmula da coluna calculada foi alterada para:

Idade na Inscrição = IF(

```
ISBLANK(INSCRICAO[ENTIDADE.DATANASCIMENTO]); BLANK();  
INT(YEARFRAC(INSCRICAO[ENTIDADE.DATANASCIMENTO];  
INSCRICAO[DATA])))
```

Logo após a criação da coluna calculada referente à idade no ato da inscrição, foi feita outra coluna calculada para dividir as idades em faixa etária, com seis classificações (figura 44).

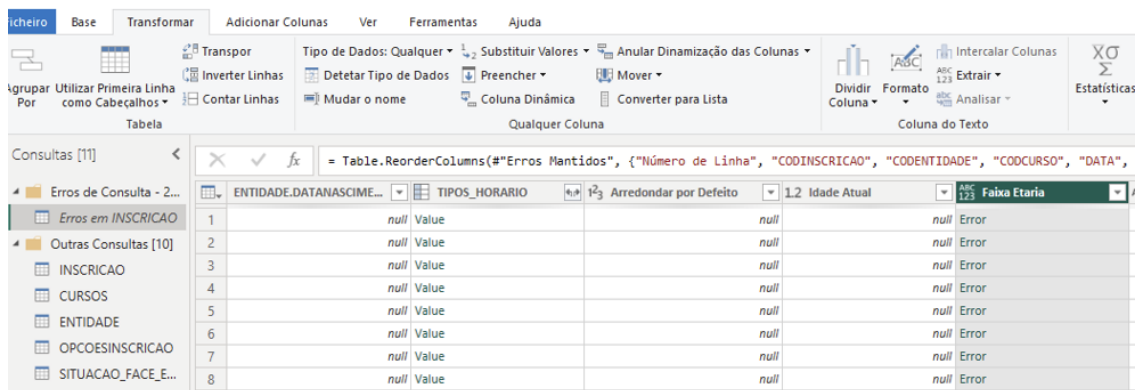


```
1 Faixa Etária = IF(  
2   INSCRICAO[Idade na Inscrição]>=60;  
3   "60 ou mais";  
4   IF(  
5     INSCRICAO[Idade na Inscrição]>=50;  
6     "50 - 59";  
7     IF(  
8       INSCRICAO[Idade na Inscrição]>=40;  
9       "40 - 49";  
10    IF(  
11      INSCRICAO[Idade na Inscrição]>=30;  
12      "30 - 39";  
13      IF(  
14        INSCRICAO[Idade na Inscrição]>=20;  
15        "20 - 29";  
16        "19 ou menos")))))]
```

Figura 44 - Classificações da faixa etária

# POLI TÉCNICO GUARDA

Como o campo data de nascimento não é obrigatório, algumas entidades, na tabela INSCRICAO, estavam com esse campo vazio. Isso fez gerar um erro na coluna da Faixa Etária (figura 45).



The screenshot shows the Power Query interface. The ribbon includes 'Base', 'Transformar', 'Adicionar Colunas', 'Ver', 'Ferramentas', and 'Ajuda'. The 'Transformar' ribbon is active, showing options like 'Transpor', 'Inverter Linhas', 'Contar Linhas', 'Tipo de Dados', 'Substituir Valores', 'Anular Dinamização das Colunas', 'Mover', 'Dividir Coluna', 'Formato', 'Intercalar Colunas', 'Extrair', and 'Analisar'. The formula bar shows a query named 'Table.ReorderColumns'. The table below has columns: 'ENTIDADE.DATANASCIME...', 'TIPOS\_HORARIO', 'Arredondar por Defeito', 'Idade Atual', and 'Faixa Etária'. The 'Faixa Etária' column contains 'Error' for rows 1 through 8, while other columns contain 'null' or 'Value'.

	ENTIDADE.DATANASCIME...	TIPOS_HORARIO	Arredondar por Defeito	Idade Atual	Faixa Etária
1	null	Value		null	Error
2	null	Value		null	Error
3	null	Value		null	Error
4	null	Value		null	Error
5	null	Value		null	Error
6	null	Value		null	Error
7	null	Value		null	Error
8	null	Value		null	Error

Figura 45 - Erro na coluna da faixa etária

Para a correção desse erro foi necessário abrir o *Power Query*, clicar com o botão direito do rato, em cima da coluna de faixa etária, e escolher a opção “Substituir Erros”, conforme visualizado na figura 46.

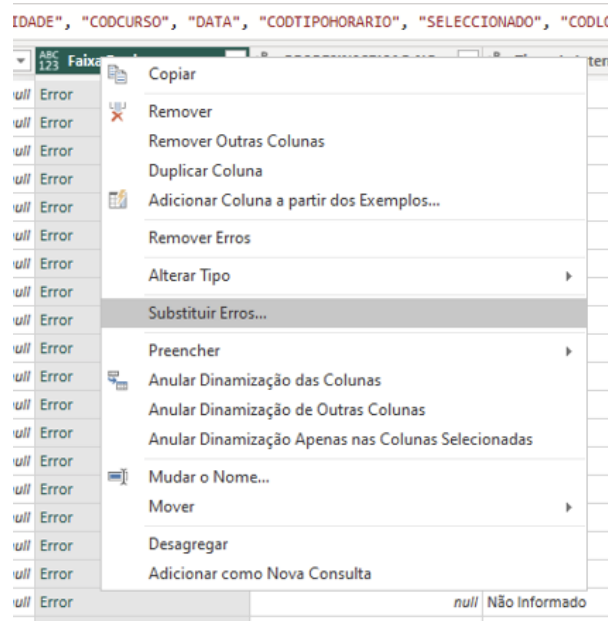


Figura 46 - Correção de erro na coluna

# POLI TÉCNICO GUARDA

Logo após o clique na opção “Substituir Erros”, é aberta uma janela com um campo para ser introduzido qualquer valor. Nesse caso, foi introduzido o zero (figura 47).

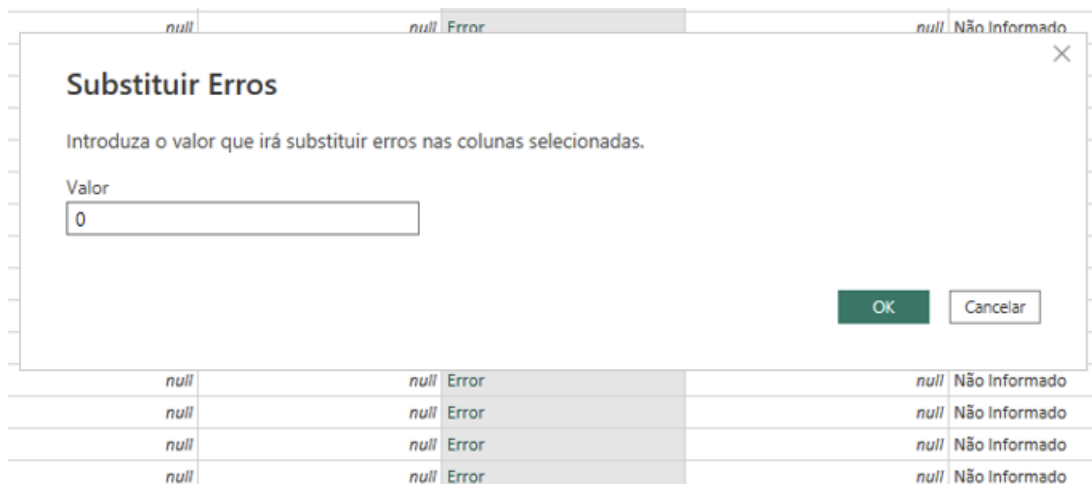


Figura 47 - Janela para introduzir o valor na correção do erro

Da mesma forma do projeto anterior, as medidas no projeto dos clientes da Mestreclique foram organizadas em uma tabela (figura 48).

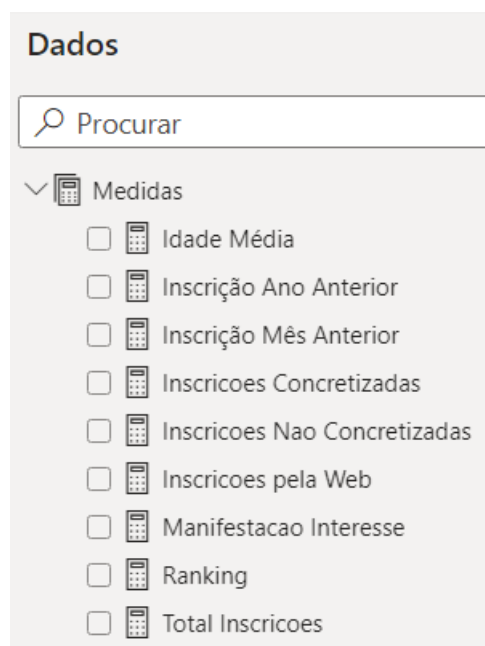


Figura 48 - Tabela com as medidas - projeto dos clientes

# POLI TÉCNICO GUARDA

A primeira medida construída foi a Idade Média das Entidades. Junto com a função *AVERAGE* foi utilizada a função *ROUND* que é responsável por arredondar o número e pede dois parâmetros: o número que desejamos arredondar e o número de dígitos que queremos. O valor zero, no segundo parâmetro, arredonda para o número inteiro mais próximo (figura 49).

```
1 Idade Média = ROUND(  
2 |   AVERAGE(INSCRICAO[Idade na Inscricao]);  
3 |   0  
4 | )  
5
```

Inscrições por Cursos

Figura 49 - Uso da função *AVERAGE* com *ROUND*

Uma outra medida calculada foi o número de inscrições no ano anterior para poder comparar com o número do ano atual. Novamente foi utilizada a função *CALCULATE*, que é uma das mais usadas, juntamente com a *DATEADD*, que é uma das fórmulas de inteligência de tempo. Ela permite criar comparações e análises inteligentes como o acumulado do ano, acumulado dos últimos dez dias, comparação com o mesmo período do ano anterior e diversas outras. A *DATEADD* pede três argumentos: a data, o intervalo de tempo e a quantidade. Quando a quantidade esta com sinal negativo significa que a medida é referente a um período anterior (figura 50).

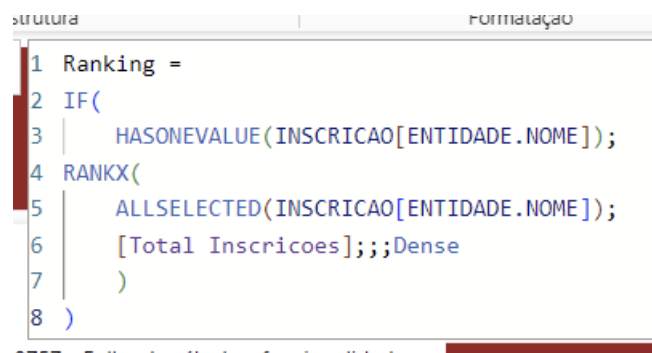
```
1 Inscrição Ano Anterior = CALCULATE([Total Inscricoes]; DATEADD(Calendario[Início do Mês]; -1; YEAR))
```

13	616	20	Tudo	Tudo
----	-----	----	------	------

Figura 50 - Cálculo do número de inscrições do ano anterior

# POLI TÉCNICO GUARDA

Por último, foi calculado o *Ranking* das Entidades Empregadoras que mais realizavam inscrições nas empresas clientes da Mestreclique. Para este cálculo, foi utilizada a função RANKX (figura 51), que permite criar uma classificação e que pede dois argumentos. Para que as entidades não sejam comparadas com elas mesmas e todas fiquem na primeira posição do *ranking*, foi necessário utilizar a função *ALLSELECTED* para que a classificação fosse feita considerando todas as entidades. Para chegar ao “dense”, na fórmula, foi necessário passar por dois argumentos que são opcionais e, por isso, foram omitidos pois não cabiam na avaliação pretendida. O motivo de ter colocado a palavra “dense” foi para que o próximo valor de classificação, após um empate, apresentasse o valor seguinte.



```
1 Ranking =
2 IF(
3     HASONEVALUE(INSCRICAO[ENTIDADE.NOME]);
4 RANKX(
5     ALLSELECTED(INSCRICAO[ENTIDADE.NOME]);
6     [Total Inscricoes];;;Dense
7 )
8 )
```

Figura 51 - Cálculo do *ranking*

# POLI TÉCNICO GUARDA

Para evitar que a linha da matriz, referente ao Total de Inscrições, mostrasse o número um na coluna do *Ranking*, foi feito o uso da função *HASONEVALUE* (figura 52). Com a ausência desta função, o *Power BI* enxergaria um único valor de entidade e não todas e mostraria o número um na linha do total. A função *HASONEVALUE* identifica onde tem um único valor de entidade na matriz. Neste caso, tem-se apenas um único valor de entidade em cada uma das linhas. Para isso, também foi usado o IF para calcular o *ranking* apenas se existir um valor de entidade. Caso contrário, que é o que acontece na linha do total, pois apresenta todas as entidades, não se retorna nada.

Entidade Empregadora	Inscrições	Ranking
Pessoaal Lda		
Município de Vila do Conde	87	6
Associação Social Estrelas de Silveirinhos	83	7
Lar Familiar da Tranquilidade	79	8
Itau-Instituto Tecnico de Alimentação Humana S.A	78	9
Município de Santo Tirso	78	9
Agrupamento de Escolas Tomaz Pelayo	76	10
Guarda Nacional Republicana	75	11
<b>Total</b>	<b>6.380</b>	

Figura 52 - Visualização do *ranking* em uma matriz



# POLI TÉCNICO GUARDA

## 7 O trabalho realizado: Comunicação e Visualização dos *Dashboards*

Foram criadas três páginas de *dashboards* em ambos os projetos. Para a empresa Mestreclique, a primeira página é referente ao quadro de funcionários. A segunda é relativa aos clientes que a empresa atende e a terceira e última página demonstra a previsão e anomalia referentes à quantidade de *tickets* abertos na empresa. No projeto dos clientes da Mestreclique, as três páginas referem-se exclusivamente à componente “inscrição dos formandos”.

### 7.1 DASHBOARD DA MESTRECLIQUE

A construção da página um do *dashboard* da Mestreclique foi relacionada aos técnicos (figura 53). Foram utilizados um funil, um gráfico de barras, um gráfico de colunas, um gráfico de linha e um gráfico de área. Na parte superior foi inserido dois cartões, um com o total de *tickets* abertos e outro com o total de *tickets*, e uma segmentação de dados para filtrar as informações por data. Essa parte superior também pode ser encontrada na página 2.



Figura 53 - Página 1 do *dashboard* da Mestreclique

# POLI TÉCNICO GUARDA

Para começar a construção do *dashboard* basta clicar em qualquer elemento do painel visualizações para ele aparecer em branco no relatório (figura 54).

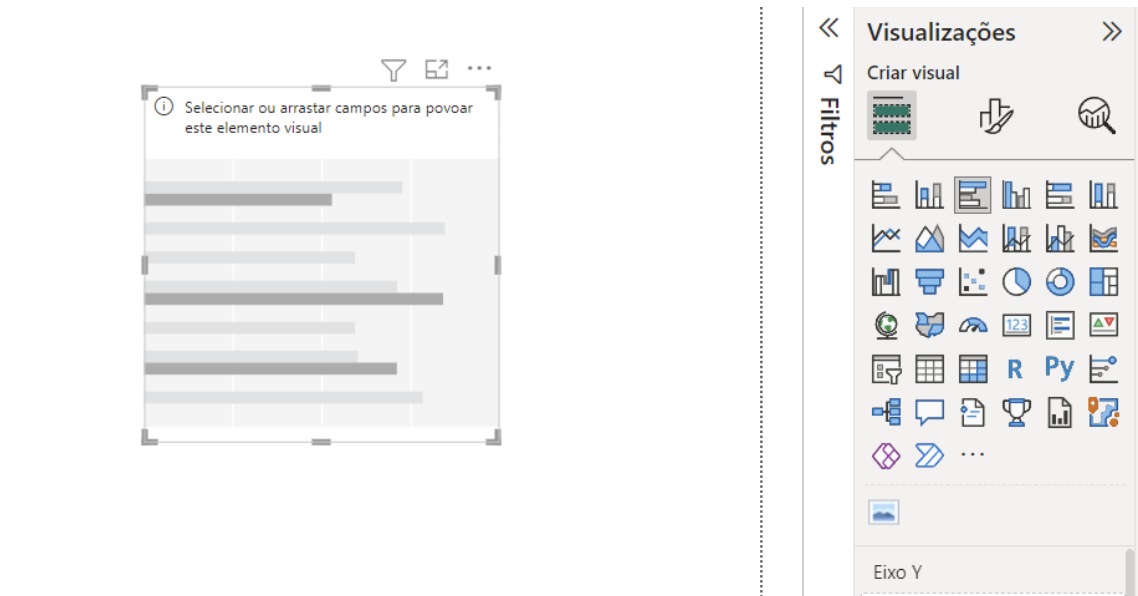


Figura 54 - Início da construção do relatório

Sempre que qualquer objeto é incluído no relatório, aparecem campos no lado direito, abaixo do menu Visualizações, que devem ser configurados para construir o visual desejado. Na figura 55 pode-se visualizar a configuração do cartão que apresenta o indicador de clientes que estão com o contrato ativo na empresa.

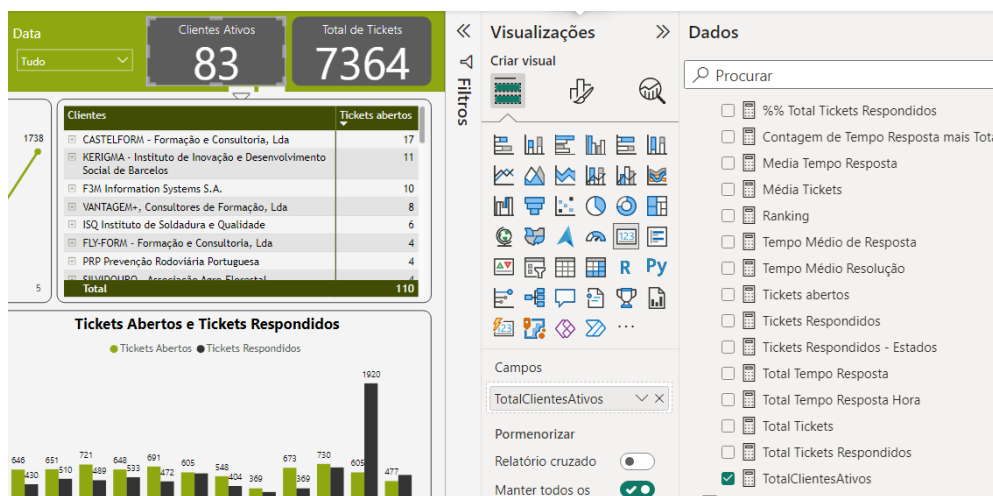


Figura 55 - Configuração do cartão

# POLI TÉCNICO GUARDA

Além do visual criado, é possível criar *tooltips* que nada mais são do que dicas de ferramenta de relatório visualmente ricas que aparecem quando o rato é posicionado sobre os elementos visuais. Essas dicas de ferramentas personalizadas podem incluir elementos visuais, imagens e qualquer outra coleção de itens. É uma ferramenta que o usuário tem para poder detalhar ainda mais algumas informações gráficas. Para isso, é preciso criar uma página de relatório e na opção “Definições de tela” escolher o tipo “Descrição” (figura 56). Esse tipo já fornece o tamanho de tela ideal para construir a dica de ferramenta.



Figura 56 - Construção da dica de ferramenta

Com essa alteração realizada pode-se partir para a criação das informações que serão visualizadas ao passar o rato sobre algum visual.

# POLI TÉCNICO GUARDA

O funil utilizado apresenta a quantidade de *tickets* abertos que cada técnico possui. Isto facilita na hora da distribuição dos *tickets* pois dá para saber, de uma forma mais rápida, o técnico que possui mais *tickets* em andamento. Ao passar o rato por cima da barra é aberta uma dica de ferramenta que mostra as categorias desses *tickets* em andamento de cada técnico (figura 57).

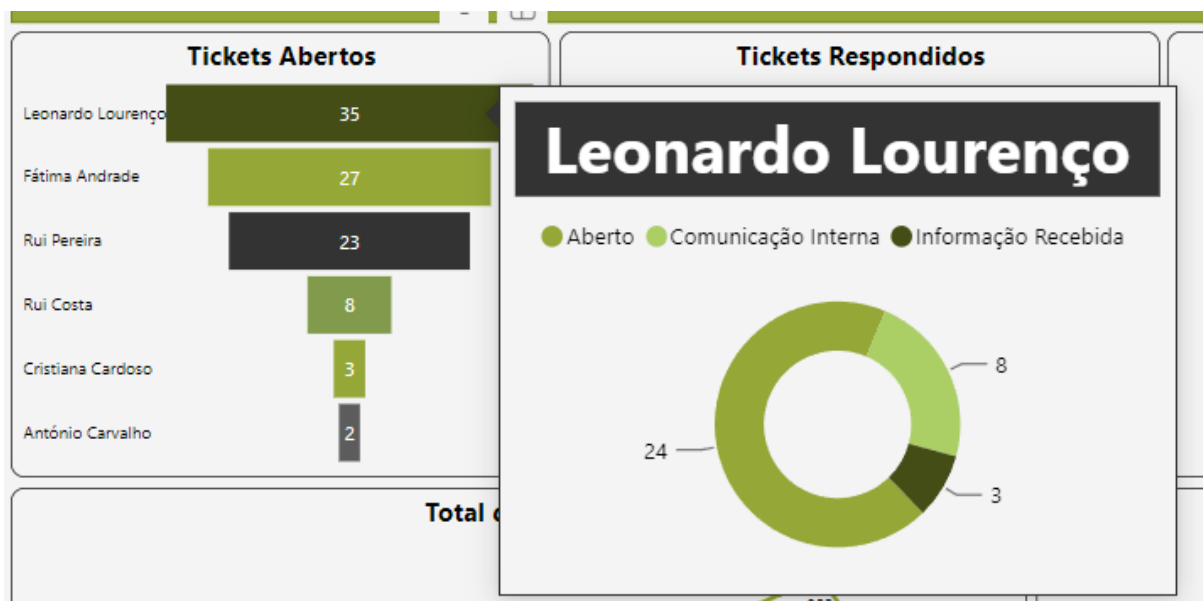


Figura 57 - Dica de ferramenta dos *tickets* em andamento

# POLI TÉCNICO GUARDA

O objetivo do gráfico de barras foi visualizar o percentual de *tickets* respondidos por cada técnico. Ao posicionar o rato em cada barra é aberta a segunda dica de ferramenta que mostra o total de *tickets* que o técnico respondeu e, novamente, o percentual desses *tickets* (figura 58).



Figura 58 - Dica de ferramenta dos *tickets* respondidos

No gráfico de colunas é possível observar o tempo médio de resposta, em minutos, ao *ticket* por parte de cada técnico (figura 59).

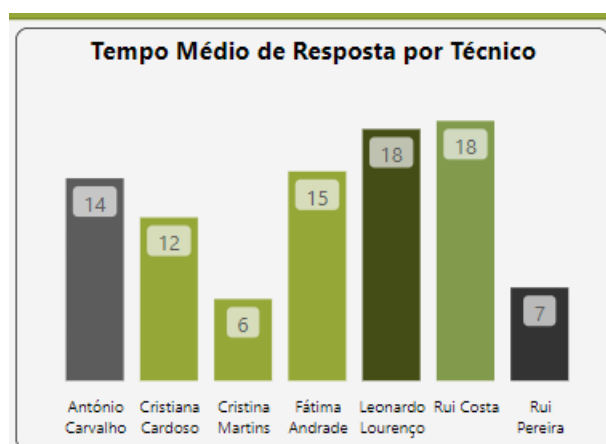


Figura 59 - Tempo médio de resposta ao *ticket*

# POLI TÉCNICO GUARDA

No gráfico de linha é possível visualizar o total de *tickets* em cada mês do ano (figura 60). Nem no gráfico de coluna acima e nem no gráfico de linha foi utilizada dica de ferramenta.

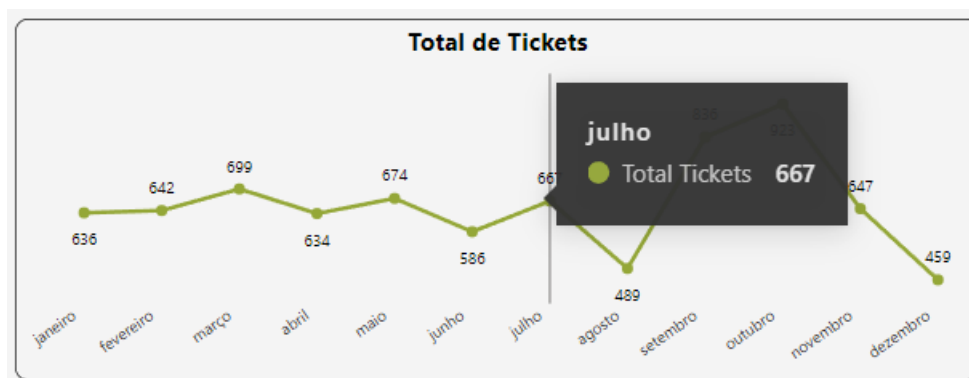


Figura 60 - Total de *tickets* em cada mês

O gráfico de áreas tem por objetivo mostrar o tempo médio, em horas, de resposta aos *tickets* no decorrer do ano. Este gráfico também possui uma dica de ferramenta que demonstra as cinco categorias de *tickets* que mais demandaram tempo e o tempo médio de cada categoria (figura 61).

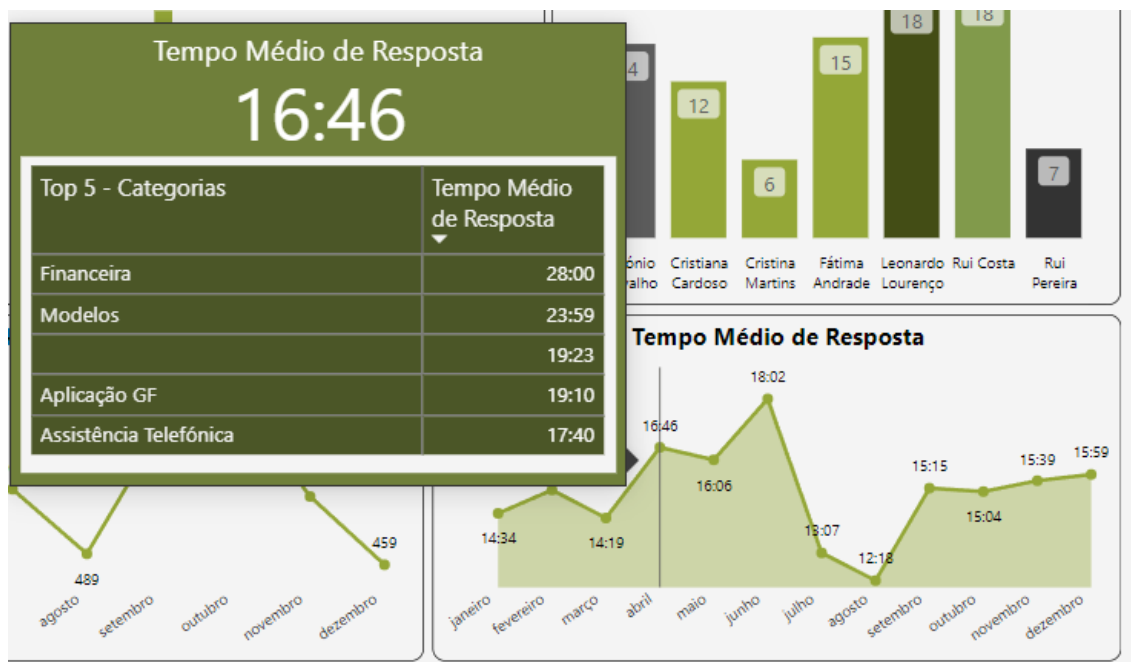


Figura 61 - Tempo médio de resposta aos *tickets* no decorrer do ano

# POLI TÉCNICO GUARDA

A página 2 foi relacionada com os clientes da Mestreclique (figura 62) e, além da parte superior que também se encontra na página 1, foi utilizado um treemap (gráfico de árvore), um gráfico de linha, uma matriz, um gráfico de barras e um gráfico de colunas agrupadas.



Figura 62 - Página 2 do dashboard da Mestreclique

O treemap tem como objetivo mostrar os cinco clientes que mais possuem tickets com a Mestreclique. Ao posicionar o rato em cada cliente é aberta mais uma barra de ferramenta onde é possível visualizar o nome do cliente, o total de tickets que ele possui e, dentre esses, a quantidade de tickets que ainda não foram finalizados (figura 63).

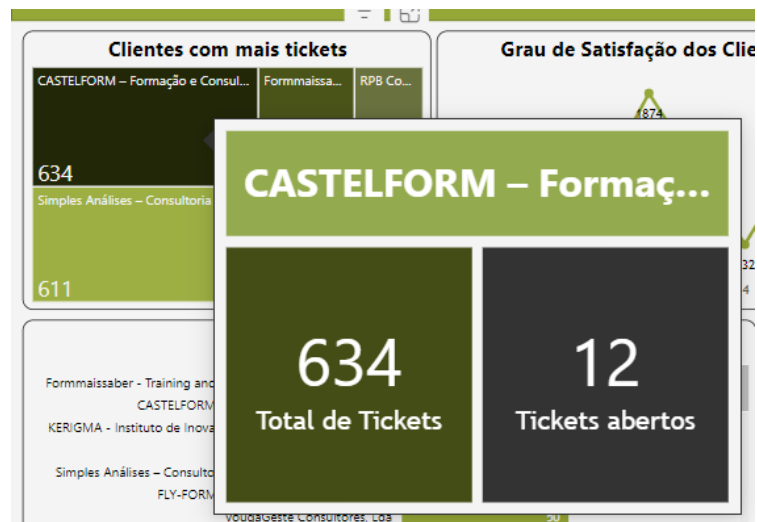


Figura 63 - Treemap dos clientes com mais tickets

# POLI TÉCNICO GUARDA

A figura 64 mostra o grau de satisfação dos clientes, que vai de um a cinco. Ao posicionar o rato em cada grau de satisfação é possível visualizar, em ordem decrescente, a quantidade de classificação naquele grau de cada cliente.

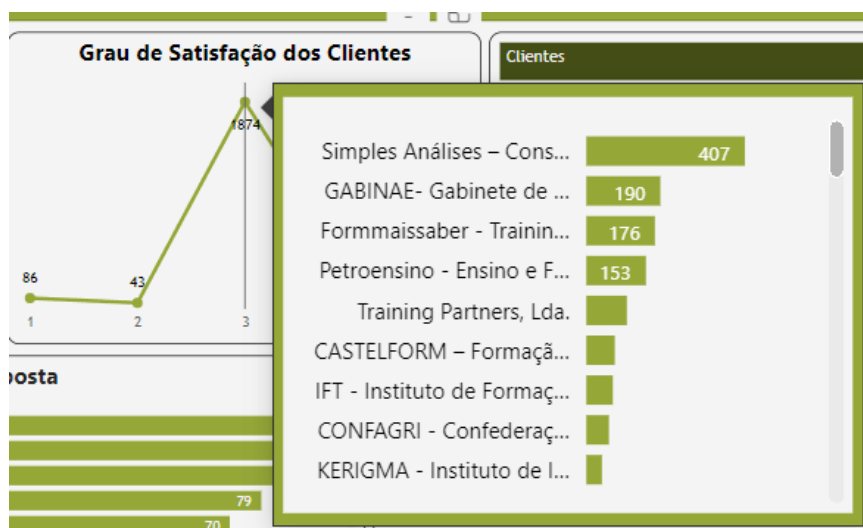


Figura 64 - Grau de Satisfação dos Clientes

A matriz utilizada apresenta os clientes que possuem *tickets* abertos e para saber a categoria desses *tickets* é necessário clicar no símbolo “+” que fica ao lado do nome do cliente que as categorias são visualizadas (figura 65).

Clientes	Tickets abertos
CASTELFORM - Formação e Consultoria, Lda	12
Aberto	9
Comunicação Interna	3
Formmaissaber - Training and Development, Unipessoal, Lda	11
Aberto	9
Recusado	2
F3M Information Systems S.A.	8
KERIGMA - Instituto de Inovação e	8
<b>Total</b>	<b>103</b>

Figura 65 - Clientes com *tickets* em andamento



# POLI TÉCNICO GUARDA

O gráfico de barras da página 2 é a última visualização desse projeto que possui dica de ferramenta personalizada. Este gráfico mostra o tempo total de resposta, em horas, dos *tickets* de cada cliente e pela dica de ferramenta pode-se visualizar o nome do cliente e o tempo gasto em horas e em minutos (figura 66).

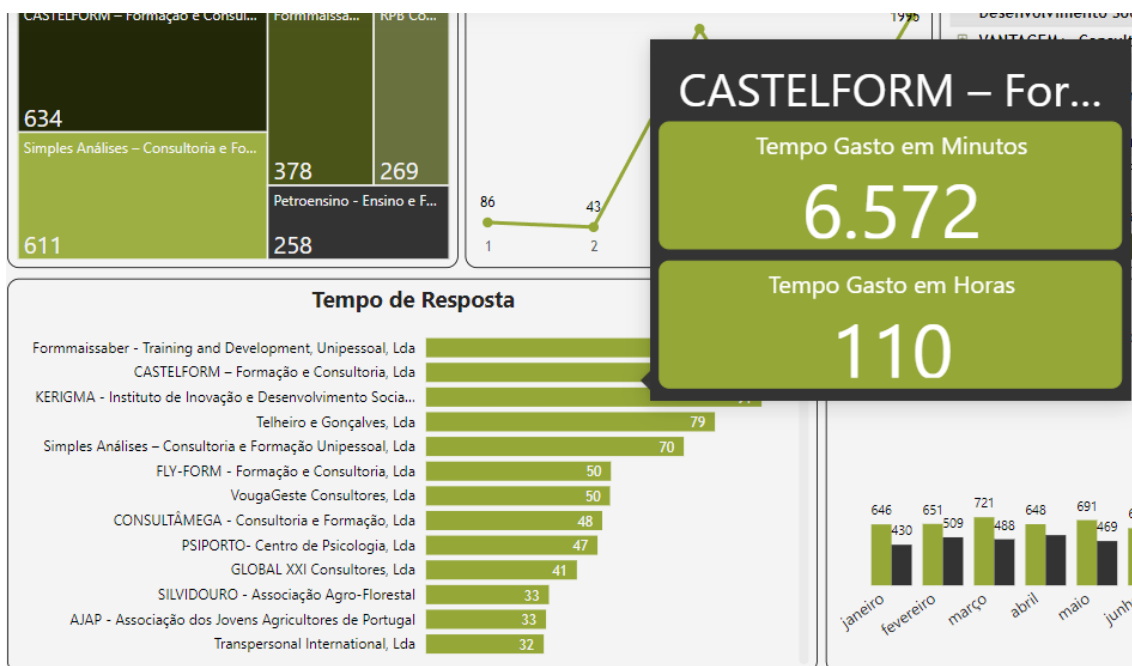


Figura 66 - Tempo de resposta a cada cliente

# POLI TÉCNICO GUARDA

A página 2 termina com o gráfico de colunas agrupadas que mostra a quantidade de *tickets* abertos e *tickets* respondidos ao ano (figura 67).

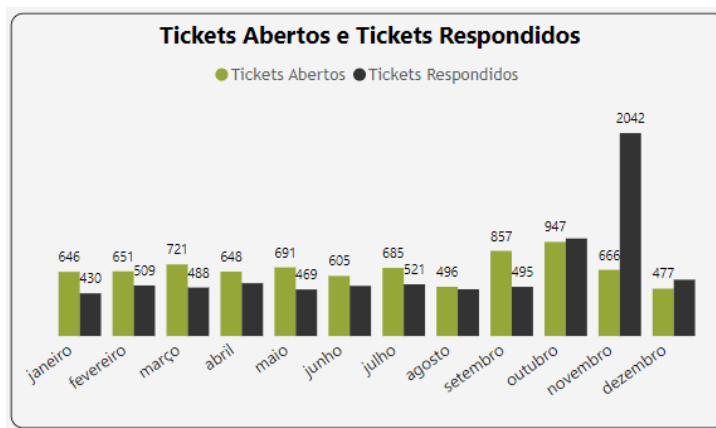


Figura 67 - Comparação entre *tickets* abertos e respondidos

A página 3 possui dois gráficos de linhas. O primeiro faz uma previsão da quantidade de *tickets* para os próximos dois anos e o segundo faz uma análise de anomalia da quantidade de *tickets* (figura 68).

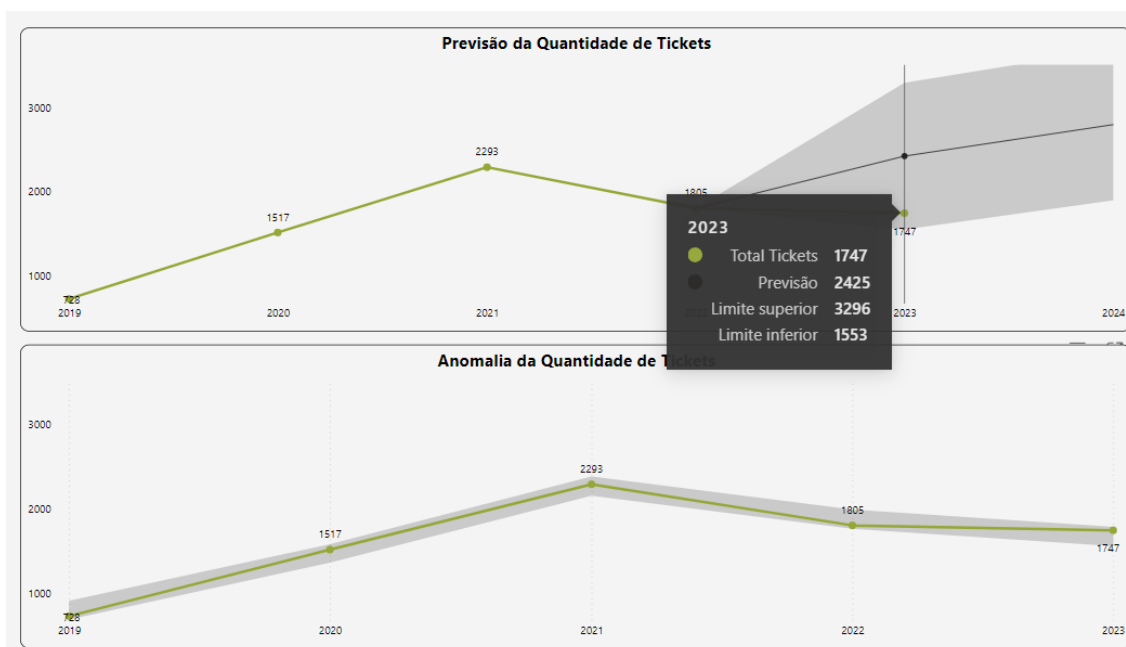


Figura 68 - Previsão e anomalia da quantidade de *tickets*

# POLI TÉCNICO GUARDA

Para ambas as opções, é necessário usar dados de tempo e para construir a previsão ou anomalia basta clicar no ícone de análises no menu Visualizações (figura 69).

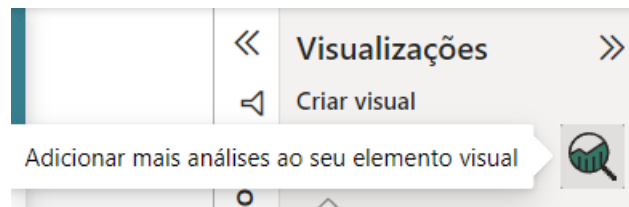


Figura 69 - Ícone de análises

As alterações mais importantes na previsão são relacionadas com o comprimento da previsão e com seu intervalo de confiança (figura 70). Nesse projeto foi utilizado dois anos como previsão e o intervalo de confiança foi deixado em 95%, o que indica que a margem de erro foi deixada em 5%.

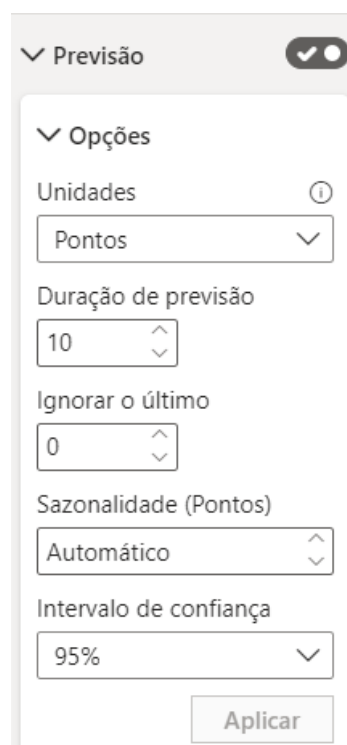


Figura 70 - Configurações da previsão

# POLI TÉCNICO GUARDA

Na anomalia, quanto maior o número da confidencialidade, ou sensibilidade, mais pontos de dados aparecem. Quanto menor, mais seletivo o gráfico fica. Neste projeto deixamos a confidencialidade em 70% (figura 71).

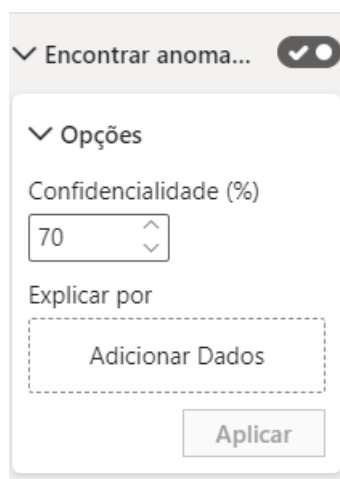


Figura 71 - Configuração da anomalia

## 7.2 DASHBOARD DO CLIENTE

Tal como no *dashboard* da Mestreclique, o *dashboard* dos clientes também apresenta três páginas e também faz uso de *tooltips*.

Na página 1 (figura 72) foram utilizados dois cartões na parte superior e que indicam o total de inscrições e a média das idades dos inscritos. Logo ao lado existe a possibilidade de filtrar os dados de três formas: selecionando se a inscrição foi concretizada ou não, se a formação foi financiada ou não e, ainda, selecionando o tempo pretendido para a devida análise. Nas Inscrições por Localização foi dada a preferência de utilização de um mapa interativo. O mapa pode ser aproximado ou afastado com o uso do rato e é possível clicar no distrito e visualizar a quantidade de inscrições por concelho. O KPI de Inscrições Mensais mostra a quantidade de inscrições mensais do mês atual e abaixo mostra a quantidade do mês anterior. Nas configurações foi escolhida a cor vermelha quando o eixo de tendência estivesse negativo, azul para neutro e verde para positivo. Também foi utilizado um gráfico de área que contém a

# POLI TÉCNICO GUARDA

quantidade de inscrições por faixa etária e uma matriz que apresenta a quantidade de inscrições das entidades empregadoras juntamente com o *ranking*, que já foi explicado anteriormente neste relatório.

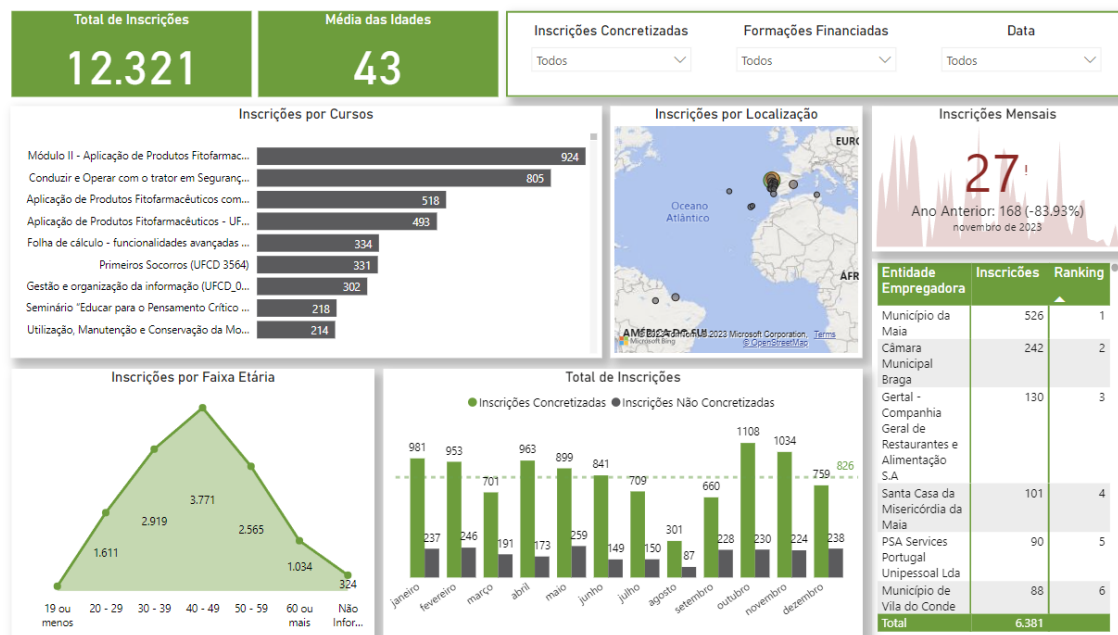


Figura 72 - Página 1 do *dashboard* dos clientes

# POLI TÉCNICO GUARDA

No gráfico de barras (figura 74) pode ser visualizado o total de inscrições por curso. Este gráfico fez uso de uma dica de ferramenta que acrescenta, em cada curso, o mês e ano que teve mais inscrições, a média das idades dos inscritos naquele curso e o distrito dos inscritos com mais inscrições.

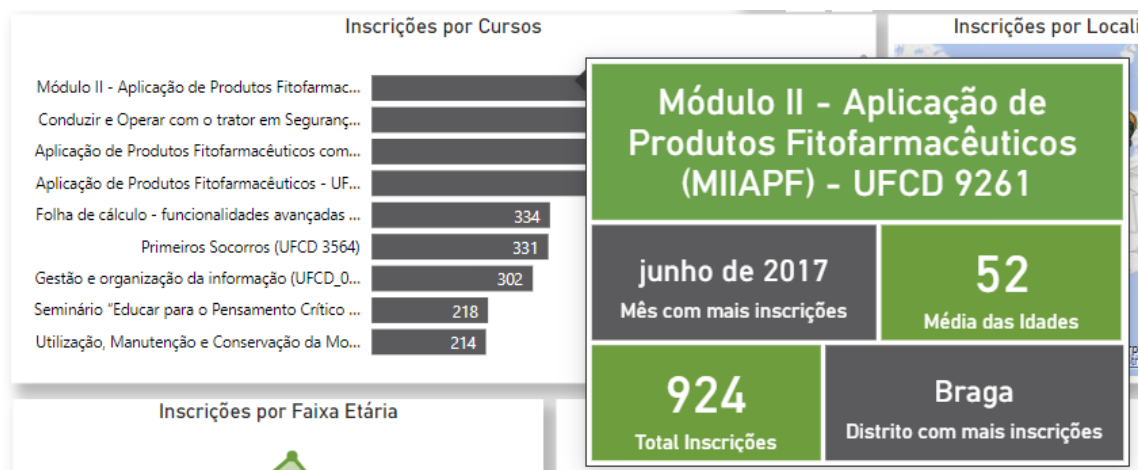


Figura 73 - Dica de ferramenta das inscrições de cada curso

O gráfico de colunas que apresenta o total de inscrições, dividido nas categorias concretizadas e não concretizadas, também faz uso da dica de ferramenta personalizada que acrescenta o total das inscrições naquele mês, somando as concretizadas e não concretizadas, e o curso que mais teve inscrição (figura 74).

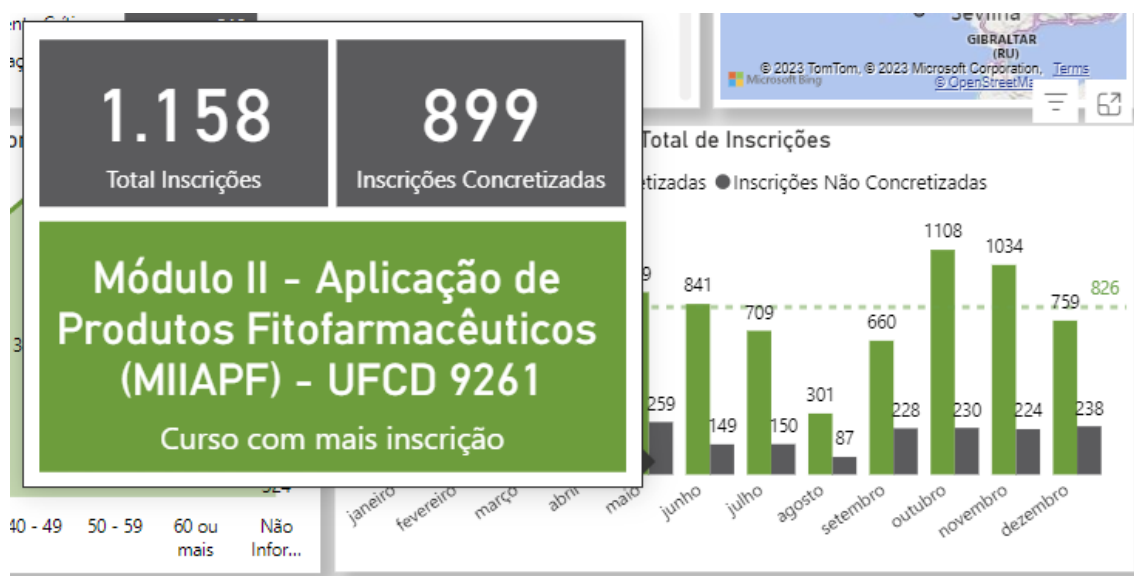


Figura 74 - Gráfico de colunas agrupadas com total de inscrições

# POLI TÉCNICO GUARDA

A página 2 (figura 75) deste projeto também utiliza dois cartões que informa o total de inscrições e o total de manifestações de interesse. Ao lado pode-se filtrar os dados por data. Esta página também faz uso de uma matriz que informa as habilitações dos inscritos com o total de inscrições por habilitações, um gráfico de área que informa a quantidade mensal de manifestações de interesse, um gráfico de barras com as formas de conhecimento do curso e um gráfico de colunas com os tipos de interesse dos inscritos.

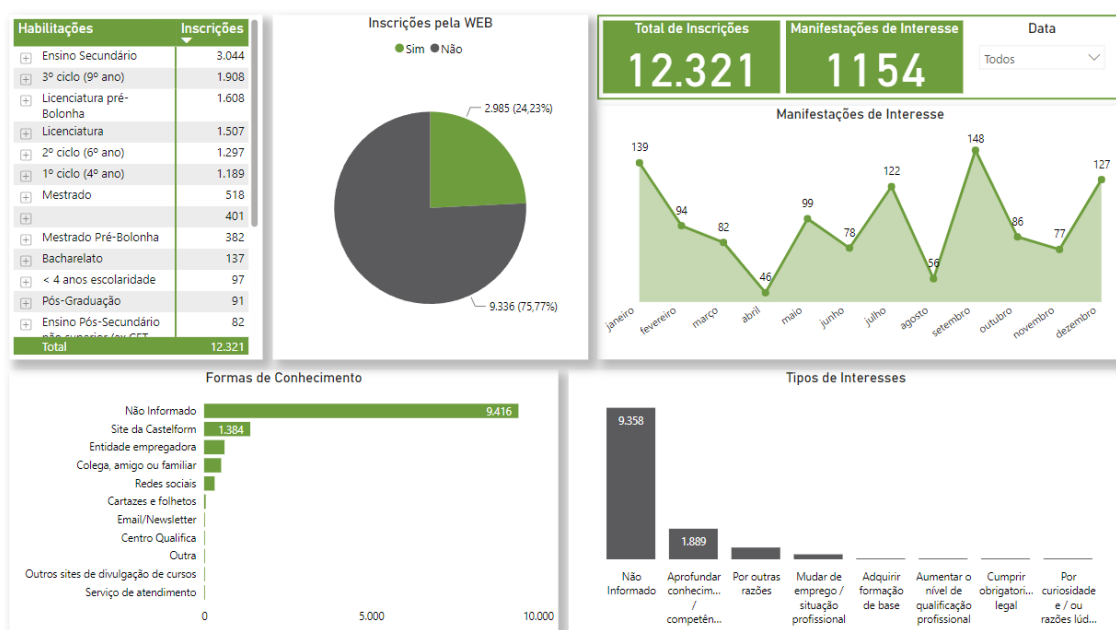


Figura 75 - Página 2 do dashboard dos clientes

# POLI TÉCNICO GUARDA

O gráfico circular, ou gráfico de pizza, divide as inscrições pela forma de realização, se foi através da *web* ou não. Para complementar ainda mais a informação, foi feito o uso da dica de ferramenta que acrescentou a média das idades dos inscritos e o curso com mais inscrições (figura 76).



Figura 76 - Gráfico de pizza



# POLI TÉCNICO GUARDA

A última página deste projeto (figura 77) foi construída exatamente da mesma forma da última página do *dashboard* da Mestreclique.

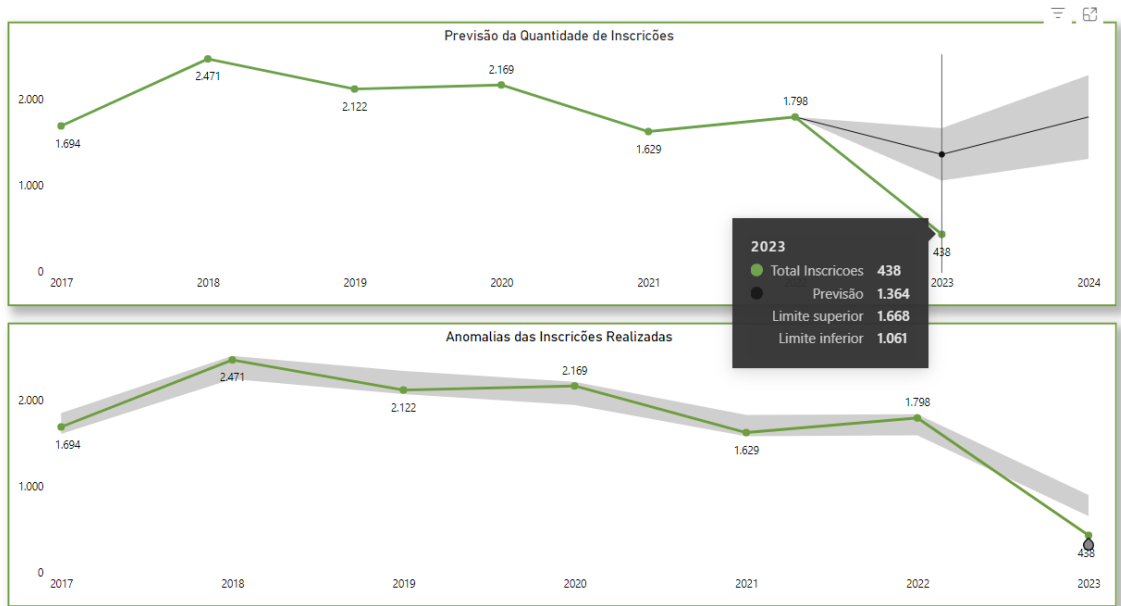


Figura 77 - Página 3 do *dashboard* dos clientes

# POLI TÉCNICO GUARDA

## 8 Dashboards em Serviço

Após o término de toda a caracterização, os *dashboards* foram publicados no *Power BI Service* através do botão “Publicar” do *Power BI Desktop*. É aberta uma janela de forma que haja a escolha do *workspace* da empresa e, em seguida, basta clicar no botão “Selecionar” (figura 78).

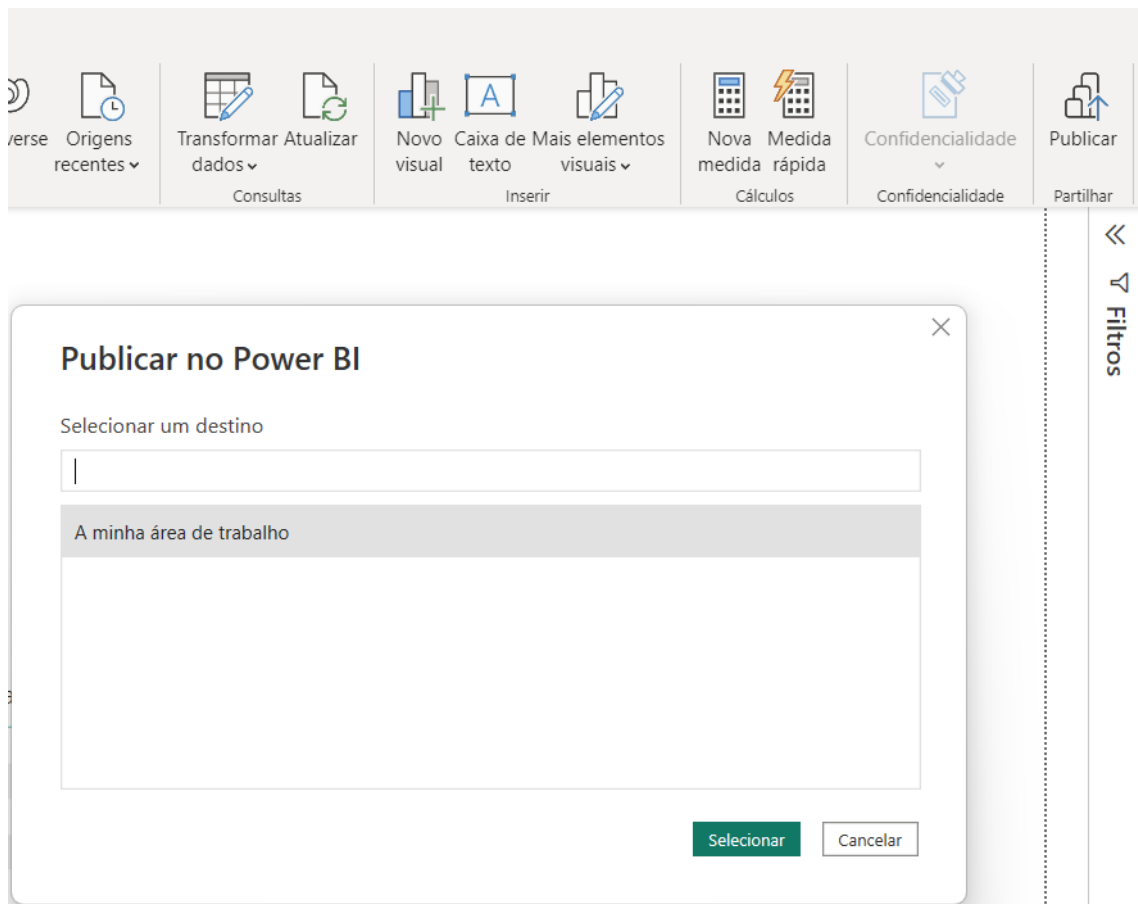


Figura 78 - Publicação através do *Power BI Desktop*

# POLI TÉCNICO GUARDA

## 8.1 GATEWAY

O *gateway* possibilita transferir os dados das fontes para vários serviços on-line de forma rápida e segura. Como a fonte de dados é local foi imprescindível a utilização de um *gateway*.

### 8.1.1 INSTALAÇÃO E CONFIGURAÇÃO DO GATEWAY

O *gateway* é de muita utilidade para transferir os dados locais para os serviços on-line, desta forma é possível atualizar o conjunto de dados utilizados nos *dashboards*.

Antes de fazer qualquer atualização, é necessário realizar o *download*, instalar e configurar o *gateway*. Basta acessar o *Power BI Service*, através do link <https://app.Powerbi.com/>, colocar o *login* e senha e iniciar o *download* e instalação do *Gateway* de Dados.

### 8.1.2 CONFIGURAÇÃO DA FONTE DE DADOS

A configuração do *gateway* em uma fonte de dados é feita através do *Power BI Service*. Basta clicar no botão Configurações e, em seguida, na opção “Gerenciar conexões e gateways” (figura 79).

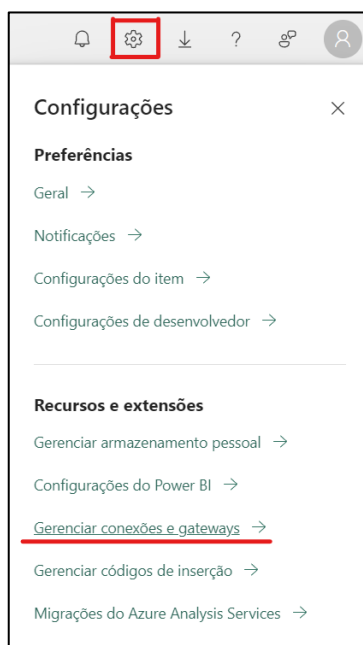


Figura 79 - Gestão de gateways

# POLI TÉCNICO GUARDA

Após este passo basta clicar em “Novo” e inserir as informações requeridas como o nome da fonte de dados, tipo da fonte, servidor, nome da BD e método de autenticação.

## 8.2 ATUALIZAÇÃO DO DASHBOARD

No *Power BI* existem duas formas de atualização dos dados: manual ou automática. No momento da construção dos *dashboards*, no *Power BI Desktop*, pode-se atualizar os dados de forma manual bastando clicar no botão “Atualizar” (figura 80).

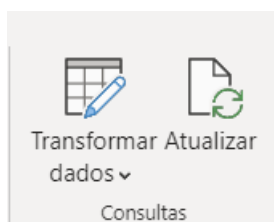


Figura 80 - Botão para atualização manual

Após a publicação do *dashboard* no *Power BI Service*, pode-se utilizar a atualização automática. Basta preencher a frequência de atualização e os horários desejados. É possível agendar até oito horários para ser realizada a atualização automática (figura 81).

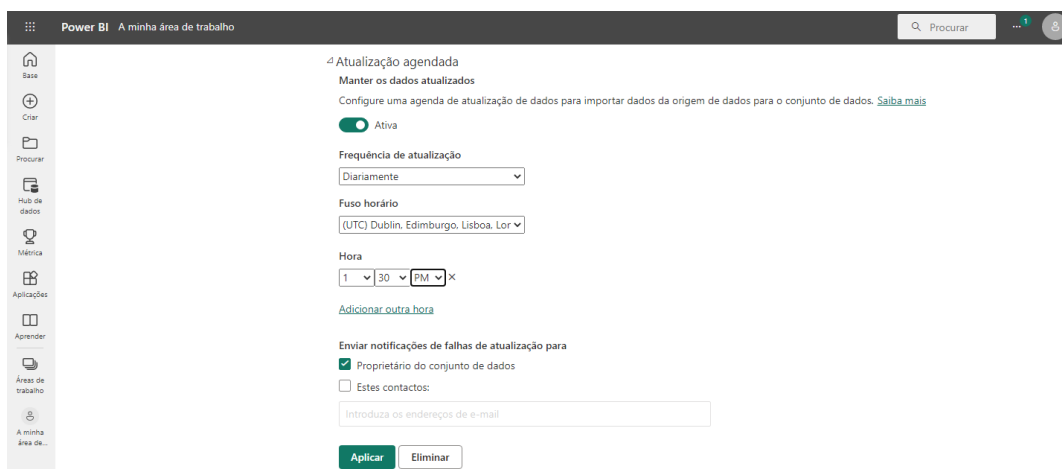


Figura 81 - Atualização automática

# POLI TÉCNICO GUARDA

## 8.3 PUBLICAÇÃO NA WEB

O primeiro passo foi abrir o *FileZilla* e clicar na pasta do cliente. Os arquivos “*Powerbi.aspx*” e “*Powerbi.aspx.vb*” foram transferidos para a pasta do cliente para que a página da *web* fosse criada (figura 82). Para confirmar tal operação, após esse passo o teste era realizado com o endereço da *web* do cliente, acrescentando ao final “*/Powerbi.aspx*”.

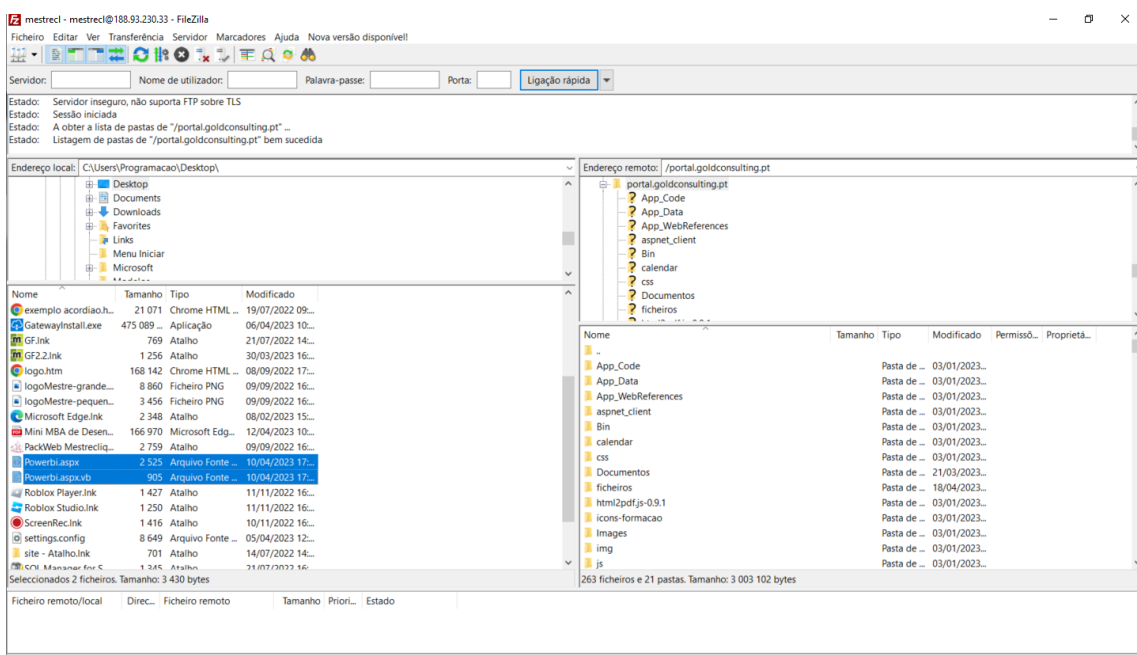


Figura 82 - Transferência dos arquivos no FileZilla

No segundo passo era necessário abrir o *SQL Server*, acessar a base de dados desejada e executar os seguintes comandos:

```
ALTER TABLE [dbo].  
[CONFIGURACOES]  
ADD [LINKPOWERBI]  
varchar(max) NULL  
GO
```

```
ALTER TABLE [dbo].  
[CONFIGURACOES]  
ADD [POWERBI] bit DEFAULT 1  
NOT NULL  
GO
```

# POLI TÉCNICO GUARDA

Esse código SQL altera a tabela CONFIGURACOES incluindo dois campos: o *POWERBI* que é apenas um bit e indica se o cliente tem ou não o *Power BI*, e o *LINKPOWERBI* que é o campo onde é inserido o html gerado pelo *Power BI Service*.

Logo após executar os comandos no *SQL Server*, foi necessário retornar ao *Power BI Service* para gerar o código em html e inserir no campo *LINKPOWERBI* criado no *SQL Server*.

No *Power BI Service*, foi escolhida a opção “Publicar na Web (público)” que aparece após clicar em Ficheiro e Incorporar relatório (figura 83).

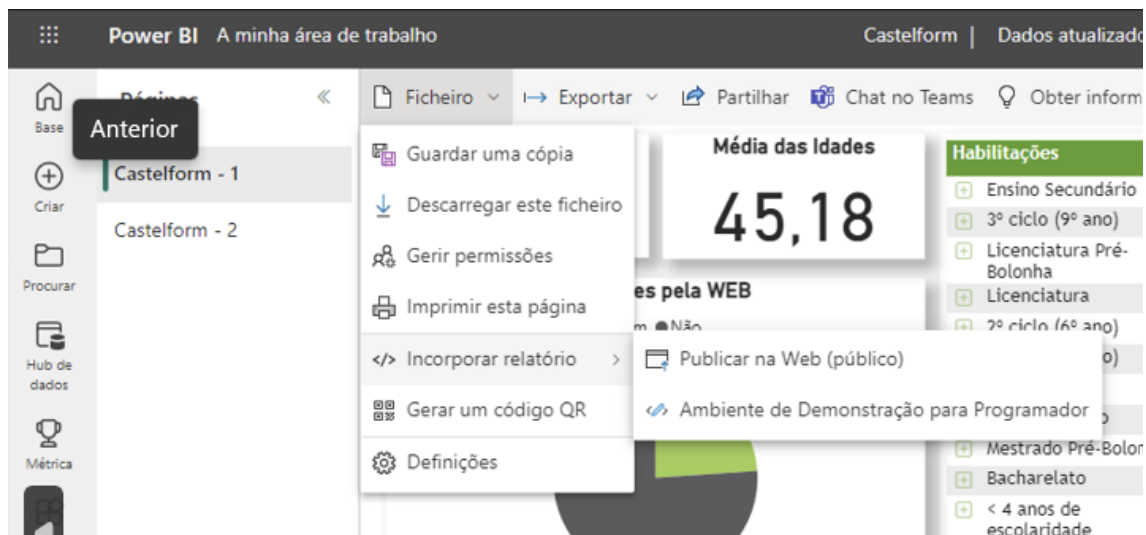


Figura 83 - Publicar na web pelo *Power BI Service*

# POLI TÉCNICO GUARDA

É aberta uma janela onde podemos copiar o link gerado pelo *Power BI Service* (figura 84).

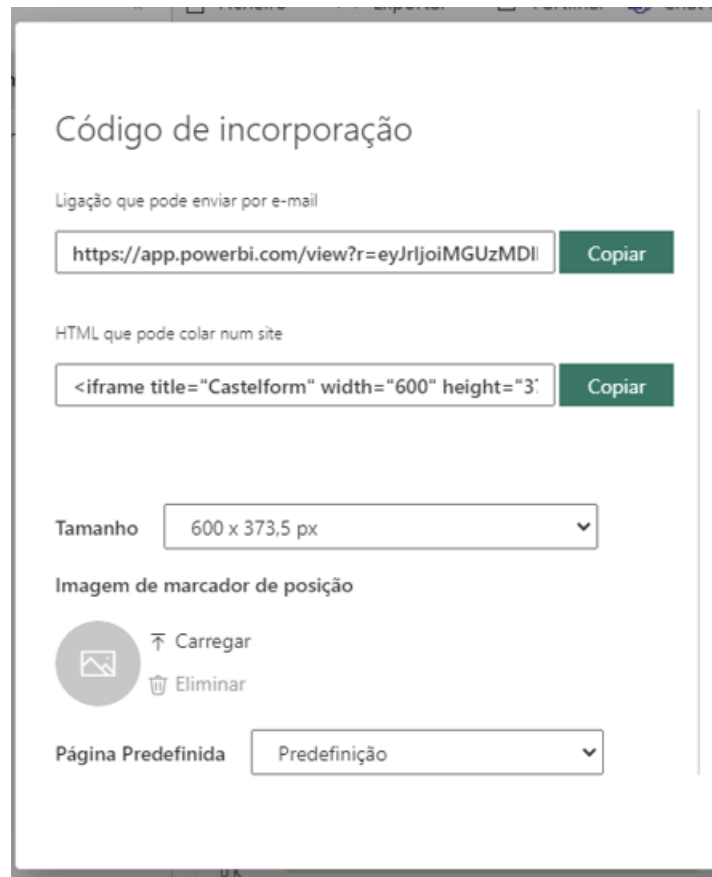


Figura 84 - Código gerado pelo *Power BI Service*

Após a devida cópia do html gerado, seguiu-se para o *SQL Server* onde a tabela *CONFIGURACOES* foi aberta e o link gerado foi inserido no campo *LINKPOWERBI* que foi criado anteriormente.

A finalização do estágio foi realizada no dia no dia 18 de julho de 2023, com a apresentação do *dashboard* em um workshop, com mais de 60 pessoas, direcionado aos clientes da *Mestreclique*, com o tema “Ferramenta de Análise de Dados (*Power BI*)”. A apresentação foi inserida posteriormente no *Youtube* e pode ser acessada através do link: [https://www.youtube.com/watch?v=p\\_75yabkOwE](https://www.youtube.com/watch?v=p_75yabkOwE) .

# POLI TÉCNICO GUARDA

## 9 CONCLUSÃO

A conclusão do estágio pertencente ao CTeSP em Análise de Dados, realizado na empresa Mestreclique, representou uma jornada enriquecedora, trespassada por desafios e aprendizados significativos. Durante esse período, foi dada a oportunidade de não apenas aplicar os conhecimentos teóricos adquiridos ao longo do curso, como também de aprofundar, através de muito estudo e pesquisas, as habilidades práticas na área de *BI* e dados e na utilização da ferramenta *Power BI*.

A criação de dois projetos distintos, um voltado à análise interna da Mestreclique e outro direcionado aos clientes da empresa, permitiu-me explorar as variantes da transformação de dados em verdadeiros insights. A utilização da linguagem DAX na criação de fórmulas e tabelas calculadas expandiu a minha compreensão sobre a análise de dados e proporcionou uma visão mais dinâmica e abrangente. Os projetos propostos foram cumpridos 100% e creio que foi entregue além do que foi pedido.

Ao longo do estágio, pude me deparar com desafios que exigiram criativas soluções e uma abordagem analítica. A compreensão aprofundada do funcionamento do *Power BI* foi essencial para a criação de *dashboards* eficazes, fornecendo à Mestreclique valiosas ferramentas para a tomada de decisões estratégicas embasadas em dados.

Este estágio foi útil para consolidar meus conhecimentos técnicos adquiridos no curso e, também, para desenvolver ainda mais as habilidades interpessoais, o trabalho em equipe e a gestão de projetos. Interagir com profissionais experientes foi uma imersão valiosa no ambiente corporativo e complementou meu aprendizado acadêmico com experiências práticas.

Em suma, o estágio na Mestreclique foi um marco significativo na minha jornada acadêmica e profissional. A experiência que eu adquiri e os desafios que superei enriqueceram o meu repertório e me preparam para futuros desafios na área de Análise de Dados e *Business Intelligence*. Posso afirmar que saio deste estágio com uma base sólida para contribuir nessas áreas e ansiosa por novas conquistas e aprendizados.



# POLI TÉCNICO GUARDA

## 10 Bibliografia

Chen, Hsinchun & Chiang, Roger & Storey, Veda. (2012). *Business Intelligence and Analytics: From Big Data to Big Impact*. MIS Quarterly. 36. 1165-1188. 10.2307/41703503.

Fernandes, D. R. (2017). Uma contribuição sobre a construção de indicadores e sua importância para a gestão empresarial. Revista Da FAE, 7(1). Recuperado de <https://revistafae.fae.edu/revistafae/article/view/430>

Jaklič, Jurij. (2008). Assessing Benefits of *Business Intelligence* Systems – A Case Study. Management: Journal of Contemporary Management Issues (mbuble@efst.hr); Vol.15 No.1.

VS Khatuwal e D. Puri, "*Business Intelligence* Tools for *Dashboard* Development", 2022 3ª Conferência Internacional sobre Engenharia e Gerenciamento Inteligente (ICIEM), Londres, Reino Unido, 2022, pp.

Microsoft. (2023). O que é o *Power BI*? | Microsoft Learn. <https://learn.microsoft.com/pt-pt/Power-bi/fundamentals/Power-bi-overview>

Velcu-Laitinen, Oana & Yigitbasioglu, Ogan. (2012). The Use of *Dashboards* in Performance Management: Evidence from Sales Managers. International Journal of Digital Accounting Research. 12. 39-58. 10.4192/1577-8517-v12\_2.

# POLI TÉCNICO GUARDA

## 11 ANEXOS

# POLI TÉCNICO GUARDA

## 11.1 ANEXO A - Plano de estágio

<b>POLI TÉCNICO GUARDA</b>	<b>PLANO DE TRABALHO</b>	<b>MODELO</b>
	Licenciaturas Cursos Técnicos Superiores Profissionais (CTeSP)	EA.125.07 2022 / 2023

Este documento é um complemento do formulário EA.124 - Convenção.

Tipologia:  Curricular  Extracurricular  Outro: \_\_\_\_\_  
Ao abrigo de protocolo ou especificidade formativa?  Sim. Qual? \_\_\_\_\_

Informação adicional: (se aplicável)  
Designação: \_\_\_\_\_  
Ano curricular: \_\_\_\_\_ Semestre: \_\_\_\_\_  1.º período  2.º período  3.º período

### 1. IDENTIFICAÇÃO DOS INTERVENIENTES

Estudante: Juliana Carneiro da Motta  
Curso: Técnico Superior Profissional em Análise de Dados N.º de estudante: 1706988  
Docente orientador(a): Paulo Alexandre Andrade Vieira  
Supervisor(a)/Tutor(a): Leonardo Lourenço

### 2. PLANO DE TRABALHO

Designação do Projeto:  
Utilização de business intelligence nas bases de dados da Mestrelisque

Resumo do trabalho a desenvolver:  
Desenvolvimento de interfaces no Power BI para análise de dados às bases de dados da Mestrelisque.

Objetivos previstos:  
Estudo e análise do software Power BI;  
Ligação do Power BI à base de dados Microsoft SQL da gestão de tickets da empresa Mestrelisque e às bases de dados dos clientes de gestão de formação;  
Identificação do conjunto de dados a analisar;  
Criação de visualizações de dados;  
Transformação de dados: filtragem, classificação, agrupamento, colunas calculadas, tabelas dinâmicas, entre outros;  
Publicação dos painéis Power BI online.

Conclusões/resultados previstos:  
Conseguir uma interface amigável e clara para a análise de dados às bases de dados da Mestrelisque.

### 3. DATAS E ASSINATURAS

O(A) Estudante  
Data: 08/03/2023 Assinatura: [assinatura]  
(assinatura)

O(A) Docente Orientador(a)  
Data: 08/03/2023 Assinatura: [assinatura]  
(assinatura)

O(A) Supervisor(a) / Tutor(a)  
Data: 8/03/2023 Assinatura: [assinatura]  
Mestrelisque  
Sistemas de Informação, Lda.  
A Gerência  
(assinatura e carimbo da Entidade)

# POLI TÉCNICO GUARDA

## 11.2 ANEXO B – Transformação dos dados nas demais tabelas

Editor Avançado

### TICKETS

Opções de Apresentação

```
let
  Origem = Sql.Database(██████████, "mestreclique_gf"),
  dbo_TICKETS = Origem[[Schema="dbo",Item="T_TICKETS"]][Data],
  #"Colunas Removidas" = Table.RemoveColumns(dbo_TICKETS,{"DESCRICAO", "ASSUNTO", "RESPOSTA", "FICHEIRO", "PRINTSCREEN", "USERINSERT", "NOMECLIENTE", "OBSERVACOES"},
  #"Coluna Condicional Adicionada" = Table.AddColumn(#"Colunas Removidas", "EstadoTicket", each if [CODESTADO] = 1 then "Aberto" else if [CODESTADO] = 2 then "Respo",
  #"Consultas Intercaladas" = Table.NestedJoin(#"Coluna Condicional Adicionada", {"CODTICKET"}, TICKETS_ESTADOS, {"CODTICKET"}, "T_TICKETS_ESTADOS.1", JoinKind.LeftOuter),
  #"Adicionado Tempo Resposta" = Table.AggregateTableColumn(#"Consultas Intercaladas", "T_TICKETS_ESTADOS.1", {{"TEMPO", List.Sum, "Tempo Resposta"}},
  #"Adicionada a Entidade" = Table.ExpandRecordColumn(#"Adicionado Tempo Resposta", "ENTIDADE(CODCLIENTE)", {"NOME", {"Nome da Entidade"}},
  #"Tipo Alterado" = Table.TransformColumnTypes(#"Adicionada a Entidade", {{"Tempo Resposta", Int64.Type}},
  #"Consultas Intercaladas1" = Table.NestedJoin(#"Tipo Alterado", {"CODTECNICO"}, ENTIDADE, {"CODENTIDADE"}, "ENTIDADE", JoinKind.LeftOuter),
  #"ENTIDADE Expandida" = Table.ExpandTableColumn(#"Consultas Intercaladas1", "ENTIDADE", {"NOME", {"ENTIDADE.NOME"}},
  #"Linhas Filtradas" = Table.SelectRows(#"ENTIDADE Expandida", each true),
  #"Tipo Alterado1" = Table.TransformColumnTypes(#"Linhas Filtradas",{"DATAINSERT", type date}, {"DATAFIM", type date})
in
  #"Tipo Alterado1"
```

✓ Não foram detetados erros de sintaxe.

Concluído Cancelar

Editor Avançado

### ENTIDADE

Opções de Apresentação

```
let
  Origem = Sql.Database(██████████, "mestreclique_gf"),
  dbo_ENTIDADE = Origem[[Schema="dbo",Item="ENTIDADE"]][Data],
  #"Colunas Removidas" = Table.RemoveColumns(dbo_ENTIDADE,{"MORADA1L2", "TELEFONE2", "TELEFONE1", "TELENOVEL", "FAX", "CODNACIONALIDADE", "ESTADOCIVIL", "NOMEPAI"},
in
  #"Colunas Removidas"
```

✓ Não foram detetados erros de sintaxe.

Concluído Cancelar

Editor Avançado

### ESTADOS

Opções de Apresentação

```
let
  Origem = Sql.Database(██████████, "mestreclique_gf"),
  dbo_T_ESTADOS = Origem[[Schema="dbo",Item="T_ESTADOS"]][Data],
  #"Colunas Removidas" = Table.RemoveColumns(dbo_T_ESTADOS,{"T_TICKETS", "T_TICKETS_ESTADOS"})
in
  #"Colunas Removidas"
```

✓ Não foram detetados erros de sintaxe.

Concluído Cancelar

# POLI TÉCNICO GUARDA

Editor Avançado

## licencas

Opções de Apresentação ?

```
let
Origem = Sql.Database(██████████, "mestreclique_gf"),
dbo_licencas = Origem([Schema="dbo",Item="licencas"])[Data],
##Colunas Removidas = Table.RemoveColumns(dbo_licencas,{"CHAVE", "DIASOFF", "VERSAO", "URLWS", "URLPW", "ANOTACOES", "KEYNIF", "ENTIDADE", "LICENCAS_REGISTOS"})
in
##Colunas Removidas"
```

✓ Não foram detetados erros de sintaxe.

Concluído Cancelar

Editor Avançado

## TICKETS\_ESTADOS

Opções de Apresentação ?

```
let
Origem = Sql.Database(██████████, "mestreclique_gf"),
dbo_T_TICKETS_ESTADOS = Origem([Schema="dbo",Item="T_TICKETS_ESTADOS"])[Data],
##Coluna Condicional Adicionada = Table.AddColumn(dbo_T_TICKETS_ESTADOS, "Estado_Ticket", each if [CODESTADO] = 1 then "Aberto" else if [CODESTADO] = 2 then "Res
##TICKETS Expandida = Table.ExpandRecordColumn(##Coluna Condicional Adicionada, "T_TICKETS", {"CODETECNICO", {"T_TICKETS.CODETECNICO"}},
##Tipo Alterado = Table.TransformColumnTypes(##T_TICKETS Expandida,{"T_TICKETS.CODETECNICO", type text}),
##Colunas Removidas = Table.RemoveColumns(##Tipo Alterado,{"OBSERVACOES", "DESCRICAO", "FICHEIRO", "PRINTSCREEN", "OPERATION", "T_ESTADOS"}),
##Tipo Alterado1 = Table.TransformColumnTypes(##Colunas Removidas,{"Estado_Ticket", type text})
in
##Tipo Alterado1"
```

✓ Não foram detetados erros de sintaxe.

Concluído Cancelar

Editor Avançado

## INSCRICAO

Opções de Apresentação ?

```
let
Origem = Sql.Database(██████████, "soprofor_nerga"),
dbo_INSCRICAO = Origem([Schema="dbo",Item="INSCRICAO"])[Data],
##Tipo Alterado = Table.TransformColumnTypes(dbo_INSCRICAO,{"DATA", type date}, {"SELECIONADO", type text}),
##Valor Substituído = Table.ReplaceValue(##Tipo Alterado,"true","Sim",Replacer.ReplaceText,{"SELECIONADO"}),
##Valor Substituído1 = Table.ReplaceValue(##Valor Substituído,"false","Não",Replacer.ReplaceText,{"SELECIONADO"}),
##ENTIDADE Expandida = Table.ExpandRecordColumn(##Valor Substituído1, "ENTIDADE", {"CODENTIDADEEE"}, {"ENTIDADE.CODENTIDADEEE}),
##Consultas Intercaladas = Table.NestedJoin(##ENTIDADE Expandida, {"ENTIDADE.CODENTIDADEEE"}, ENTIDADE, {"CODENTIDADEEE"}, "ENTIDADE", JoinKind.LeftOuter),
##ENTIDADE Expandida1 = Table.ExpandTableColumn(##Consultas Intercaladas, "ENTIDADE", {"NOME"}, {"ENTIDADE.NOME}),
##Tipo Alterado1 = Table.TransformColumnTypes(##ENTIDADE Expandida1,{"WEB", type text}),
##Valor Substituído2 = Table.ReplaceValue(##Tipo Alterado1,"true","Sim",Replacer.ReplaceText,{"WEB"}),
##Valor Substituído3 = Table.ReplaceValue(##Valor Substituído2,"false","Não",Replacer.ReplaceText,{"WEB"}),
##Valor Substituído4 = Table.ReplaceValue(##Valor Substituído3,null,"Não",Replacer.ReplaceText,{"WEB"}),
##Consultas Intercaladas1 = Table.NestedJoin(##Valor Substituído4, {"FORMACONHECIMENTO"}, OPCOESINSCRICAO, {"COOLISTA"}, {"OPCOESINSCRICAO", JoinKind.LeftOuter},
##OPCOESINSCRICAO Expandida = Table.ExpandTableColumn(##Consultas Intercaladas1, "OPCOESINSCRICAO", {"NOME"}, {"Forma Conhecimento"}),
##Valor Substituído5 = Table.ReplaceValue(##OPCOESINSCRICAO Expandida,null,"Não Informado",Replacer.ReplaceText,{"Forma Conhecimento"}),
##Consultas Intercaladas2 = Table.NestedJoin(##Valor Substituído5, {"INTERESSE"}, OPCOESINSCRICAO, {"COOLISTA"}, {"OPCOESINSCRICAO", JoinKind.LeftOuter},
##OPCOESINSCRICAO Expandida1 = Table.ExpandTableColumn(##Consultas Intercaladas2, "OPCOESINSCRICAO", {"NOME"}, {"Interesses"}),
##Valor Substituído6 = Table.ReplaceValue(##OPCOESINSCRICAO Expandida1,null,"Não Informado",Replacer.ReplaceText,{"Interesses"}),
in
##Tipo Alterado1"
```

✓ Não foram detetados erros de sintaxe.

Concluído Cancelar

# POLI TÉCNICO GUARDA

Editor Avançado

## ACCAO

Opções de Apre

```
let
  Origem = Sql.Database(██████████████████, "soprofor_nerga"),
  dbo_ACCAO = Origem{[Schema="dbo",Item="ACCAO"]}[Data],
  #"Consultas Intercaladas" = Table.NestedJoin(dbo_ACCAO, {"CODTIPOPROJ"}, TIPO_PROJECTOS, {"CODTIPOPROJ"}, "TIPO_PROJECTOS", JoinKind.LeftOuter),
  #"TIPO_PROJECTOS Expandida" = Table.ExpandTableColumn(#"Consultas Intercaladas", "TIPO_PROJECTOS", {"FINANCIADO"}, {"FINANCIADO"}),
  #"Outras Colunas Removidas" = Table.SelectColumns(#"TIPO_PROJECTOS Expandida",{"CODACCAO", "CODCURSO", "DESCACCAO", "NUMACCAO", "FINANCIADO"})
in
  #"Outras Colunas Removidas"
```

Editor Avançado

## CONCELHOS

```
let
  Origem = Sql.Database(██████████████████, "soprofor_nerga"),
  dbo_CONCELHOS = Origem{[Schema="dbo",Item="CONCELHOS"]}[Data],
  #"Outras Colunas Removidas" = Table.SelectColumns(dbo_CONCELHOS,{"CODCONCELHO", "CONCELHO", "CODDISTRITO"})
in
  #"Outras Colunas Removidas"
```

Editor Avançado

## CURSOS

Opções de Apresentação

```
let
  Origem = Sql.Database(██████████████████, "soprofor_nerga"),
  dbo_CURSOS = Origem{[Schema="dbo",Item="CURSOS"]}[Data],
  #"Consultas Intercaladas" = Table.NestedJoin(dbo_CURSOS, {"CODMODALIDADE"}, TIPO_PROJECTOS, {"CODTIPOPROJ"}, "TIPO_PROJECTOS", JoinKind.LeftOuter),
  #"TIPO_PROJECTOS Expandida" = Table.ExpandTableColumn(#"Consultas Intercaladas", "TIPO_PROJECTOS", {"FINANCIADO"}, {"FINANCIADO"}),
  #"Valor Substituído1" = Table.ReplaceValue(#"TIPO_PROJECTOS Expandida", "3564-Primeiros Socorros", "3564 - Primeiros Socorros", Replacer.ReplaceText, {"DESCCURSO"}),
  #"Valor Substituído2" = Table.ReplaceValue(#"Valor Substituído1", "Primeiros Socorros", "3564 - Primeiros Socorros", Replacer.ReplaceText, {"DESCCURSO"}),
  #"Valor Substituído3" = Table.ReplaceValue(#"Valor Substituído2", "3564 - 3564 - Primeiros Socorros", "3564 - Primeiros Socorros", Replacer.ReplaceText, {"DESCCURSO"}),
  #"Outras Colunas Removidas" = Table.SelectColumns(#"Valor Substituído3", {"CODCURSO", "DESCCURSO", "CODMODALIDADE", "FINANCIADO"}),
  #"Valor Substituído4" = Table.ReplaceValue(#"Outras Colunas Removidas", "0460 - Desenho assistido por computador", "0460 - Desenho assistido por computador", Replac
  #"Valor Substituído5" = Table.ReplaceValue(#"Valor Substituído4", "5436 - Liderança e Motivação de Equipas", "5436 - Liderança e motivação de equipas", Replac
  #"Linhas Ordenadas" = Table.Sort(#"Valor Substituído5", {"DESCCURSO", Order.Ascending}),
  #"Valor Substituído6" = Table.ReplaceValue(#"Linhas Ordenadas", "Ambiente, Segurança, Higiene e Saúde no Trabalho - conceitos básicos", "0349 - Ambiente, Segurança,
  #"Valor Substituído7" = Table.ReplaceValue(#"Valor Substituído6", "Língua inglesa - atendimento", "0354 - Língua inglesa - atendimento", Replacer.ReplaceText, {"DESCC
  #"Valor Substituído8" = Table.ReplaceValue(#"Valor Substituído7", "0382 - Gestão do tempo e organização do trabalho", "0382 - Gestão do Tempo e Organização do Traba
  #"Valor Substituído9" = Table.ReplaceValue(#"Valor Substituído8", "0382-Gestão do Tempo e Organização do trabalho", "0382 - Gestão do Tempo e Organização do Traba
  #"Valor Substituído10" = Table.ReplaceValue(#"Valor Substituído9", "0683-Ética e deontologia profissionais", "0683 - Ética e deontologia profissionais", Replacer.Rep
  #"Valor Substituído11" = Table.ReplacValue(#"Valor Substituído10", "0437-Criatividade em comunicação", "0437 - Criatividade em comunicação", Replac.ReplacText /
```

✓ Não foram detetados erros de sintaxe.

Concluído Cancelar

# POLI TÉCNICO GUARDA

## DISTRITOS

```
let
  Origem = Sql.Database(██████████, "soprofor_nerga"),
  dbo_DISTRITOS = Origem{[Schema="dbo",Item="DISTRITOS"]}[Data],
  #"Outras Colunas Removidas" = Table.SelectColumns(dbo_DISTRITOS,{"CODDISTRITO", "DISTRITO"})
in
  #"Outras Colunas Removidas"
```

Editor Avançado

## ENTIDADE

Opções de Apresentação ⓘ

```
let
  Origem = Sql.Database(██████████, "soprofor_nerga"),
  dbo_ENTIDADE = Origem{[Schema="dbo",Item="ENTIDADE"]}[Data],
  #"Tipo Alterado" = Table.TransformColumnTypes(dbo_ENTIDADE,{{"CODHABILITACAO", type text}}),
  #"Valor Substituído" = Table.ReplaceValue(#"Tipo Alterado",null,"Não Informado",Replacer.ReplaceValue,{"CODHABILITACAO"}),
  #"Valor Substituído1" = Table.ReplaceValue(#"Valor Substituído",null,"Não Informado",Replacer.ReplaceValue,{"DESCHABILITACAO_COMP"}),
  #"Valor Substituído2" = Table.ReplaceValue(#"Valor Substituído1","", "Não Informado",Replacer.ReplaceValue,{"DESCHABILITACAO_COMP"}),
  #"Outras Colunas Removidas" = Table.SelectColumns(#"Valor Substituído2",{"CODENTIDADE", "NOME", "MORADA111", "CODPOSTAL", "LOCALIDADE", "CONCELHO", "DATANASCIMENTO"})
in
  #"Outras Colunas Removidas"
```

✓ Não foram detetados erros de sintaxe.

Concluído Cancelar

Editor Avançado

## HABILITACOES

```
let
  Origem = Sql.Database(██████████, "soprofor_nerga"),
  dbo_HABILITACOES = Origem{[Schema="dbo",Item="HABILITACOES"]}[Data],
  #"Outras Colunas Removidas" = Table.SelectColumns(dbo_HABILITACOES,{"CODHABILITACAO", "DESCHABILITACAO"})
in
  #"Outras Colunas Removidas"
```

# POLI TÉCNICO GUARDA

Editor Avançado

## OPCOESINSCRICAO

```
let
  Origem = Sql.Database(██████████, "soprofor_nerga"),
  dbo OPCOESINSCRICAO = Origem{[Schema="dbo",Item="OPCOESINSCRICAO"]}[Data],
  #"Outras Colunas Removidas" = Table.SelectColumns(dbo OPCOESINSCRICAO,{"CODLISTA", "NOME", "TIPO", "DESCRICAO"})
in
  #"Outras Colunas Removidas"
```

Editor Avançado

## SITUACAO\_FACE\_EMPREGO

```
let
  Origem = Sql.Database(██████████, "soprofor_nerga"),
  dbo_SITUACAO_FACE_EMPREGO = Origem{[Schema="dbo",Item="SITUACAO_FACE_EMPREGO"]}[Data],
  #"Outras Colunas Removidas" = Table.SelectColumns(dbo_SITUACAO_FACE_EMPREGO,{"CODSFE", "DESCSFE", "DESCSFESIFFSE"})
in
  #"Outras Colunas Removidas"
```

Editor Avançado

## TIPO\_PROJECTOS

```
let
  Origem = Sql.Database(██████████, "soprofor_nerga"),
  dbo_TIPO_PROJECTOS = Origem{[Schema="dbo",Item="TIPO_PROJECTOS"]}[Data],
  #"Tipo Alterado" = Table.TransformColumnTypes(dbo_TIPO_PROJECTOS,{"FINANCIADO", type text}),
  #"Valor Substituído" = Table.ReplaceValue(#"Tipo Alterado", "true", "Sim", Replacer.ReplaceText, {"FINANCIADO"}),
  #"Valor Substituído1" = Table.ReplaceValue(#"Valor Substituído", "false", "Não", Replacer.ReplaceText, {"FINANCIADO"}),
  #"Outras Colunas Removidas" = Table.SelectColumns(#"Valor Substituído1", {"CODTIPOPROJ", "DESCTIPOPROJ", "FINANCIADO"})
in
  #"Outras Colunas Removidas"
```