

■ Dashboard de Ocorrências - Seguradora

Demonstração e Manual de Uso

■■■ IMPORTANTE

Este dashboard é uma aplicação Python/Streamlit que NÃO pode ser acessada diretamente pelo link do GitHub. É necessário fazer um fork do repositório e executar localmente conforme instruções abaixo.

■ Como Acessar o Dashboard

Passo 1: Fazer Fork do Repositório

1. Acesse: https://github.com/ru-fagundes/Projeto_Seguradora
2. Clique no botão 'Fork' no canto superior direito
3. O repositório será copiado para sua conta GitHub

Passo 2: Clonar o Repositório

```
git clone https://github.com/SEU_USUARIO/Projeto_Seguradora.git  
cd Projeto_Seguradora
```

Passo 3: Instalar Dependências

```
pip install -r requirements.txt
```

Dependências instaladas: pandas, numpy, matplotlib, seaborn, plotly, streamlit, scikit-learn

Passo 4: Executar o Dashboard

```
streamlit run src/dashboard_streamlit.py
```

Passo 5: Acessar no Navegador

O dashboard será aberto automaticamente em: <http://localhost:8501>

■ Interface do Dashboard

■ Barra Lateral - Filtros

Filtro	Opções Disponíveis
Período	Seletor de intervalo de datas (01/01/2024 - 31/12/2024)
Tipo de Problema	Todos, Atraso no Atendimento, Cobrança Indevida, Erro de Sistema, Falta de Comunicação, Negativ
Canal de Entrada	Todos, E-mail, Telefone, Presencial, Chat Online, App Móvel
Classificação	Todos, Urgente, Alta, Média, Baixa

■ Métricas Principais (KPIs)

KPI	Valor Exemplo
■ Total de Ocorrências	500
■ Tempo Médio (dias)	6.5
■ Casos Críticos	120
■ Taxa de Reincidência	15.2%
■ SLA Cumprido	78.4%

■ Visualizações Interativas (10 Gráficos)

1. Distribuição por Tipo de Problema

Tipo: Gráfico de Barras

Descrição: Mostra a quantidade de ocorrências por tipo. Exemplo: Negativa de Cobertura (145), Atraso no Atendimento (125), Cobrança Indevida (98)

2. Distribuição por Canal de Entrada

Tipo: Gráfico de Pizza

Descrição: Proporção de cada canal. Exemplo: E-mail (35%), Telefone (28%), Chat Online (20%), Presencial (12%), App Móvel (5%)

3. Tempo Médio por Classificação

Tipo: Barra Horizontal

Descrição: Tempo de resolução por prioridade. Urgente: 12.5 dias, Alta: 8.3 dias, Média: 5.2 dias, Baixa: 3.1 dias

4. Distribuição por Setor Responsável

Tipo: Gráfico de Barras

Descrição: Casos por setor. Atendimento (180), Financeiro (145), TI (95), Médico (80)

5. Evolução Temporal

Tipo: Gráfico de Área

Descrição: Tendência mensal de ocorrências ao longo de 2024. Identifica picos e vales

6. Heatmap Mensal

Tipo: Mapa de Calor

Descrição: Cruzamento de meses x tipos de problema. Cores indicam intensidade (azul=baixo, vermelho=alto)

7. Estatísticas Gerais

Tipo: Tabela

Descrição: Métricas consolidadas: média, mediana, desvio padrão, mín/máx, distribuição por prioridade

8. Reincidente por Tipo

Tipo: Barras Agrupadas

Descrição: Compara primeira ocorrência vs reincidente por tipo de problema

9. Tempo por Canal

Tipo: Barras Horizontal

Descrição: Eficiência de cada canal. App Móvel (4.2 dias) é o mais rápido, Presencial (8.1 dias) o mais lento

10. Tabela de Dados Detalhados

Tipo: Data Table

Descrição: 500 registros com scroll, paginação, ordenação. Colunas: Data, Tipo, Tempo, Canal, Classificação, Setor, Reincidente

■ Recursos Interativos

Filtros Dinâmicos

- Todos os gráficos atualizam automaticamente ao alterar filtros
- KPIs recalculam em tempo real
- Tabela de dados reflete seleção atual

Gráficos Plotly (Hover sobre os gráficos para ver)

- **Zoom:** Scroll do mouse ou pinça (touch)
- **Pan:** Arrastar com mouse
- **Hover:** Tooltip com detalhes ao passar o mouse
- **Download:** Botão para salvar gráfico como PNG
- **Reset:** Botão para restaurar visualização original

■ Exportação de Dados

Botão 'Download CSV - Dados Filtrados' localizado abaixo da tabela permite exportar os dados atualmente visíveis em formato CSV para análises externas (Excel, Python, R, etc.).

■ Design e UX

Paleta de Cores: Primary #FF4B4B (vermelho), Background #FFFFFF (branco), Secondary #F0F2F6 (cinza claro)

Tipografia: Sans Serif com títulos em negrito

Layout: Responsivo, adapta a diferentes tamanhos de tela

Performance: Carregamento < 2s, atualização de filtros < 0.5s

■ Casos de Uso

Para Gestores

1. Identificar tipos de problema mais frequentes
2. Analisar eficiência por canal de entrada
3. Monitorar tempo médio de resolução
4. Avaliar taxa de reincidência
5. Tomar decisões baseadas em dados reais

Para Analistas

1. Explorar tendências temporais e sazonalidade
2. Correlacionar variáveis (canal x tempo, tipo x reincidência)
3. Exportar dados para análises externas
4. Identificar gargalos operacionais

Para Equipes Operacionais

1. Visualizar volume de trabalho por setor
2. Priorizar casos urgentes e críticos
3. Acompanhar métricas de SLA
4. Comunicar resultados com stakeholders

■ Solução de Problemas

Dashboard não abre

```
# Verifique se a porta 8501 está livre  
netstat -ano | findstr :8501  
# Execute novamente  
streamlit run src/dashboard_streamlit.py
```

Erro de dependências

```
pip install --upgrade -r requirements.txt
```

Dados não carregam

```
# Verifique se o arquivo existe  
ls data/processed/dados_ocorrencias_formatado.csv
```

■■ Tecnologias Utilizadas

Tecnologia	Versão	Função
Python	3.8+	Linguagem principal
Streamlit	1.28.0+	Framework web
Plotly	5.14.0+	Gráficos interativos
Pandas	2.0.0+	Análise de dados
Scikit-learn	1.3.0+	Machine Learning
NumPy	1.24.0+	Computação numérica

■ Contato

Desenvolvedora: Rubia Fagundes

E-mail: rubiafagundes_ds@outlook.com

LinkedIn: <https://www.linkedin.com/in/rubiafagundes>

GitHub: https://github.com/ru-fagundes/Projeto_Seguradora

Data do Documento: 06/12/2025

Versão do Dashboard: 1.0.0