Table of Contents

# 🎛️ CÓRTEX BI - Guia do Dashboard Administrativo

**Versão:** 2.0  
**Data:** Outubro 2025  
**Desenvolvido em parceria com:** Manus AI

## 📋 Visão Geral

O **Dashboard Administrativo** do CÓRTEX BI é uma interface visual completa que permite gerenciar todo o sistema sem precisar editar arquivos de configuração manualmente. Com ele, você pode:

✅ Monitorar o sistema em tempo real  
✅ Configurar detecção de anomalias visualmente  
✅ Gerenciar fontes de dados  
✅ Treinar modelos de ML  
✅ Configurar alertas  
✅ Visualizar logs e atividades

## 📁 Localização dos Arquivos

Os arquivos do dashboard estão organizados assim:

cortex-bi/  
├── templates/  
│ └── admin/  
│ ├── dashboard.html # ⭐ Dashboard principal  
│ └── anomaly\_config.html # Configurador visual de métricas  
│  
└── static/  
 ├── css/  
 ├── js/  
 │ └── admin-dashboard.js # JavaScript do dashboard  
 └── images/

## 🚀 Como Acessar

### Método 1: Via Navegador

1. Certifique-se de que o CÓRTEX BI está rodando:

./scripts/start\_ai.sh # Linux/macOS  
# ou  
.\scripts\start\_ai.bat # Windows

1. Abra seu navegador e acesse:

http://localhost:5000/admin/dashboard

### Método 2: Configurar Rota no Flask

Se a rota ainda não existe, adicione ao arquivo main\_ai.py:

from flask import Flask, render\_template, send\_from\_directory  
  
app = Flask(\_\_name\_\_,   
 template\_folder='templates',  
 static\_folder='static')  
  
@app.route('/admin/dashboard')  
def admin\_dashboard():  
 """Dashboard administrativo"""  
 return render\_template('admin/dashboard.html')  
  
@app.route('/admin/anomaly-config')  
def anomaly\_config():  
 """Configurador de métricas de anomalia"""  
 return render\_template('admin/anomaly\_config.html')  
  
@app.route('/static/<path:filename>')  
def serve\_static(filename):  
 """Servir arquivos estáticos"""  
 return send\_from\_directory('static', filename)

## 🎨 Funcionalidades do Dashboard

### 1. Visão Geral (Overview)

**O que você vê:** - 📊 **Cards de Estatísticas**: Análises executadas, usuários ativos, anomalias detectadas, taxa de sucesso - 📈 **Gráfico de Análises**: Visualização das análises dos últimos 7 dias - 🤖 **Status dos Agentes**: Verificação em tempo real de todos os 8 agentes de IA - 📋 **Atividade Recente**: Últimas ações realizadas no sistema

**Como usar:** - Clique em “Atualizar” para recarregar os dados - Os cards mostram comparação com períodos anteriores - O gráfico é interativo (passe o mouse para ver detalhes)

### 2. Detecção de Anomalias

**O que você pode fazer:**

#### 2.1 Visualizar Métricas Configuradas

* Lista todas as métricas que estão sendo monitoradas
* Mostra status (ativa/inativa), fonte de dados, intervalo
* Permite editar ou excluir métricas

#### 2.2 Adicionar Nova Métrica

1. Clique no botão **“Nova Métrica”**
2. Você será redirecionado para o configurador visual
3. Preencha os campos:
   * **Informações Básicas**: ID, nome, descrição
   * **Fonte de Dados**: Escolha entre SQL, CSV, API, etc.
   * **Detecção**: Configure sensibilidade e thresholds
   * **Alertas**: Defina severidade e destinatários
   * **Monitoramento**: Defina intervalo e horários
4. Clique em **“Salvar e Ativar Monitoramento”**

#### 2.3 Ver Anomalias Detectadas

* Tabela com todas as anomalias recentes
* Filtro por severidade (crítica, alta, média, baixa)
* Detalhes de cada anomalia (valor, score, timestamp)

### 3. Modelos de Machine Learning

**O que você pode fazer:** - Ver status de todos os 5 modelos de ML - Verificar quando foi o último treinamento - Ver métricas de performance (acurácia, amostras) - Treinar ou retreinar modelos individualmente - Treinar todos os modelos de uma vez

**Modelos disponíveis:** 1. **Preditor de Análises** - Prevê tipo de análise baseado no contexto 2. **Classificador de Qualidade** - Avalia qualidade das análises 3. **Detector de Anomalias** - Identifica padrões anormais 4. **Clusterizador de Usuários** - Agrupa usuários por comportamento 5. **Preditor de Performance** - Prevê tempo de execução

### 4. Fontes de Dados

**O que você pode fazer:** - Listar todas as fontes de dados configuradas - Ver status de conexão (conectado/desconectado) - Testar conexão com cada fonte - Adicionar novas fontes de dados - Editar configurações existentes

**Tipos suportados:** - SQL Server - Arquivos CSV - Arquivos Excel - APIs REST - Power BI - Métricas internas

### 5. Templates PPTX

**O que você pode fazer:** - Visualizar templates disponíveis - Upload de novos templates - Ativar/desativar templates - Definir template padrão - Testar geração de apresentação

### 6. Usuários

**O que você pode fazer:** - Listar todos os usuários do sistema - Ver atividade de cada usuário - Gerenciar permissões - Bloquear/desbloquear usuários

### 7. API Keys

**O que você pode fazer:** - Gerar novas API keys - Listar keys ativas - Revogar keys - Ver uso de cada key - Definir limites de rate

### 8. Alertas

**O que você pode fazer:** - Configurar canais de alerta (email, Teams, webhook) - Definir templates de mensagem - Testar envio de alertas - Ver histórico de alertas enviados

### 9. Logs

**O que você pode fazer:** - Visualizar logs do sistema em tempo real - Filtrar por nível (info, warning, error) - Buscar por termos específicos - Exportar logs - Limpar logs antigos

### 10. Configurações

**O que você pode fazer:** - Configurar variáveis de ambiente - Ajustar parâmetros do sistema - Configurar integrações (M365, Power BI) - Backup e restore - Atualização do sistema

## 🎯 Fluxo de Trabalho Típico

### Configurar Detecção de Anomalias (Passo a Passo)

**Cenário:** Você quer monitorar vendas diárias e receber alerta se caírem mais de 15%

#### Passo 1: Acessar o Dashboard

http://localhost:5000/admin/dashboard

#### Passo 2: Ir para Detecção de Anomalias

* Clique em “Detecção de Anomalias” no menu lateral

#### Passo 3: Clicar em “Nova Métrica”

* Botão no canto superior direito

#### Passo 4: Preencher Informações Básicas

ID da Métrica: vendas\_diarias  
Nome: Vendas Diárias  
Descrição: Monitoramento de vendas totais por dia  
☑️ Métrica habilitada

#### Passo 5: Configurar Fonte de Dados

**Se usar SQL Server:**

Tipo de Fonte: SQL Server  
Conexão SQL: sql\_principal  
Query SQL:   
 SELECT SUM(valor\_venda) as total   
 FROM vendas   
 WHERE CAST(data\_venda AS DATE) = CAST(GETDATE() AS DATE)  
Coluna de Valor: total  
Query Histórica:  
 SELECT CAST(data\_venda AS DATE) as data, SUM(valor\_venda) as total  
 FROM vendas   
 WHERE data\_venda >= DATEADD(day, -30, GETDATE())  
 GROUP BY CAST(data\_venda AS DATE)

**Se usar CSV:**

Tipo de Fonte: Arquivo CSV  
Caminho: data/raw/vendas\_diarias.csv  
Coluna de Valor: valor\_total  
Coluna de Data: data  
Delimitador: , (vírgula)  
Encoding: UTF-8

#### Passo 6: Configurar Detecção

Algoritmo: Isolation Forest  
Contaminação: 0.1 (10%)  
Threshold: -0.5  
Amostras Mínimas: 30

#### Passo 7: Configurar Alertas

Severidade: Alta  
Tipo de Threshold: Percentual (%)  
Valor do Threshold: 15  
☑️ Email  
☐ Microsoft Teams  
☐ Webhook  
Destinatários: gerente@empresa.com, diretor@empresa.com  
Cooldown: 60 minutos

#### Passo 8: Configurar Monitoramento

Intervalo: 1 hora  
Horário Ativo: 24 horas  
Dias Ativos: ☑️ Seg ☑️ Ter ☑️ Qua ☑️ Qui ☑️ Sex ☑️ Sáb ☑️ Dom

#### Passo 9: Salvar e Ativar

* Clique em **“Salvar e Ativar Monitoramento”**

#### Passo 10: Verificar

* Volte para o dashboard
* A métrica aparecerá na lista como “Ativa”
* O sistema começará a monitorar automaticamente

## 💡 Dicas e Boas Práticas

### 1. Organize suas Métricas

* Use IDs descritivos (ex: vendas\_diarias, não metrica1)
* Agrupe métricas relacionadas com prefixos (ex: vendas\_\*, custos\_\*)
* Documente cada métrica na descrição

### 2. Configure Alertas Inteligentes

* Use severidade apropriada (não tudo como “crítico”)
* Configure cooldown para evitar spam de alertas
* Envie alertas apenas para quem precisa agir

### 3. Ajuste a Sensibilidade

* Comece com valores padrão (contamination=0.1, threshold=-0.5)
* Se muitos falsos positivos: aumente threshold para -0.7
* Se não detecta anomalias óbvias: diminua threshold para -0.3

### 4. Monitore a Performance

* Não configure intervalos muito curtos (< 5 minutos) para queries pesadas
* Use horários ativos para economizar recursos
* Desative métricas que não são mais necessárias

### 5. Teste Antes de Ativar

* Use os botões “Testar Query” e “Testar API”
* Verifique se os dados históricos estão disponíveis
* Envie um alerta de teste antes de ativar

## 🔧 Personalização

### Alterar Cores e Estilo

Edite o arquivo templates/admin/dashboard.html na seção <style>:

:root {  
 --primary-color: #1e40af; /\* Azul principal \*/  
 --secondary-color: #10b981; /\* Verde secundário \*/  
 --danger-color: #dc3545; /\* Vermelho \*/  
 --warning-color: #ffc107; /\* Amarelo \*/  
}

### Adicionar Nova Seção no Menu

Edite templates/admin/dashboard.html:

<ul class="sidebar-menu">  
 <!-- Menus existentes... -->  
   
 <!-- Novo menu -->  
 <li>  
 <a href="#" onclick="showSection('minha-secao')">  
 <i class="fas fa-star"></i> Minha Seção  
 </a>  
 </li>  
</ul>

Adicione a seção no conteúdo:

<div id="section-minha-secao" class="section-content" style="display: none;">  
 <div class="content-section">  
 <div class="section-header">  
 <h2>Minha Seção Customizada</h2>  
 </div>  
 <p>Conteúdo aqui...</p>  
 </div>  
</div>

### Adicionar Novo Gráfico

Use Chart.js no arquivo static/js/admin-dashboard.js:

function createMyChart() {  
 const ctx = document.getElementById('myChart');  
 new Chart(ctx, {  
 type: 'bar',  
 data: {  
 labels: ['Jan', 'Fev', 'Mar'],  
 datasets: [{  
 label: 'Vendas',  
 data: [12, 19, 3],  
 backgroundColor: '#1e40af'  
 }]  
 }  
 });  
}

## 🐛 Troubleshooting

### Dashboard não carrega

**Problema:** Página em branco ou erro 404

**Soluções:** 1. Verificar se o servidor está rodando:

curl http://localhost:5000/health

1. Verificar se as rotas estão configuradas no main\_ai.py
2. Verificar logs:

tail -f logs/cortexbi.log

### Dados não aparecem

**Problema:** Dashboard carrega mas não mostra dados

**Soluções:** 1. Abrir console do navegador (F12) 2. Verificar erros de JavaScript 3. Verificar se a API Key está configurada:

localStorage.setItem('cortex\_api\_key', 'sua-api-key');

1. Testar endpoints da API diretamente:

curl -H "X-API-Key: sua-key" http://localhost:5000/admin/stats

### Configuração não salva

**Problema:** Ao clicar em “Salvar”, nada acontece

**Soluções:** 1. Verificar console do navegador (F12) 2. Verificar se todos os campos obrigatórios estão preenchidos 3. Testar endpoint da API:

curl -X POST http://localhost:5000/anomaly/metrics/create \  
 -H "Content-Type: application/json" \  
 -H "X-API-Key: sua-key" \  
 -d '{"id": "teste", "name": "Teste"}'

## 📱 Acesso Mobile

O dashboard é responsivo e funciona em tablets e smartphones:

* **Tablets**: Interface completa
* **Smartphones**: Menu lateral recolhe automaticamente
* **Touch**: Todos os botões e gráficos são touch-friendly

## 🔐 Segurança

### Autenticação

Por padrão, o dashboard usa API Key. Para adicionar autenticação:

1. Crie uma página de login
2. Implemente sessões no Flask
3. Proteja as rotas administrativas

Exemplo:

from flask import session, redirect, url\_for  
from functools import wraps  
  
def login\_required(f):  
 @wraps(f)  
 def decorated\_function(\*args, \*\*kwargs):  
 if 'user\_id' not in session:  
 return redirect(url\_for('login'))  
 return f(\*args, \*\*kwargs)  
 return decorated\_function  
  
@app.route('/admin/dashboard')  
@login\_required  
def admin\_dashboard():  
 return render\_template('admin/dashboard.html')

### HTTPS

Para produção, sempre use HTTPS:

# Usando nginx como proxy reverso  
sudo apt install nginx  
sudo nano /etc/nginx/sites-available/cortexbi

Configuração nginx:

server {  
 listen 443 ssl;  
 server\_name cortexbi.empresa.com;  
   
 ssl\_certificate /etc/ssl/certs/cortexbi.crt;  
 ssl\_certificate\_key /etc/ssl/private/cortexbi.key;  
   
 location / {  
 proxy\_pass http://localhost:5000;  
 proxy\_set\_header Host $host;  
 proxy\_set\_header X-Real-IP $remote\_addr;  
 }  
}

## 📊 Resumo

**Onde está o dashboard:**

/home/ubuntu/cortex-bi-repository/templates/admin/dashboard.html

**Como acessar:**

http://localhost:5000/admin/dashboard

**Configurador visual de anomalias:**

http://localhost:5000/admin/anomaly-config

**Principais funcionalidades:** 1. ✅ Monitoramento em tempo real 2. ✅ Configuração visual de anomalias 3. ✅ Gerenciamento de modelos ML 4. ✅ Administração de fontes de dados 5. ✅ Visualização de logs e alertas

**Vantagens:** - ✅ Não precisa editar JSON manualmente - ✅ Interface intuitiva e moderna - ✅ Validação de campos em tempo real - ✅ Testes integrados - ✅ Visualização de dados em gráficos

**CÓRTEX BI v2.0** - *Cognitive Operations & Real-Time EXpert Business Intelligence*  
Desenvolvido em parceria com **Manus AI** | Outubro 2025

🎛️ **Dashboard Administrativo - Gerencie tudo visualmente!**