

Reporter - Documentação Completa para Marketing

Versão: 4.0.0 | **Última Atualização:** Janeiro 2026 **Desenvolvido por:** Lerian Studio **Repositório:** <https://github.com/LerianStudio/reporter>

1. O QUE É O REPORTER

1.1 Definição

O **Reporter** é uma plataforma enterprise de geração de relatórios dinâmicos desenvolvida pela Lerian Studio. É um serviço de missão crítica que permite às organizações criar, gerenciar e gerar relatórios customizáveis a partir de templates predefinidos, com foco especial em **compliance regulatório** para o mercado financeiro brasileiro.

1.2 Proposta de Valor

"Transforme dados financeiros complexos em relatórios regulatórios precisos, automatizados e auditáveis."

O Reporter resolve um dos maiores desafios das instituições financeiras brasileiras: a geração de relatórios obrigatórios para órgãos reguladores como **BACEN** (Banco Central do Brasil) e **RFB** (Receita Federal do Brasil), eliminando processos manuais, reduzindo erros e garantindo conformidade total.

1.3 Missão

Capacitar usuários não técnicos a criar relatórios orientados a dados de qualquer fonte, mantendo padrões enterprise de segurança, compliance e escalabilidade.

2. PARA QUE SERVE

2.1 Casos de Uso Principais

A) Compliance Regulatório (Principal)

- **BACEN CADOC**: Geração automática de balancetes (4010) e balanços patrimoniais (4016)
- **BACEN APIX**: Estatísticas de transações PIX
- **RFB e-Financeira**: Declarações de eventos financeiros (abertura, fechamento, movimentações)
- **RFB DIMP**: Declaração de Movimentação Patrimonial

B) Relatórios Financeiros Operacionais

- Demonstrativos financeiros customizados
- Relatórios de transações e operações
- Extratos de contas e saldos
- Análises de portfólio e carteiras

C) Business Intelligence

- Relatórios analíticos com agregações (soma, média, contagem)
- Dashboards exportáveis em múltiplos formatos
- Análises de tendências e comparativos

D) Auditoria e Controle Interno

- Trilhas de auditoria
- Relatórios de reconciliação
- Documentação de compliance

2.2 Problemas que Resolve

Problema	Solução Reporter
Relatórios manuais propensos a erros	Templates validados com extração automática de dados
Multas por não-conformidade regulatória	Validação em 3 etapas (3-Gate) para zero erros

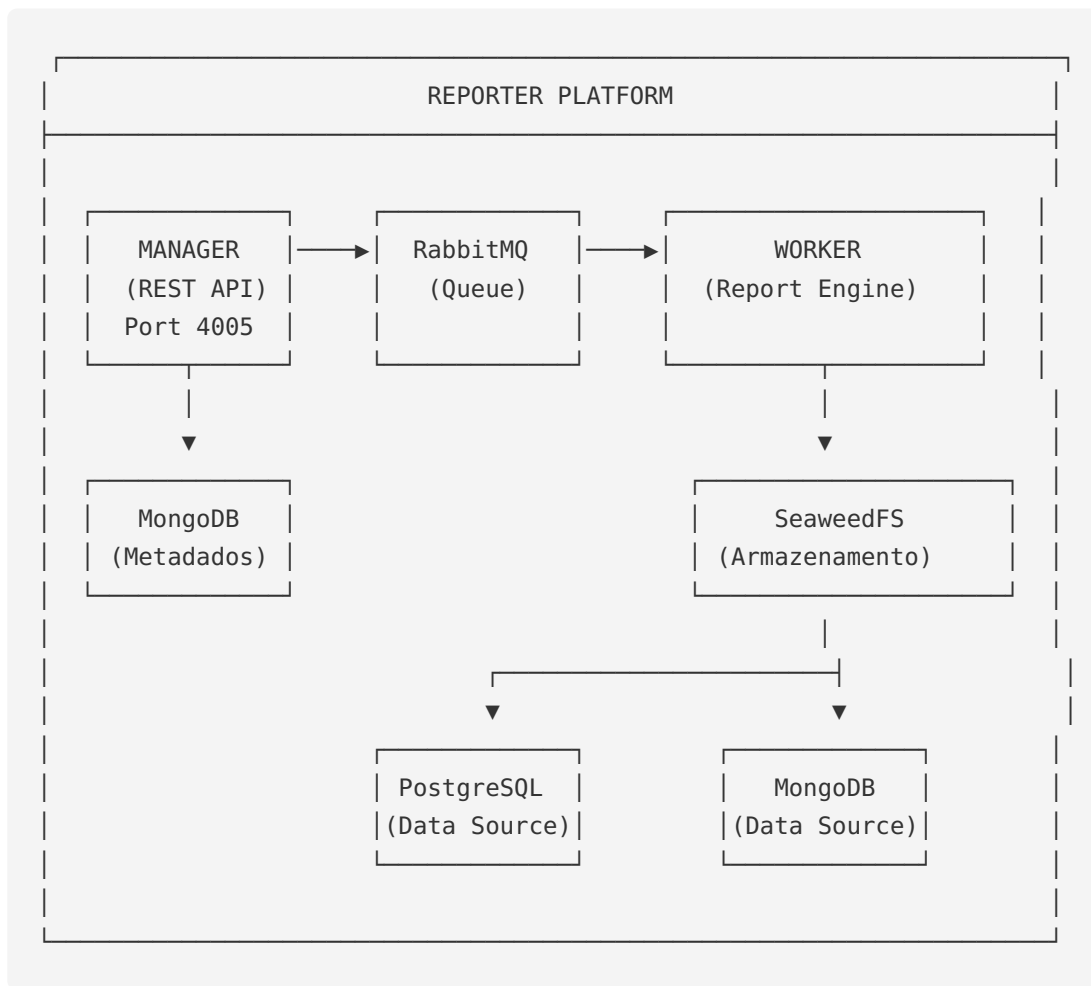
Problema	Solução Reporter
Tempo excessivo para gerar relatórios	Processamento assíncrono em segundos/ minutos
Dificuldade em integrar múltiplas fontes	Suporte nativo a PostgreSQL e MongoDB
Falta de flexibilidade nos formatos	6 formatos de saída (HTML, PDF, XML, CSV, JSON, TXT)
Escalabilidade limitada	Arquitetura de microsserviços com workers distribuídos

3. ARQUITETURA E TECNOLOGIA

3.1 Stack Tecnológico

Camada	Tecnologia	Versão
Linguagem	Go	1.25
Framework Web	Fiber	v2.52.9
Template Engine	Pongo2	v6 (sintaxe Django-like)
Banco Metadados	MongoDB	v1.17
Fontes de Dados	PostgreSQL + MongoDB	pgx v5
Fila de Mensagens	RabbitMQ	amqp091-go v1.10
Armazenamento	SeaweedFS	Sistema de arquivos distribuído
Cache	Redis/Valkey	go-redis v9
Geração PDF	ChromeDP	Headless Chrome
Observabilidade	OpenTelemetry	v1.38.0
Autenticação	JWT via lib-auth	v2.2.0

3.2 Arquitetura de Componentes



3.3 Componentes Principais

Manager (Porta 4005)

- API RESTful para todas as operações
- CRUD completo de templates e relatórios
- Descoberta automática de fontes de dados
- Validação de campos contra schemas
- Documentação Swagger integrada

Worker

- Consumidor de mensagens RabbitMQ
- Motor de renderização de templates (Pongo2)
- Geração de PDFs com Chrome headless
- Queries dinâmicas contra múltiplas bases
- Pool de workers configurável

Infraestrutura

- Docker Compose para orquestração
- SeaweedFS para armazenamento distribuído
- Circuit breaker para resiliência
- Health checks automáticos

3.4 Integração com Midaz

O Reporter é projetado para integrar nativamente com o **Midaz**, a plataforma financeira open-source da Lerian Studio:

Fonte de Dados	Conteúdo
midaz_onboarding	Organizações, ledgers, contas, entidades
midaz_transaction	Transações, operações, saldos
midaz_pix	Transações PIX específicas
midaz_compliance	Dados de compliance regulatório

4. FUNCIONALIDADES PRINCIPAIS

4.1 Sistema de Templates

Formatos Suportados

Formato	Extensão Saída	Uso Principal
HTML	.html	Visualização web, relatórios interativos
PDF	.pdf	Documentos impressos, arquivamento
XML	.xml	Compliance regulatório (BACEN, RFB)
CSV	.csv	Análise em planilhas
JSON	.json	Integração com sistemas
TXT	.txt	Arquivos de largura fixa (legado)

Sintaxe de Templates (Pongo2/Django-like)

```
<!-- Placeholders -->
{{ database.table.field }}

<!-- Loops -->
{% for item in collection %}
    {{ item.name }}
{% endfor %}

<!-- Condicionais -->
{% if account.status == "active" %}
    Conta Ativa
{% endif %}

<!-- Agregações -->
{% sum_by transaction.operation by "amount" if status == "completed" %}
{% count_by transaction.operation %}
{% avg_by transaction.operation by "amount" %}
```

Filtros Disponíveis

Filtro	Descrição	Exemplo
floatformat:N	N casas decimais	{{ 1234.5\ floatformat:2 }} → 1234.50
date:"formato"	Formatação de data	{{ date\ date:"d/m/Y" }} → 15/01/2026
slice:":N"	Extração de substring	{{ cnpj\ slice:":8" }} → primeiros 8 dígitos
upper / lower	Maiúsculas/minúsculas	{{ text\ upper }}
ljust:N / rjust:N	Padding esquerda/direita	Alinhamento de texto
percent_of	Cálculo percentual	{{ value\ percent_of: total }}

4.2 Sistema de Filtros Avançados

O Reporter v4.0 introduziu um sistema de filtros avançados para consultas complexas:

```
{
  "filters": {
    "database": {
      "table": {
        "field": {
          "eq": ["valor"],           // Igual
          "gt": [100],             // Maior que
          "gte": ["2025-01-01"],   // Maior ou igual
          "lt": [1000],            // Menor que
          "lte": ["2025-12-31"],   // Menor ou igual
          "between": [100, 1000],  // Entre valores
          "in": ["a", "b", "c"],   // Lista inclusiva
          "nin": ["x", "y"]        // Lista exclusiva
        }
      }
    }
  }
}
```

4.3 API RESTful

Endpoints Principais

Método	Endpoint	Descrição
POST	/v1/templates	Criar template (upload + metadados)
GET	/v1/templates	Listar todos os templates
GET	/v1/templates/{id}	Obter template específico
PATCH	/v1/templates/{id}	Atualizar template
DELETE	/v1/templates/{id}	Remover template
POST	/v1/reports	Criar relatório
GET	/v1/reports	Listar relatórios
GET	/v1/reports/{id}	Status do relatório

Método	Endpoint	Descrição
GET	/v1/reports/{id}/download	Download do arquivo
GET	/v1/data-sources	Listar fontes de dados

4.4 Processamento Assíncrono

1. **Requisição** → Manager recebe pedido de relatório
2. **Validação** → Verifica template e campos
3. **Enfileiramento** → Mensagem enviada ao RabbitMQ
4. **Processamento** → Worker executa queries e renderiza
5. **Armazenamento** → Arquivo salvo no SeaweedFS
6. **Conclusão** → Status atualizado para "finished"

5. COMPLIANCE REGULATÓRIO

5.1 Regulamentações Suportadas

BACEN (Banco Central do Brasil)

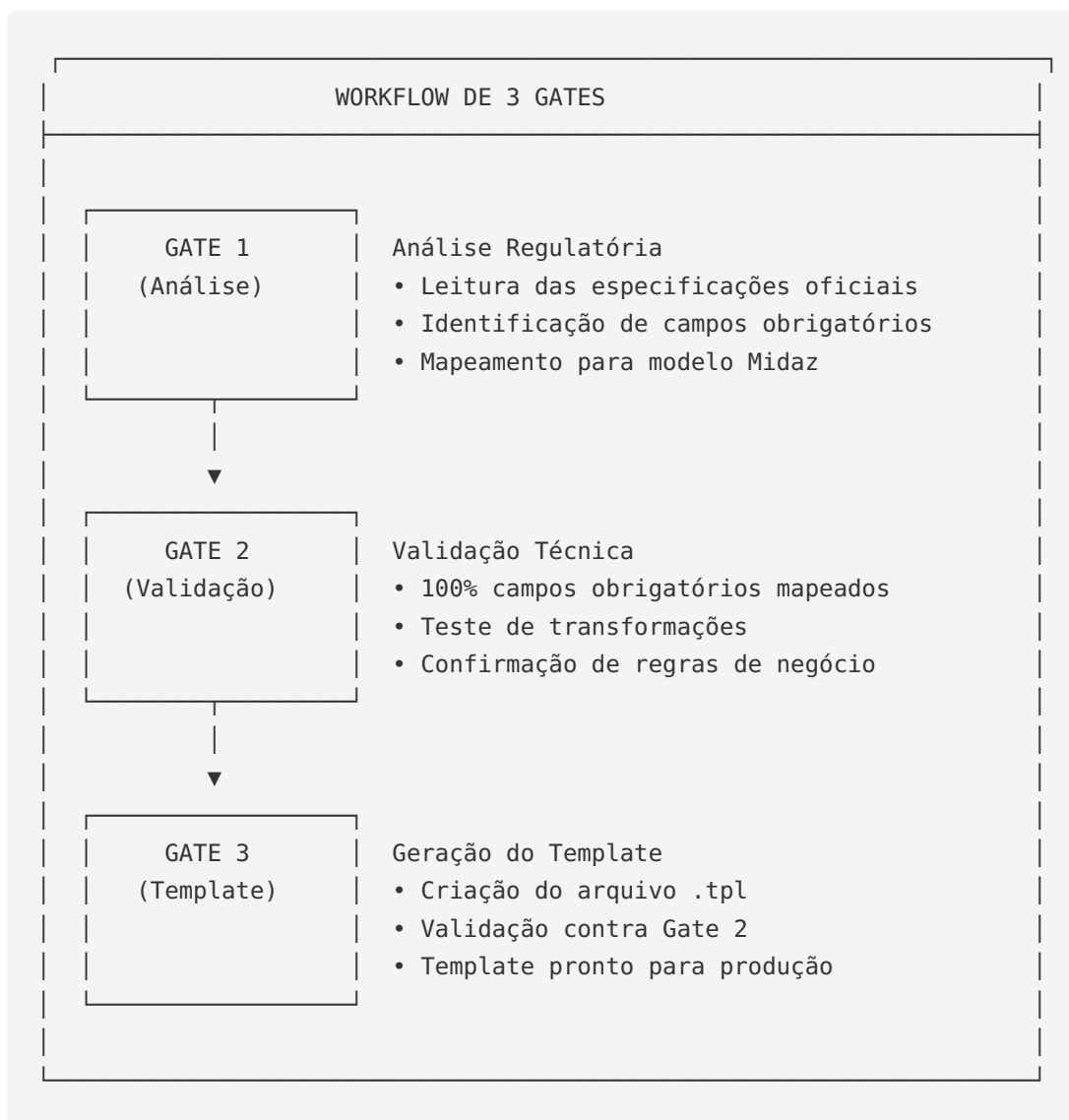
Documento	Descrição	Periodicidade
CADOC 4010	Balancete Mensal	Mensal
CADOC 4016	Balanço Patrimonial Analítico	Semestral
CADOC 4111	Operações de Câmbio	Conforme operações
APIX 001	Estatísticas PIX	Mensal
APIX 002	Contas e Transações PIX	Mensal

RFB (Receita Federal do Brasil)

Documento	Descrição
e-Financeira	Declaração de eventos financeiros
DIMP v10	Movimentação Patrimonial

5.2 Workflow de 3 Gates (Zero-Tolerance)

O Reporter implementa um processo rigoroso de validação em 3 etapas para garantir conformidade total:



5.3 Penalidades Evitadas

BACEN: Multas de R\$ 10.000 a R\$ 500.000 + sanções de licença **RFB:**
Responsabilização criminal por declarações falsas

O Reporter elimina esses riscos através de validação automática e templates pré-aprovados.

6. DIFERENCIAIS COMPETITIVOS

6.1 Tecnológicos

Diferencial	Benefício
Multi-Database Nativo	Consulta PostgreSQL e MongoDB na mesma query
Template Engine Flexível	Sintaxe Django-like familiar e poderosa
Processamento Assíncrono	Escalabilidade para milhares de relatórios simultâneos
Circuit Breaker	Resiliência contra falhas de conexão
Armazenamento Distribuído	SeaweedFS para alta disponibilidade

6.2 Funcionais

Diferencial	Benefício
6 Formatos de Saída	Flexibilidade total para qualquer necessidade
Filtros Avançados	Queries complexas sem código
Validação de Campos	Erros detectados antes da execução
Descoberta de Schema	Auto-documentação das fontes de dados
Health Checks	Monitoramento proativo de saúde

6.3 Compliance

Diferencial	Benefício
Templates Regulatórios	Conformidade BACEN/RFB out-of-the-box
Workflow 3-Gates	Zero erros em declarações
Auditoria Completa	Rastreabilidade total de operações
Multi-tenancy	Isolamento por organização

7. PECULIARIDADES E CARACTERÍSTICAS ÚNICAS

7.1 Prevenção de Segurança

- **Bloqueio de `<script>`** : Templates com tags JavaScript são rejeitados automaticamente
- **SQL Injection Prevention**: Queries parametrizadas via Squirrel
- **Isolamento de Organização**: Dados segregados por `X-Organization-Id`
- **JWT Authentication**: Integração com sistema de autenticação centralizado

7.2 Conversão Inteligente de Datas

O sistema automaticamente converte datas no formato `YYYY-MM-DD` para ranges completos: - Input: `"2025-01-15"` - Output: `2025-01-15T00:00:00Z` até `2025-01-15T23:59:59.999Z`

7.3 Pool de Geração de PDF

Workers dedicados com Chrome headless em pool gerenciado: - Configurável via `PDF_POOL_WORKERS` - Timeout configurável (`PDF_TIMEOUT_SECONDS`) - Reutilização de instâncias para performance

7.4 Dead Letter Queue (DLQ)

Mensagens que falham repetidamente são movidas para DLQ: - Previne loops infinitos de retry - Permite investigação de falhas - Garante estabilidade do sistema

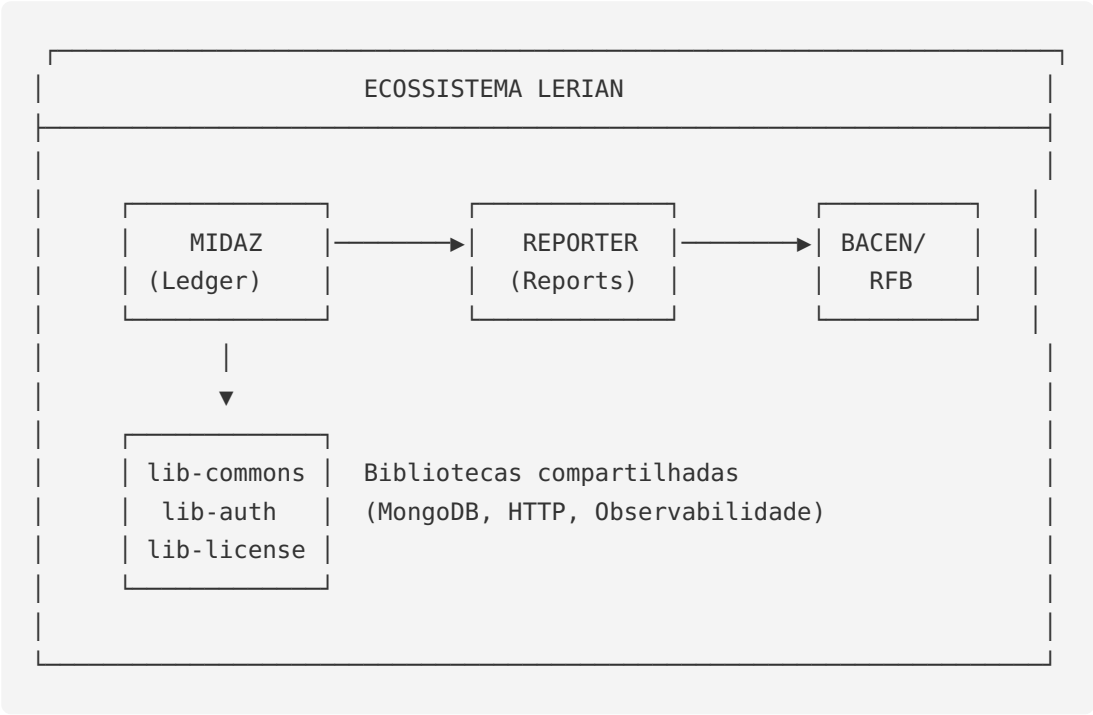
7.5 Retry com Backoff Exponencial

Conexões com bancos de dados seguem padrão enterprise: - Máximo 3 tentativas - Backoff inicial: 1 segundo - Multiplicador: 2.0x - Máximo: 10 segundos

8. ECOSSISTEMA LERIAN

8.1 Integração com Midaz

O Reporter é parte do ecossistema Lerian, integrando nativamente com:



8.2 Open Source

- **Licença:** Open source (verificar licença específica)
- **Comunidade:** Contribuições via GitHub
- **Documentação:** Swagger + Markdown + Skills integradas

9. MÉTRICAS E PERFORMANCE

9.1 Benchmarks de Performance

Operação	Tempo Esperado
Criação de template	< 500ms
Filtros simples (eq, in)	< 50ms conversão
Filtros complexos (between, múltiplos)	< 100ms conversão
Relatório HTML simples	< 2 segundos
Relatório PDF complexo	< 30 segundos

9.2 Escalabilidade

Configuração	Capacidade
Conexões PostgreSQL	25 ativas / 10 idle
Conexões MongoDB	100 pool / 10 min
Workers RabbitMQ	Configurável (padrão: 4)
PDF Workers	Configurável (padrão: 2)

9.3 Resiliência

- **Circuit Breaker:** 15 falhas para abrir
- **Recovery:** 30 segundos timeout
- **Health Check:** Intervalo de 30 segundos

10. CASOS DE USO DETALHADOS

10.1 Instituição Financeira - Relatório CADOC 4010

Cenário: Banco precisa enviar balancete mensal ao BACEN

Solução: 1. Template CADOC 4010 pré-configurado 2. Conexão com sistema core (PostgreSQL) 3. Geração automática no dia 15 de cada mês 4. Arquivo XML validado e pronto para STA

Resultado: Conformidade 100%, zero intervenção manual

10.2 Fintech - Relatório e-Financeira

Cenário: Fintech precisa declarar movimentações financeiras à RFB

Solução: 1. Template e-Financeira com todos eventos 2. Filtros por período e tipo de operação 3. Geração sob demanda ou agendada 4. XML no padrão exato da especificação

Resultado: Declarações precisas, auditoria simplificada

10.3 Empresa - Relatórios Operacionais

Cenário: Gestor financeiro precisa de relatório diário de transações

Solução: 1. Template HTML/PDF customizado 2. Filtros por data, conta, status 3. Agregações (totais, médias) 4. Distribuição automática por email

Resultado: Visibilidade total, decisões informadas

11. ROADMAP E EVOLUÇÃO

11.1 Versão Atual (v4.0.0)

- SeaweedFS para armazenamento escalável
- Sistema de filtros avançados
- Dead Letter Queue
- Circuit breaker aprimorado
- Performance otimizada com índices

11.2 Futuras Melhorias (Potencial)

- Operadores adicionais: `contains` , `starts_with` , `regex`
 - Grupos de filtros com lógica AND/OR
 - Templates e presets reutilizáveis
 - Agendamento nativo de relatórios
 - Notificações via webhook
-

12. GLOSSÁRIO

Termo	Definição
BACEN	Banco Central do Brasil
RFB	Receita Federal do Brasil
CADOC	Catálogo de Documentos do BACEN
COSIF	Plano Contábil das Instituições Reguladas
e-Financeira	Sistema de declaração de eventos financeiros

Termo	Definição
DIMP	Declaração de Informações sobre Movimentação Patrimonial
Template	Modelo de relatório com placeholders
Worker	Processo que executa geração de relatórios
Circuit Breaker	Padrão de resiliência para falhas de conexão
DLQ	Dead Letter Queue - fila para mensagens com falha
Pongo2	Engine de templates com sintaxe Django

13. MENSAGENS-CHAVE PARA MARKETING

13.1 Taglines

- *"Compliance regulatório sem código, sem erros, sem preocupações."*
- *"Do dado ao relatório BACEN em minutos."*
- *"Sua conformidade regulatória, automatizada."*
- *"Relatórios enterprise, simplicidade startup."*

13.2 Value Propositions

Para C-Level:

"Reduza custos de compliance em até 80% e elimine riscos de multas regulatórias com geração automatizada de relatórios BACEN e RFB."

Para Compliance Officers:

"Templates validados em 3 etapas garantem zero erros em declarações regulatórias. Auditoria completa de cada relatório gerado."

Para Desenvolvedores:

"API RESTful moderna, integração em horas não semanas. PostgreSQL, MongoDB, 6 formatos de saída, escalabilidade infinita."

Para Operações:

"Processe milhares de relatórios simultaneamente. Resiliência enterprise com circuit breakers e health checks automáticos."

13.3 Números que Impressionam

- **6** formatos de saída suportados
- **8** operadores de filtro avançados
- **3** gates de validação para compliance
- **<2s** tempo de geração para relatórios simples
- **100%** conformidade com especificações regulatórias
- **Zero** código necessário para criar templates

14. RECURSOS ADICIONAIS

14.1 Documentação Técnica

- Swagger UI: `http://localhost:4005/swagger/index.html`
- Guia de Templates: `/docs/features/`
- Especificações Regulatórias: `/.claude/docs/regulatory/`

14.2 Repositórios Relacionados

- **Reporter**: <https://github.com/LerianStudio/reporter>
- **Midaz**: <https://github.com/LerianStudio/midaz>
- **lib-commons**: <https://github.com/LerianStudio/lib-commons>
- **lib-auth**: <https://github.com/LerianStudio/lib-auth>

*Documento preparado para uso exclusivo da equipe de Marketing da Lerian Studio.
Para informações técnicas adicionais, consulte a documentação do desenvolvedor.*