

# Resumo Executivo - Dados Dashboard de Atendimento



## Dados Gerados com Sucesso

Foram criados **dados fictícios realistas** para um dashboard completo de atendimento ao cliente, simulando 6 meses de operação (Fevereiro a Agosto 2024).



## Números Principais

- **35 agentes** distribuídos em 5 departamentos
- **14.943 tickets** com histórico completo
- **187 dias** de dados operacionais
- **Taxa de resolução:** 62.8%
- **Satisfação média:** 4.0/5.0
- **Volume médio:** 79 tickets/dia



## Estrutura Organizacional

### Departamentos e Distribuição

1. **Atendimento Comercial** (12 agentes) - Vendas e renovações
2. **Suporte Técnico** (8 agentes) - Problemas técnicos complexos
3. **Financeiro** (6 agentes) - Cobranças e pagamentos
4. **Onboarding** (5 agentes) - Implementação de clientes
5. **Relacionamento** (4 agentes) - Gestão de contas

### Características Realistas Implementadas

- ☒ Sazonalidade (mais tickets em dias úteis)
- ☒ Variação por horário (picos no comercial)
- ☒ Correlação prioridade vs tempo de resolução
- ☒ Performance baseada em experiência do agente
- ☒ Distribuição natural de status e satisfação



## Arquivos Entregues

### 1. Dados Principais

- `dados_dashboard_atendimento.json` (15MB)
- Estrutura completa com todos os dados
- Formato JSON para fácil integração
- Encoding UTF-8 para caracteres especiais

### 2. Documentação

- `documentacao_dados_dashboard.md` + PDF
- Estrutura detalhada dos dados

- Exemplos de uso e filtros
- Sugestões de visualizações

### 3. Scripts de Geração

- `gerar_dados_dashboard.py`
- Script completo para regenerar dados
- Configurável e extensível
- Comentado e documentado

### 4. Exemplos Práticos

- `exemplo_visualizacoes.py`
- 4 dashboards interativos em HTML
- Gráficos com Plotly
- Análises executiva, temporal e departamental

### 5. Visualizações Geradas

- `dashboard_executivo.html` - KPIs principais
- `analise_agentes.html` - Performance individual
- `analise_temporal.html` - Tendências e sazonalidade
- `analise_departamental.html` - Comparativo entre áreas



## Como Usar os Dados

### Carregamento Básico (JavaScript)

```
// Carregar dados
fetch('dados_dashboard_atendimento.json')
  .then(response => response.json())
  .then(dados => {
    console.log('Total de tickets:', dados.resumo_geral.total_tickets);
    console.log('Agentes ativos:', dados.resumo_geral.agentes_ativos);
  });
```

### Carregamento em Python

```
import json
import pandas as pd

# Carregar dados
with open('dados_dashboard_atendimento.json', 'r', encoding='utf-8') as f:
    dados = json.load(f)

# Converter para DataFrames
df_tickets = pd.DataFrame(dados['tickets'])
df_agentes = pd.DataFrame(dados['agentes'])
df_metricas = pd.DataFrame(dados['metricas_agentes'])
```

## Exemplos de Análises

### 1. KPIs Principais

```
resumo = dados['resumo_geral']
print(f"Taxa de Resolução: {resumo['taxa_resolucao_geral']:.1f}%")
print(f"Satisfação Média: {resumo['satisfacao_geral']:.1f}/5.0")
print(f"Tempo Médio: {resumo['tempo_medio_resolucao_geral']:.0f} min")
```

### 2. Top Performers

```
# Top 5 agentes por satisfação
top_agentes = sorted(dados['metricas_agentes'],
                    key=lambda x: x['satisfacao_media'],
                    reverse=True)[:5]

for agente in top_agentes:
    print(f"{agente['agente_nome']}: {agente['satisfacao_media']:.1f}")
```

### 3. Análise Temporal

```
# Volume por mês
volume_mensal = dados['dados_volume']['volume_mensal']
for mes, dados_mes in volume_mensal.items():
    print(f"{mes}: {dados_mes['total_tickets']} tickets")
```

### 4. Performance Departamental

```
# Ranking de departamentos
dept_ranking = sorted(dados['metricas_departamentos'],
                    key=lambda x: x['satisfacao_media'],
                    reverse=True)

for dept in dept_ranking:
    print(f"{dept['departamento']}: {dept['satisfacao_media']:.1f} ★")
```



## Métricas Disponíveis

### Por Agente

- Volume de tickets (total, resolvidos, abertos)
- Taxa de resolução e tempo médio
- Satisfação média e cumprimento de SLA
- Produtividade diária e meta atingida
- Distribuição por canal e prioridade

### Por Departamento

- Volumes totais e taxas de resolução
- Tempos médios e satisfação
- Distribuição por status e prioridade
- Comparativos de performance

## Temporais

- Volume diário com sazonalidade
- Agregações mensais
- Tendências e padrões
- Picos e vales de demanda

## Sugestões de Dashboards

---

### 1. Dashboard Executivo

- Scorecard com KPIs principais
- Gráfico de tendência mensal
- Ranking de departamentos
- Alertas de performance

### 2. Dashboard Operacional

- Volume em tempo real
- Fila de tickets por status
- Performance individual dos agentes
- Distribuição por canal

### 3. Dashboard Analítico

- Correlações entre métricas
- Análise de sazonalidade
- Previsões de demanda
- Análise de satisfação

### 4. Dashboard Gerencial

- Performance por equipe
- Cumprimento de metas
- Análise de produtividade
- ROI do atendimento

## Personalização

---

### Regenerar Dados

```
python gerar_dados_dashboard.py
```

### Modificar Parâmetros

- Alterar período no script
- Ajustar número de agentes
- Modificar distribuição de departamentos
- Customizar tipos de tickets

### Estender Funcionalidades

- Adicionar novos campos

- Criar métricas customizadas
- Implementar novos departamentos
- Incluir dados de custo

## ✓ Validação dos Dados

---

### Consistência

- ✓ Todas as referências entre tabelas são válidas
- ✓ Datas seguem ordem cronológica
- ✓ Métricas calculadas conferem com dados base
- ✓ Distribuições seguem padrões realistas

### Realismo

- ✓ Sazonalidade implementada
- ✓ Correlações lógicas entre variáveis
- ✓ Variação natural nos dados
- ✓ Padrões de comportamento humano

### Compleitude

- ✓ Todos os campos obrigatórios preenchidos
- ✓ Histórico completo do período
- ✓ Métricas agregadas calculadas
- ✓ Documentação completa

## 📞 Próximos Passos

---

1. **Integração:** Importar dados no sistema de dashboard
2. **Customização:** Adaptar visualizações às necessidades
3. **Automação:** Configurar atualizações automáticas
4. **Treinamento:** Capacitar equipe no uso do dashboard
5. **Monitoramento:** Estabelecer rotinas de acompanhamento

---

**Data de Geração:** 07/08/2024

**Versão:** 1.0

**Formato:** JSON estruturado

**Tamanho:** ~15MB

**Período:** 01/02/2024 a 07/08/2024