Projeto Sistema de Farmácia

Documento Técnico – **Hardening de Produção** (Logs Estruturados, Backups, Monitoramento)

Objetivo: preparar backend e banco para operar em produção com confiabilidade, rastreabilidade e observabilidade. Este passo é pré-requisito para colocar o sistema em operação com risco controlado.

1) Entregáveis do Hardening (o que deve existir)

- Logs estruturados (JSON) com correlação por requestId.
- Métricas básicas e healthchecks (liveness/readiness).
- Monitoramento e alertas (erros, latência, disponibilidade).
- Backups automáticos do PostgreSQL + política de retenção.
- Rotina de restore testado (simulação) em ambiente de staging.
- Configuração de ambiente (ENV) e segredos fora do repositório.

2) Organização física (C:\pharma)

```
C:\pharma
  \backend
    \src
      \common
        logger.ts
        request-id.middleware.ts
        health.controller.ts
      \modules
        \audit (já existe)
    \infra
      \docker
        Dockerfile
        docker-compose.yml
  \docs
    \16_hardening_producao
```

3) Logs estruturados (padrão recomendado)

Formato recomendado: JSON por linha, com requestId para correlacionar eventos.

```
{
  "ts":"2026-02-15T12:34:56.789Z",
  "level":"info",
  "service":"pharma-backend",
  "env":"prod",
  "requestId":"a1b2c3",
  "storeId":"...",
  "userId":"...",
  "route":"POST /sales/{id}/pay",
  "msg":"payment_confirmed",
  "meta": { "saleId":"...", "amount":"123.45" }
}
```

- Incluir: ts, level, requestId, storeId (quando houver), userId (quando houver), route, msg, meta.

- Não logar dados sensíveis: senha, token completo, dados pessoais desnecessários.
- Logar erros com stack + código interno.

4) Healthchecks (produção)

- GET /health/live → retorna OK se o processo está rodando.
- GET /health/ready → testa dependências: conexão DB (select 1).
- Monitoramento deve chamar /health/ready periodicamente.

Exemplo de resposta:

```
{ "status":"ok", "db":"ok", "uptimeSec": 12345 }
```

5) Métricas mínimas e alertas

- Taxa de erro (5xx) por minuto.
- Latência p95 por rota crítica (login, pay, close cash).
- Disponibilidade (/health/ready).
- Uso de CPU/memória (container/VM).
- Conexões ao banco e erros de conexão (P1001/P1002).

6) Backups do PostgreSQL (política objetiva)

- Backup diário full + retenção (ex.: 14 a 30 dias).
- Backup incremental/WAL (se disponível) para ponto no tempo (PITR).
- Armazenar em local separado (ex.: bucket S3/Blob).
- Criptografia em repouso + acesso mínimo (IAM).

Comando base (exemplo) para backup lógico:

```
pg_dump -Fc "$DATABASE_URL" -f backup_YYYYMMDD.dump
```

Restore (exemplo):

```
pg_restore -d "$DATABASE_URL" backup_YYYYMMDD.dump
```

7) Staging + Restore testado (obrigatório)

- Criar ambiente staging (DB separado).
- 1x por semana: restaurar um backup em staging e validar: login, criar venda, fechar caixa.
- Registrar evidência (log + checklist).

8) Configuração de ambiente e segredos

- .env nunca no git; use variáveis no provedor.
- Segredos: JWT_SECRET, REFRESH_SECRET, DATABASE_URL.
- Rotação programada de segredos (trimestral ou semestral).

9) Checklist executável (faça isso em ordem)

- 1) Criar pasta docs: C:\pharma\docs\16_hardening_producao\
- 2) Implementar requestId middleware + logger JSON (backend/src/common).
- 3) Implementar /health/live e /health/ready.
- 4) Configurar monitoramento: ping /health/ready + alerta de indisponibilidade.
- 5) Configurar alertas de erro e latência (p95) nas rotas críticas.
- 6) Configurar backup diário do DB + retenção.
- 7) Criar staging DB e testar restore semanal (procedimento).
- 8) Documentar parâmetros ENV e segredos (sem expor valores).

Critério para seguir adiante: checklist acima concluído e testado (health ok, backup ok, restore ok).

10) Definition of Done (DoD) – Hardening

- Logs JSON com requestId e erro com stack.
- /health/live e /health/ready funcionando e monitorado.
- Backup diário automático com retenção definida.
- Restore testado em staging com evidência.
- Alertas ativos (indisponibilidade + taxa de erro).

Pasta recomendada: **C:\pharma\docs\16_hardening_producao**