



INSTITUTO POLITÉCNICO
DO CÁVADO E DO AVE
ESCOLA SUPERIOR
DE TECNOLOGIA

Data Processing and Visualization (P02)

DECISION SUPPORT SYSTEMS, 2024-25

João Ribeiro (23795), José Senra (21154), Pedro Pérez (18623)

Introduction

Este relatório descreve o desenvolvimento de uma solução analítica de Business Intelligence (BI) no âmbito do Projeto 02 da unidade curricular de Sistemas de Apoio à Decisão (SAD). O projeto tem como objetivo apoiar a tomada de decisões estratégicas na companhia aérea **Rota do Sol**, através da análise integrada de dados operacionais e comerciais.

A solução foi desenvolvida utilizando o Microsoft Power BI e baseia-se principalmente na base de dados `br_airport_dw`, construída a partir de dados abertos da Agência Nacional de Aviação Civil (ANAC), complementada com fontes auxiliares.

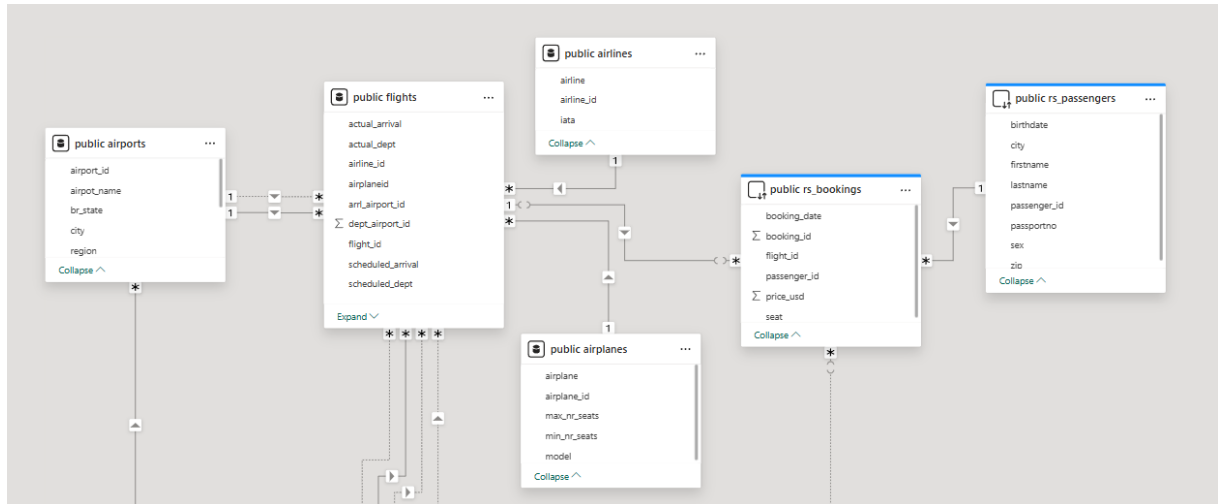
Os objetivos principais do projeto são:

- Criar um modelo de dados robusto que integre diversas fontes de informação;
- Preparar e transformar os dados segundo as melhores práticas, utilizando Power Query e DAX;
- Calcular medidas relevantes e indicadores de desempenho (KPIs) para análise estratégica;
- Desenvolver dashboards e relatórios interativos que permitam:
 - Analisar o desempenho da companhia em 2024 comparado com anos anteriores e com previsões;
 - Avaliar atrasos de voos por categorias de tempo;
 - Explorar rotas operadas e oportunidades de otimização;
 - Apresentar recomendações baseadas em dados para a gestão da companhia.

A solução final visa fornecer à direção da Rota do Sol uma ferramenta eficaz para monitorizar a operação, identificar oportunidades de melhoria e tomar decisões baseadas em dados.

Data acquisition and preparation

1.1. br_airport_dw



Base de dados criada a partir de dados fornecidos pela agência nacional de aviação civil (ANAC).
Realizamos as seguintes alterações:

- Limpeza de dados (Remoção de linhas na tabela flights com valores nulos);
- Criação da tabela DateTime.

DateTime	Date	Year	Month	Month Number	Day	Weekday	Weekday Number	Hour	Minute	Quarter	AM/PM	Time Segment
1/1/2024 12:00:00 AM	1/1/2024 12:00:00 AM	2024	January		1	Monday		0	0	Q1	AM	Early Morning
1/8/2024 12:00:00 AM	1/8/2024 12:00:00 AM	2024	January		8	Monday		0	0	Q1	AM	Early Morning
1/15/2024 12:00:00 AM	1/15/2024 12:00:00 AM	2024	January		15	Monday		0	0	Q1	AM	Early Morning
1/22/2024 12:00:00 AM	1/22/2024 12:00:00 AM	2024	January		22	Monday		0	0	Q1	AM	Early Morning
1/29/2024 12:00:00 AM	1/29/2024 12:00:00 AM	2024	January		29	Monday		0	0	Q1	AM	Early Morning
1/1/2024 12:01:00 AM	1/1/2024 12:00:00 AM	2024	January		1	Monday		0	1	Q1	AM	Early Morning
1/8/2024 12:01:00 AM	1/8/2024 12:00:00 AM	2024	January		8	Monday		0	1	Q1	AM	Early Morning
1/15/2024 12:01:00 AM	1/15/2024 12:00:00 AM	2024	January		15	Monday		0	1	Q1	AM	Early Morning
1/22/2024 12:01:00 AM	1/22/2024 12:00:00 AM	2024	January		22	Monday		0	1	Q1	AM	Early Morning
1/29/2024 12:01:00 AM	1/29/2024 12:00:00 AM	2024	January		29	Monday		0	1	Q1	AM	Early Morning
1/1/2024 12:02:00 AM	1/1/2024 12:00:00 AM	2024	January		1	Monday		0	2	Q1	AM	Early Morning
1/8/2024 12:02:00 AM	1/8/2024 12:00:00 AM	2024	January		8	Monday		0	2	Q1	AM	Early Morning

```
DateTime =  
VAR MinDateTime = MIN('public flights'[scheduled_dept])  
VAR MaxDateTime = MAX('public flights'[scheduled_dept])  
  
VAR DateList = CALENDAR(DATE(YEAR(MinDateTime),1,1), DATE(YEAR(MaxDateTime),12,31))  
  
VAR HourList =  
    SELECTCOLUMNS(  
        GENERATESERIES(0, 23, 1),  
        "Hour", [Value]  
    )  
  
VAR MinuteList =  
    SELECTCOLUMNS(  
        GENERATESERIES(0, 59, 1),  
        "Minute", [Value]  
    )
```

```
RETURN
ADDCOLUMNS(
    CROSSJOIN(DateList, HourList, MinuteList),
    "DateTime", [Date] + TIME([Hour], [Minute], 0),
    "Year", YEAR([Date]),
    "Month", FORMAT([Date], "MMMM"),
    "Month Number", MONTH([Date]),
    "Day", DAY([Date]),
    "Weekday", FORMAT([Date], "dddd"),
    "Weekday Number", WEEKDAY([Date], 2),
    "Quarter", "Q" & FORMAT([Date], "Q"),
    "AM/PM", IF([Hour] < 12, "AM", "PM"),
    "Time Segment",
        SWITCH(
            TRUE(),
            [Hour] < 6, "Early Morning",
            [Hour] < 12, "Morning",
            [Hour] < 17, "Afternoon",
            [Hour] < 21, "Evening",
            "Night"
        )
    )
)
```

1.2. População cidade - Wikipédia

	City	State	2024 Estimate ^[4]	2022 Census ^[5]	Change
1	São Paulo	 São Paulo	11,895,578	11,451,245	+3.88%
2	Rio de Janeiro	 Rio de Janeiro	6,729,894	6,211,423	+8.35%
3	Brasília	 Distrito Federal	2,982,818	2,817,068	+5.88%
4	Fortaleza	 Ceará	2,574,412	2,428,678	+6.00%
5	Salvador	 Bahia	2,568,928	2,418,005	+6.24%
6	Belo Horizonte	 Minas Gerais	2,416,339	2,315,560	+4.35%
7	Manaus	 Amazonas	2,279,686	2,063,547	+10.47%
8	Curitiba	 Paraná	1,829,225	1,773,733	+3.13%
9	Recife	 Pernambuco	1,587,707	1,488,920	+6.63%
10	Goiânia	 Goiás	1,494,599	1,437,237	+3.99%
11	Belém	 Pará	1,398,531	1,303,389	+7.30%
12	Porto Alegre	 Rio Grande do Sul	1,389,322	1,332,570	+4.26%
13	Guarulhos	 São Paulo	1,345,364	1,291,784	+4.15%
14	Campinas	 São Paulo	1,185,977	1,138,309	+4.19%
15	São Luís	 Maranhão	1,088,057	1,037,775	+4.85%
16	Maceió	 Alagoas	994,464	957,916	+3.82%
17	São Gonçalo	 Rio de Janeiro	960,652	896,744	+7.13%
18	Campo Grande	 Mato Grosso do Sul	954,537	897,938	+6.30%
19	Teresina	 Piauí	902,644	866,300	+4.20%

Esta tabela representa a população de várias cidades brasileiras nos anos de 2023 e 2024.
(https://en.wikipedia.org/wiki/List_of_cities_in_Brazil_by_population)

Realizamos as seguintes alterações:

- Remover todas as cidades que não estão presentes na base de dados principal.

Merge

Select a table and matching columns to create a merged table.

população cidade

City	State	2024Estimate[4]	2022Census[5]	Change
São Paulo	São Paulo	11895578	11451245	3.88%
Rio de Janeiro	Rio de Janeiro	6729894	6211423	8.35%
Brasília	Distrito Federal	2982818	2817068	5.88%
Fortaleza	Ceará	2574412	2428678	6.00%
Salvador	Bahia	2568928	2418005	6.24%

public airports

airport_id	airpot_name	city	br_state	region
10	A. I. Belém	Belém	PA	Nordeste
20	Juscelino Kubitschek	Brasília	DF	Centro-oeste
30	Tancredo Neves	Belo Horizonte	MG	Centro-oeste
40	Afonso Pena	Curitiba	PR	Sul
50	Marechal Rondon	Cuiabá	MG	Centro-oeste

Join Kind

Inner (only matching rows)

☐ Use fuzzy matching to perform the merge

▸ Fuzzy matching options

OK

Cancel

- Trocar a tabela estado pelo ID do mesmo, usando a tabela “airports” como referência.

Expand public airports

Select the columns to expand.

Search Columns

- ☒ (Select All Columns)
- ☐ airport_id
- ☐ airport_name
- ☐ city
- ☒ br_state
- ☐ region
- ☐ public.flights(airport_id)
- ☐ public.flights(airport_id) 2

Default column name prefix (optional)

public airports

OK

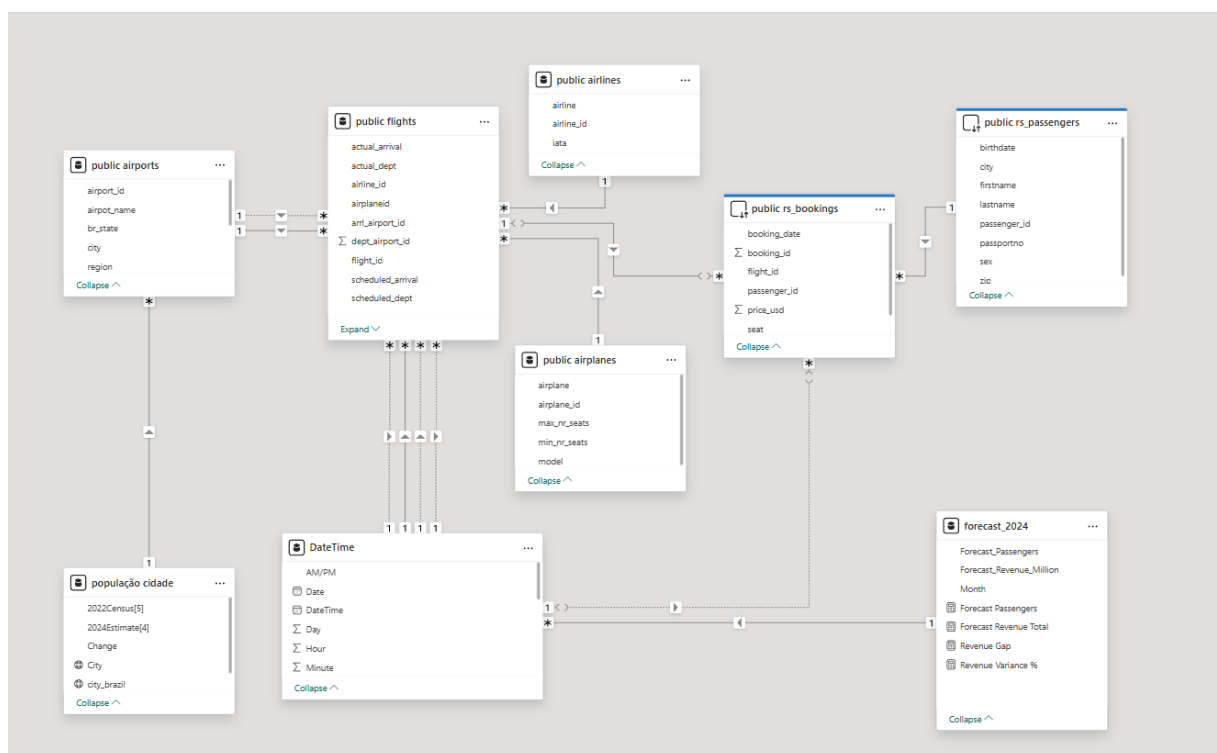
Cancel

1.3. forecast_2024

Tabela com as previsões de rendimento da companhia aérea Rota do Sol para o ano de 2024.

Month	Forecast_Passengers	Forecast_Revenue_Million
1	650000	120.86
2	550000	103.04
3	550000	102.39
4	420000	80.63
5	250000	47.16
6	330000	61.24
7	510000	95.4
8	670000	125.28
9	750000	139.77
10	880000	164.4
11	910000	169.96
12	1000000	186.84

Data modelling and processing



1.4. Relationships

- Tabela DateTime foi ligada à tabela flights 4 vezes, sendo essas relações sempre originadas da coluna DateTime e sendo ligada às colunas Scheduled_Dept, Scheduled_arrival, Actual_dept e Actual_arrival, através de uma ligação 1 para muitos.

- Tabela DateTime foi ligada à tabela rs_bookings entre as colunas DateTime e Booking_date, através de uma ligação 1 para muitos.
- Tabela Populacao cidade foi ligada à tabela airports entre as colunas city, através de uma ligação 1 para muitos.
- Tabela Forecast_2024 foi ligada à tabela DateTime entre as colunas month/month number, através de uma ligação 1 para muitos.

1.5. Hierarquias

- ✓ Hierarquia **year**

Year Hierarchy
Σ Year
Quarter
Σ Month Number
Σ Day
Σ Hour
Σ Minute

- ✓ Hierarquia **region**

region Hierarchy
region
br_state
city
airpot_name

1.6. Calculated columns

- ✓ **Delay category:** categorização do horário de chegada do avião.

```

Delay Category =
SWITCH(
    TRUE(),
    'public flights'[arrv_delay_minutes] < 0, "Early",
    'public flights'[arrv_delay_minutes] = 0, "On Time",
    'public flights'[arrv_delay_minutes] <= 15, "0-15 min",
    'public flights'[arrv_delay_minutes] <= 30, "16-30 min",

```

```

'public flights'[arrv_delay_minutes] <= 60, "31-60 min",
'public flights'[arrv_delay_minutes] <= 120, "61-120 min",
'public flights'[arrv_delay_minutes] > 120, ">120 min",
"No Data"
)

```

- ✓ **route:** define o aeroporto de saída e de chegada.

```
Route = 'public flights'[dept_airport_id] & " → " & 'public flights'[arrrl_airport_id]
```

1.7. Measures

- ✓ **% delayed flights:** calcula a percentagem de voos atrasados.

```

% Delayed Flights =
DIVIDE([Delayed Flights], [Total Flights], 0)

```

- ✓ **delayed flights:** calcula o número de voos atrasados.

```

Delayed Flights =
CALCULATE(
COUNTROWS('public flights'),
'public flights'[arrv_delay_minutes] > 15
)

```

- ✓ **FlightsFromTopAirport:** calcula o número de voos do aeroporto com mais voos.

```

FlightsFromTopAirport =
VAR SelectedAirport = [mostflights_airport]
RETURN
    CALCULATE (
        COUNTROWS ( 'public flights' ),
        FILTER (
            'public flights',
            RELATED ( 'public airports'[airpot_name] ) = SelectedAirport
        )
    )

```

- ✓ **Forecast Passengers:** calcula a soma dos passageiros previstos.

```

Forecast Passengers =
SUM(forecast_2024[Forecast_Passengers])

```

- ✓ **Forecast Revenue:** calcula a receita prevista

```
Forecast Revenue Total =  
SUM(forecast_2024[Forecast_Revenue_Million]) * 1000000
```

- ✓ **mostflights_airport:** calcula o aeroporto com mais voos

```
mostflights_airport =  
VAR MostFlights = MAXX(  
    VALUES('public airports'[airpot_name]),  
    [Total Flights]  
)  
RETURN  
MAXX(  
    FILTER(  
        VALUES('public airports'[airpot_name]),  
        [Total Flights] = MostFlights  
    ),  
    'public airports'[airpot_name]  
)
```

- ✓ **Revenue Gap:** calcula a diferença entre a previsão e o dinheiro ganho

```
Revenue Gap = [Total Revenue] - [Forecast Revenue Total]
```

- ✓ **Revenue Variance %:** calcula a percentagem da diferença entre a previsão e o dinheiro ganho

```
Revenue Variance % =  
DIVIDE([Revenue Gap], [Forecast Revenue Total], 0)
```

- ✓ **Total Flights:** calcula o total de voos

```
Total Flights =  
DISTINCTCOUNT('public flights'[flight_id])
```

- ✓ **Total Passengers:** calcula o total de passageiros

```
Total Passengers =  
SUM('public flights'[pax_count])
```

- ✓ **Total Revenue:** calcula a receita total

```
Total Revenue =  
SUMX (
```



```

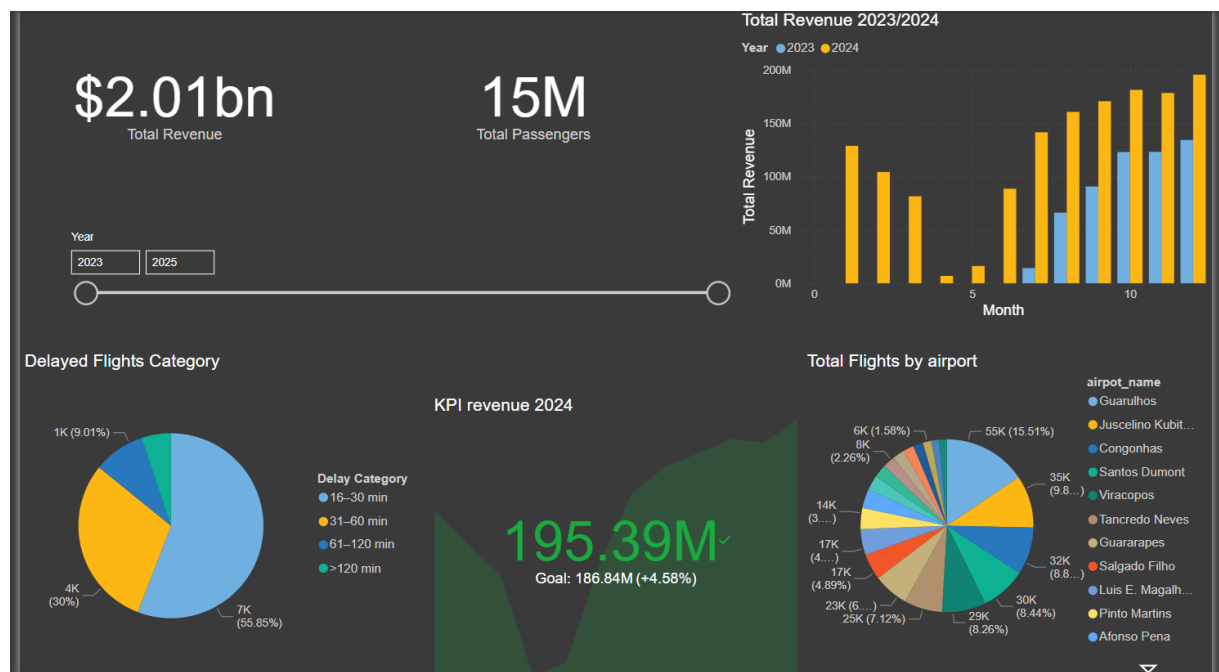
'public flights',
AVERAGEX (
RELATEDTABLE('public rs_bookings'),
'public rs_bookings'[price_usd]
) * 'public flights'[pax_count]
)

```

Data visualization

1.8. Dashboard

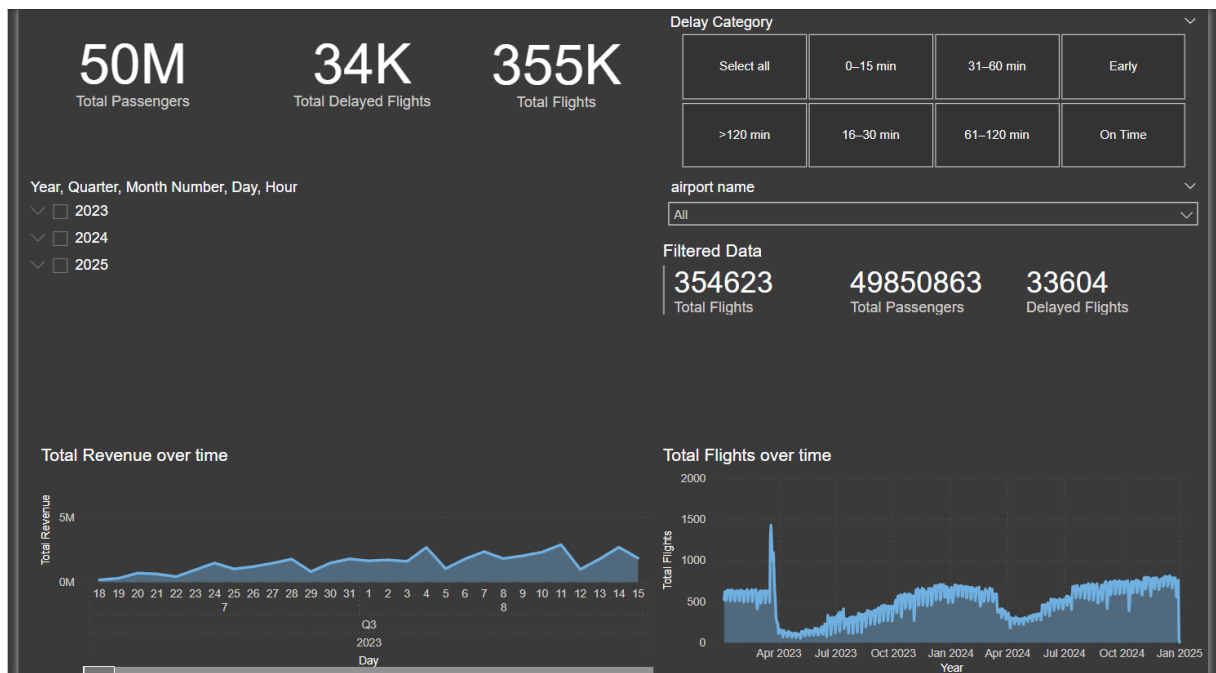
Página de visualização simplificada dos dados mais importantes da airline a rota do sol.



- ✓ **2 cards:** valor numérico da receita e do total de passageiros;
- ✓ **2 pie charts:** número de voos atrasados por categoria e total de voos por aeroporto;
- ✓ **1 clustered column chart:** receita total de 2023/2024;
- ✓ **1 KPI:** comparação da receita total com a receita prevista;
- ✓ **1 slicer:** escolher ano entre 2023 e 2025.

1.9. Report 1

Página de visualização com a informação geral de todas as companhias.



- ✓ **3 cards:** valor numérico do total de passageiros, total de voos atrasados e total de voos;
- ✓ **1 multi-row card:** mesmos valores dos cards mas filtrados por slicers;
- ✓ **2 area charts:** receita total e total de voos ao longo do tempo;
- ✓ **3 slicers:** filtrar entre tempo, categoria de atraso e aeroporto.

1.10. Report 2

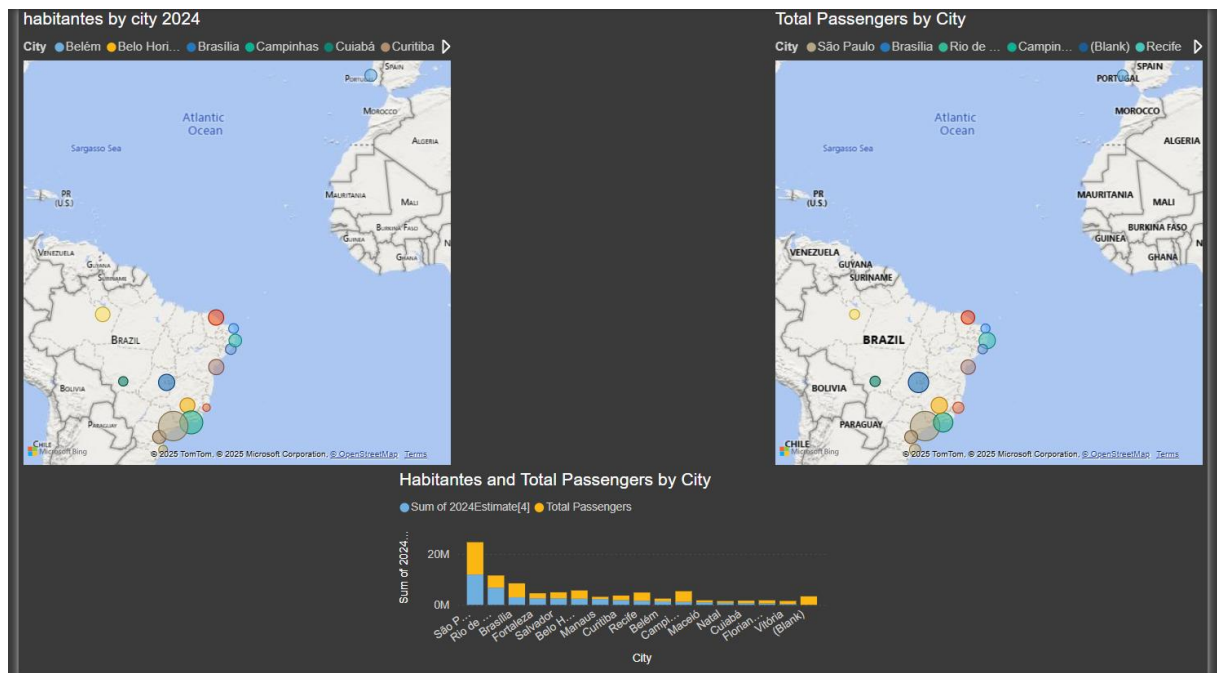
Página de visualização dos dados correspondentes às várias rotas.



- ✓ **3 stacked column charts:** total de voos, receitas e a média de passageiros por rota;
- ✓ **1 clustered column chart:** receita e receita prevista de 2024;
- ✓ **1 slicer:** filtrar entre tempo.

1.11. Report 3

Página de visualização de passageiros e habitantes por cidades.



- ✓ **2 maps:** habitantes por cidade e total de passageiros por cidade;
- ✓ **1 stacked column chart:** comparação entre total de habitantes e total de passageiros.

1.12. Report 4

Página de visualização de aeroporto com mais voos.



- ✓ **2 cards:** aeroporto com mais voos e valor numérico de número de voos do referente aeroporto;
- ✓ **2 slicers:** escolher o intervalo de tempo.

Conclusion

Este projeto permitiu o desenvolvimento de uma solução de Business Intelligence eficaz para a companhia aérea Rota do Sol, com o objetivo de apoiar a tomada de decisões estratégicas baseadas em dados. O processo envolveu a aquisição, limpeza e modelação de dados provenientes de fontes como a ANAC e projeções internas da empresa, utilizando o Microsoft Power BI.

Os dashboards e relatórios desenvolvidos oferecem uma visão clara e interativa do desempenho operacional e comercial da companhia, permitindo a análise de indicadores essenciais como receitas, número de passageiros, atrasos e rotas mais relevantes. A utilização de técnicas de Power Query, DAX e visualizações avançadas possibilitou a criação de relatórios dinâmicos e personalizados, adaptados às necessidades da gestão.

A integração de dados populacionais, previsões e informações operacionais resultou numa ferramenta abrangente, que facilita o monitoramento das operações e a identificação de oportunidades de melhoria. Resumidamente, a solução criada contribui para a eficiência da gestão da Rota do Sol, reforçando a importância da análise de dados no processo de tomada de decisões.