SPRINT 4 BACKEND - PLANO TECNICO

Consisa KB Governance: Ferramenta de Decisao Executiva

Data: 02/02/2026
Projeto: KB Governance (Java 21 + Spring Boot + PostgreSQL + Flyway)
Escopo: Backend apenas

A) DIAGNOSTICO DO ESTADO ATUAL

TAREFA 1.1 - Endpoints Existentes Relacionados

Controller	Endpoint	Proposito
GovernanceApiController	GET /api/v1/governance/issues	Lista issues paginada com filtros basicos
GovernanceApiController	PATCH /api/v1/governance/issues/{id}/status	Atualiza status
GovernanceApiController	POST /api/v1/governance/issues/{id}/assign	Atribui responsavel
GovernanceApiController	GET /api/v1/governance/issues/{id}/history	Historico de auditoria
DashboardApiController	GET /api/v1/dashboard/summary	Metricas agregadas basicas
KbAssignmentController	GET /kb/assignments/statistics	Stats de assignments de artigos
KbAssignmentController	GET /kb/assignments/overdue	Atribuicoes atrasadas

TAREFA 1.2 - Fraquezas para o Gestor

Problema	Impacto Executivo
Sem PRIORIDADE nas issues	Gestor nao sabe o que atacar primeiro
Sem SLA/Due Date nas issues	Nao ha como identificar issues atrasadas
Dashboard summary muito generico	Sem breakdown por prioridade, overdue, sem responsavel
Sem filtro "unassigned"	Nao consegue ver issues sem responsavel
Sem filtro "overdue"	Nao ha como listar issues que passaram do SLA
Resolucao sem comentario	Ao resolver, nao ha campo para comentario explicativo
Visao por sistema inexistente	Endpoint /systems/summary nao existe
Ordenacao padrao nao executiva	Listagem nao prioriza por urgencia+vencimento

TAREFA 1.3 - Fraquezas Tecnicas

Problema	Localizacao	Impacto
N+1 potencial em listagem	Repository	Performance degrada com volume
Sem indice em priority	kb_governance_issue	Impossivel ordenar/filtrar
Sem indice em sla_due_at	Campo nao existe	Impossivel calcular overdue
Contagens em memoria	countByStatus separados	Multiple roundtrips
DTO inconsistente	IssueRow	Frontend nao recebe priority/sla
Sem validacao de transicao	updateStatus	Permite RESOLVED->OPEN
Error handling generico	GlobalExceptionHandler	LGPD/seguranca

B) MODELO DE DOMINIO DO SPRINT 4

TAREFA 2.1 - PRIORIDADE (Enum GovernanceIssuePriority)

Valor	Peso	Criterio	SLA Default
CRITICAL	1	Sistema em producao afetado	24h
HIGH	2	Problema grave mas sem impacto imediato	72h (3 dias)
MEDIUM	3	Melhoria importante	168h (7 dias)
LOW	4	Cosmetico ou futuro	336h (14 dias)

> **Justificativa:** 4 niveis e suficiente para triagem executiva. Pesos numericos facilitam ordenacao SQL.

TAREFA 2.2 - SLA (Campo sla_due_at)

Decisao	Justificativa
Armazenar em kb_governance_issue	Permite queries diretas sem joins
Tipo: TIMESTAMPTZ	Consistencia com datas existentes
Calculado na criacao	created_at + SLA_DEFAULT[priority]
Editavel manualmente	Gestor pode ajustar prazo
NULL = sem SLA	Issues antigas sem prioridade

TAREFA 2.3 - "Sem Responsavel" (Identificacao)

Estrategia: Usar LEFT JOIN com kb_governance_issue_assignment

```
Issue e "sem responsavel" quando:
- NOT EXISTS assignment para issue_id com status IN ('OPEN', 'IN_PROGRESS')
- OU assignment existe mas status = 'CANCELLED' ou 'CLOSED'
```

> **Justificativa:** Reutiliza kb_governance_issue_assignment existente. Nao duplica dados.

TAREFA 2.4 - Overdue (Calculo)

```
Issue e "overdue" quando:
- sla_due_at IS NOT NULL
- sla_due_at < NOW()
- status NOT IN ('RESOLVED', 'IGNORED')
```

TAREFA 2.5 - Resolucao com Comentario

Campo	Tabela	Tipo
resolution_comment	kb_governance_issue	TEXT NULL
resolved_at	Ja existe	TIMESTAMPTZ
resolved_by	Ja existe	VARCHAR(100)

Regra: Ao mudar status para RESOLVED ou IGNORED, resolution_comment e obrigatorio.

TAREFA 2.6 - Auditoria

Tabela existente: kb_governance_issue_history

Campo	Uso
action	STATUS_CHANGE, PRIORITY_CHANGE, SLA_CHANGE, ASSIGNMENT, RESOLUTION
old_value	JSON com estado anterior
new_value	JSON com estado novo
actor	Quem fez a acao (obrigatorio)

TAREFA 2.7 - Assignment Separado vs Campo na Issue

Aspecto	Assignment Separado	Campo na Issue
Historico de atribuicoes	Multiplas	So ultima
Queries de listagem	Requer JOIN	Direto
Modelo existente	Ja existe	Mudanca de modelo
Flexibilidade	Pode ter dueDate proprio	Limitado

Decisao: Manter kb_governance_issue_assignment separado.

C) PLANO SPRINT 4 BACKEND (PASSOS)

Fase 1: Migrations e Modelo

1. Criar migration V11 - adicionar campos em kb_governance_issue
2. Criar indices para performance
3. Backfill de priority e sla_due_at em dados existentes
4. Atualizar entity KbGovernanceIssue
5. Criar enum GovernanceIssuePriority

Fase 2: Queries Agregadas

1. Criar queries nativas para overview
2. Criar queries para systems summary
3. Otimizar query de listagem com novos filtros

Fase 3: DTOs e Contratos

1. Criar DTOs para overview
2. Criar DTOs para systems summary
3. Atualizar DTO de issue com priority/sla
4. Criar DTOs para acoes (resolve, assign)

Fase 4: Endpoints

1. GET /api/v1/governance/overview
2. GET /api/v1/governance/systems/summary
3. Atualizar GET /api/v1/governance/issues com novos filtros
4. PATCH /api/v1/governance/issues/{id}/resolve
5. Atualizar POST /api/v1/governance/issues/{id}/assignments

6. Atualizar PATCH /api/v1/governance/issues/{id}/status

Fase 5: Validacoes e Auditoria

1. Validar transicoes de status

2. Garantir registro em history

3. Adicionar logs estruturados

Fase 6: Testes

1. Testes unitarios dos services

2. Testes de integracao dos endpoints

3. Testes de migrations

D) LISTA DE MUDANCAS POR ARQUIVOS

Novos Arquivos

Path	Descricao
src/main/resources/db/migration/ V11__governance_sprint4_priority_sla.sql	Migration
src/main/java/.../domain/GovernanceIssuePriority.java	Enum de prioridade
src/main/java/.../controller/api/dto/ GovernanceOverviewResponse.java	DTO overview
src/main/java/.../controller/api/dto/ SystemsSummaryResponse.java	DTO sistemas
src/main/java/.../controller/api/dto/ResolveIssueRequest.java	Request resolver
src/main/java/.../controller/api/dto/UpdatePriorityRequest.java	Request prioridade

Arquivos Alterados

Path	Alteracao
.../domain/KbGovernanceIssue.java	+priority, +slaDueAt, +resolutionComment
.../repository/ KbGovernanceIssueRepository.java	+queries agregadas, +filtros
.../service/GovernanceService.java	+getOverview(), +getSystemsSummary()
.../service/ GovernanceIssueWorkflowService.java	+resolveIssue(), validacoes
.../controller/api/ GovernanceApiController.java	+endpoints, +filtros
.../controller/api/dto/IssueResponse.java	+priority, +slaDueAt, +isOverdue
.../controller/api/dto/IssueListRequest.java	+overdueOnly, +unassignedOnly

Arquivos de Teste (Novos)

Path
src/test/java/.../service/GovernanceOverviewServiceTest.java
src/test/java/.../controller/GovernanceApiControllerIntegrationTest.java
src/test/java/.../repository/GovernanceIssueRepositoryTest.java

E) CONTRATOS JSON

E.1 GET /api/v1/governance/overview

Proposito: Metricas agregadas para visao executiva

Response 200:

```json
{
  "generatedAt": "2026-02-02T14:30:00Z",
  "totals": {
    "issues": 245,
    "open": 180,
    "assigned": 45,
    "inProgress": 12,
    "resolved": 8,
    "ignored": 0
  },
  "byPriority": {
    "critical": 5,
    "high": 23,
    "medium": 87,
    "low": 65,
    "unset": 65
  },
  "alerts": {
    "overdue": 12,
    "overduePercentage": 6.7,
    "unassigned": 135,
    "unassignedPercentage": 75.0,
    "dueSoon": 8
  },
  "trend": {
    "newLast7Days": 34,
    "resolvedLast7Days": 22,
    "netChange": 12
  }
}
```

E.2 GET /api/v1/governance/systems/summary

Proposito: Saude por sistema/produto

Response 200:

```json
{
  "generatedAt": "2026-02-02T14:30:00Z",
  "systems": [
    {
      "code": "BIOJOB",
      "name": "BioJob",
      "totals": {
        "issues": 45,
        "open": 30,
        "overdue": 3,
        "unassigned": 25
      },
      "byPriority": {
        "critical": 1,
        "high": 5,
        "medium": 20,
        "low": 19
      },
      "healthScore": 67.5,
      "articlesCount": 120
    }
  ],
  "summary": {
    "totalSystems": 5,
    "systemsWithCritical": 2,
    "worstSystem": "NOTAON"
  }
}
```

E.3 GET /api/v1/governance/issues

Query Params:

Param	Tipo	Default	Descricao
page	int	0	Pagina (0-indexed)
size	int	20	Itens por pagina (max 100)
systemCode	string	null	Filtrar por sistema
status	string	null	Filtrar por status
priority	string	null	Filtrar por prioridade
type	string	null	Filtrar por tipo
overdueOnly	boolean	false	Apenas issues atrasadas
unassignedOnly	boolean	false	Apenas sem responsavel
sort	string	"executive"	Ordenacao

Ordenacao padrao "executive":

```
ORDER BY
    CASE WHEN sla_due_at < NOW() THEN 0 ELSE 1 END,
    CASE priority WHEN 'CRITICAL' THEN 1 WHEN 'HIGH' THEN 2
         WHEN 'MEDIUM' THEN 3 WHEN 'LOW' THEN 4 ELSE 5 END,
    sla_due_at ASC NULLS LAST,
    created_at ASC
```

Response 200:

```json
{
  "content": [
    {
      "id": 1234,
      "articleId": 5678,
      "articleTitle": "Como configurar backup no BioJob",
      "systemCode": "BIOJOB",
      "systemName": "BioJob",
      "issueType": "OUTDATED_CONTENT",
      "severity": "WARN",
      "status": "OPEN",
      "priority": "HIGH",
      "message": "Artigo nao atualizado ha mais de 180 dias",
      "slaDueAt": "2026-02-03T18:00:00Z",
      "isOverdue": false,
      "daysUntilDue": 1,
      "assignment": {
        "agentId": "agent-123",
        "agentName": "Joao Silva",
        "assignedAt": "2026-01-28T10:00:00Z",
        "dueDate": "2026-02-03T18:00:00Z"
      },
      "createdAt": "2026-01-25T14:30:00Z",
      "updatedAt": "2026-01-28T10:00:00Z"
    }
  ],
  "page": 0,
  "size": 20,
  "totalElements": 180,
  "totalPages": 9
}
```

E.4 PATCH /api/v1/governance/issues/{id}/status

Request Body:

```json
{
  "status": "IN_PROGRESS",
  "actor": "maria.souza@consisa.com.br"
}
```

Transicoes Validas:

- OPEN → ASSIGNED, IN_PROGRESS, IGNORED

- ASSIGNED → IN_PROGRESS, OPEN, IGNORED

- IN_PROGRESS → RESOLVED (via /resolve), IGNORED, ASSIGNED

- RESOLVED → OPEN (reabrir)

- IGNORED → OPEN (reabrir)

Response 200:

```json
{
  "id": 1234,
  "previousStatus": "OPEN",
  "newStatus": "IN_PROGRESS",
  "updatedAt": "2026-02-02T14:30:00Z",
  "actor": "maria.souza@consisa.com.br"
}
```

Erro 400:

```json
{
  "error": "INVALID_TRANSITION",
  "message": "Cannot transition from RESOLVED to IN_PROGRESS",
  "currentStatus": "RESOLVED",
  "requestedStatus": "IN_PROGRESS"
}
```

E.5 PATCH /api/v1/governance/issues/{id}/resolve

Request Body:

```json
{
  "resolution": "RESOLVED",
  "comment": "Artigo atualizado com nova documentacao do modulo fiscal.",
  "actor": "joao.silva@consisa.com.br"
}
```

Campo	Obrigatorio	Validacao
resolution	Sim	RESOLVED ou IGNORED
comment	Sim	Min 10 chars, max 2000
actor	Sim	Email valido

Response 200:

```
{
  "id": 1234,
  "status": "RESOLVED",
  "resolutionComment": "Artigo atualizado com nova documentacao...",
  "resolvedAt": "2026-02-02T14:30:00Z",
  "resolvedBy": "joao.silva@consisa.com.br"
}
```

E.6 POST /api/v1/governance/issues/{id}/ assignments

Request Body:

```
{
  "agentId": "agent-456",
  "dueDate": "2026-02-05T18:00:00Z",
  "actor": "gestor@consisa.com.br",
  "note": "Prioridade por solicitacao do cliente ABC"
}
```

Response 201:

```
{
  "id": 789,
  "issueId": 1234,
  "agent": {
    "id": "agent-456",
    "name": "Carlos Mendes",
    "email": "carlos.mendes@consisa.com.br"
  },
  "assignedAt": "2026-02-02T14:30:00Z",
  "dueDate": "2026-02-05T18:00:00Z",
  "assignedBy": "gestor@consisa.com.br",
  "previousAssignment": {
    "agentId": "agent-123",
    "agentName": "Joao Silva",
    "cancelledAt": "2026-02-02T14:30:00Z"
  }
}
```

E.7 PATCH /api/v1/governance/issues/{id}/priority

Request Body:

```
{
  "priority": "CRITICAL",
  "recalculateSla": true,
  "actor": "gestor@consisa.com.br"
}
```

Response 200:

```
{
  "id": 1234,
  "previousPriority": "MEDIUM",
  "newPriority": "CRITICAL",
  "previousSlaDueAt": "2026-02-09T18:00:00Z",
  "newSlaDueAt": "2026-02-03T14:30:00Z",
  "updatedAt": "2026-02-02T14:30:00Z"
}
```

F) MIGRATIONS SQL PROPOSTAS

V11__governance_sprint4_priority_sla.sql

```sql
-- ============================================
-- SPRINT 4: Priority, SLA, Resolution Comment
-- ============================================

-- 1. Adicionar campos em kb_governance_issue
ALTER TABLE kb_governance_issue
ADD COLUMN IF NOT EXISTS priority VARCHAR(20) DEFAULT NULL,
ADD COLUMN IF NOT EXISTS sla_due_at TIMESTAMPTZ DEFAULT NULL,
ADD COLUMN IF NOT EXISTS resolution_comment TEXT DEFAULT NULL;

-- 2. Constraint para valores de priority
ALTER TABLE kb_governance_issue
ADD CONSTRAINT chk_governance_issue_priority
CHECK (priority IS NULL OR priority IN ('CRITICAL', 'HIGH', 'MEDIUM', 'LOW'));

-- 3. Indices para filtros e ordenacao executiva
CREATE INDEX IF NOT EXISTS idx_governance_issue_priority
ON kb_governance_issue(priority)
WHERE status NOT IN ('RESOLVED', 'IGNORED');

CREATE INDEX IF NOT EXISTS idx_governance_issue_sla_due_at
ON kb_governance_issue(sla_due_at)
WHERE status NOT IN ('RESOLVED', 'IGNORED');

CREATE INDEX IF NOT EXISTS idx_governance_issue_status_priority
ON kb_governance_issue(status, priority, sla_due_at);

-- 4. Indice para agregacao por sistema (via article)
CREATE INDEX IF NOT EXISTS idx_governance_issue_article_status
ON kb_governance_issue(article_id, status);

-- 5. Indice para overview agregada
CREATE INDEX IF NOT EXISTS idx_governance_issue_overview
ON kb_governance_issue(status, priority, sla_due_at, created_at);

-- 6. Backfill: Definir priority baseado em severity existente
UPDATE kb_governance_issue
SET priority = CASE
    WHEN severity = 'ERROR' THEN 'HIGH'
    WHEN severity = 'WARN' THEN 'MEDIUM'
    WHEN severity = 'INFO' THEN 'LOW'
    ELSE 'MEDIUM'
END
WHERE priority IS NULL
AND status NOT IN ('RESOLVED', 'IGNORED');

-- 7. Backfill: Calcular SLA baseado em priority
UPDATE kb_governance_issue
SET sla_due_at = CASE priority
```

```
    WHEN 'CRITICAL' THEN created_at + INTERVAL '24 hours'
    WHEN 'HIGH' THEN created_at + INTERVAL '72 hours'
    WHEN 'MEDIUM' THEN created_at + INTERVAL '168 hours'
    WHEN 'LOW' THEN created_at + INTERVAL '336 hours'
    ELSE created_at + INTERVAL '168 hours'
END
WHERE sla_due_at IS NULL
AND priority IS NOT NULL
AND status NOT IN ('RESOLVED', 'IGNORED');

-- 8. Campo para rastrear se SLA foi definido manualmente
ALTER TABLE kb_governance_issue
ADD COLUMN IF NOT EXISTS sla_manual BOOLEAN DEFAULT FALSE;

-- 9. Indice composto para query executiva principal
CREATE INDEX IF NOT EXISTS idx_governance_issue_executive_queue
ON kb_governance_issue(
    (CASE WHEN sla_due_at < NOW() THEN 0 ELSE 1 END),
    priority,
    sla_due_at,
    created_at
)
WHERE status NOT IN ('RESOLVED', 'IGNORED');

-- 10. Indice para buscar assignments ativos por issue
CREATE INDEX IF NOT EXISTS idx_governance_issue_assignment_active
ON kb_governance_issue_assignment(issue_id, status)
WHERE status IN ('OPEN', 'IN_PROGRESS');
```

Impactos em Dados Existentes

Aspecto	Impacto	Mitigacao
Issues sem priority	Backfill baseado em severity	Mapeamento conservador
Issues sem SLA	Calculado a partir de created_at	Muitas estarao overdue
Issues RESOLVED/ IGNORED	Nao alteradas	Mantem estado historico
Novo indice grande	Impacto em INSERT/UPDATE	Indices parciais (WHERE)

Rollback Script

```sql
-- Rollback V11 (usar apenas se necessario)
ALTER TABLE kb_governance_issue
DROP COLUMN IF EXISTS priority,
DROP COLUMN IF EXISTS sla_due_at,
DROP COLUMN IF EXISTS resolution_comment,
DROP COLUMN IF EXISTS sla_manual;

DROP INDEX IF EXISTS idx_governance_issue_priority;
DROP INDEX IF EXISTS idx_governance_issue_sla_due_at;
DROP INDEX IF EXISTS idx_governance_issue_status_priority;
DROP INDEX IF EXISTS idx_governance_issue_executive_queue;
```

G) QUERIES E PERFORMANCE

G.1 Overview Agregada

Recomendacao: SQL Nativo com uma unica query usando agregacoes condicionais

```sql
SELECT
    COUNT(*) AS total_issues,
    COUNT(*) FILTER (WHERE status = 'OPEN') AS open_count,
    COUNT(*) FILTER (WHERE status = 'ASSIGNED') AS assigned_count,
    COUNT(*) FILTER (WHERE status = 'IN_PROGRESS') AS in_progress_count,
    COUNT(*) FILTER (WHERE status = 'RESOLVED') AS resolved_count,
    COUNT(*) FILTER (WHERE status = 'IGNORED') AS ignored_count,

    COUNT(*) FILTER (WHERE priority = 'CRITICAL'
                    AND status NOT IN ('RESOLVED', 'IGNORED')) AS critical_count,
    COUNT(*) FILTER (WHERE priority = 'HIGH'
                    AND status NOT IN ('RESOLVED', 'IGNORED')) AS high_count,
    COUNT(*) FILTER (WHERE priority = 'MEDIUM'
                    AND status NOT IN ('RESOLVED', 'IGNORED')) AS medium_count,
    COUNT(*) FILTER (WHERE priority = 'LOW'
                    AND status NOT IN ('RESOLVED', 'IGNORED')) AS low_count,

    COUNT(*) FILTER (WHERE sla_due_at < NOW()
                    AND status NOT IN ('RESOLVED', 'IGNORED')) AS overdue_count,
    COUNT(*) FILTER (WHERE sla_due_at BETWEEN NOW() AND NOW() + INTERVAL '48 hours'
                    AND status NOT IN ('RESOLVED', 'IGNORED')) AS due_soon_count,

    COUNT(*) FILTER (WHERE created_at >= NOW() - INTERVAL '7 days') AS new_last_7_days,
    COUNT(*) FILTER (WHERE resolved_at >= NOW() - INTERVAL '7 days') AS resolved_last_7_days

FROM kb_governance_issue;
```

> **Por que SQL Nativo:** FILTER clause e especifica do PostgreSQL, performance muito superior a multiplos COUNT separados.

G.2 Systems Summary

```sql
SELECT
    s.code AS system_code,
    s.name AS system_name,
    COUNT(*) AS total_issues,
    COUNT(*) FILTER (WHERE gi.status = 'OPEN') AS open_count,
    COUNT(*) FILTER (WHERE gi.sla_due_at < NOW()
                     AND gi.status NOT IN ('RESOLVED', 'IGNORED')) AS overdue_count,
    COUNT(*) FILTER (WHERE gi.priority = 'CRITICAL'
                     AND gi.status NOT IN ('RESOLVED', 'IGNORED')) AS critical_count,
    (SELECT COUNT(*) FROM kb_article a2 WHERE a2.system_id = s.id) AS articles_count
FROM kb_governance_issue gi
JOIN kb_article a ON a.id = gi.article_id
JOIN kb_system s ON s.id = a.system_id
GROUP BY s.id, s.code, s.name
ORDER BY overdue_count DESC, critical_count DESC;
```

G.3 JPQL vs SQL Nativo - Decisao

Query	Escolha	Justificativa
Overview	SQL Nativo	FILTER clause
Systems Summary	SQL Nativo	GROUP BY complexo
Issues List	SQL Nativo	LATERAL join + ordenacao
History by Issue	JPQL	Query simples
Single Issue	JPQL	findById padrao
Create/Update	JPA	Transacional padrao

H) CRITERIOS DE ACEITACAO E TESTES

Checklist Build/Deploy

[ ] mvn clean verify passa sem erros

[ ] Flyway V11 executa em banco novo (do zero)

[ ] Flyway V11 executa em banco com dados existentes

[ ] Nenhum warning de deprecation nas entities

[ ] Application startup sem erros

Checklist Migrations

[ ] V11 e idempotente (IF NOT EXISTS)

[ ] Backfill nao afeta RESOLVED/IGNORED

[ ] Indices criados corretamente

[ ] Constraint de priority funciona

[ ] Rollback script testado

Checklist Endpoints

Endpoint	Criterio
GET /overview	Retorna todas as metricas em < 500ms
GET /overview	overdue count bate com filtro overdueOnly=true
GET /overview	unassigned count bate com filtro unassignedOnly=true
GET /systems/summary	Lista todos sistemas com issues
GET /systems/summary	healthScore calculado corretamente
GET /issues	Paginacao funciona (page, size)
GET /issues	Filtro systemCode funciona
GET /issues	Filtro priority funciona
GET /issues	Filtro overdueOnly retorna apenas overdue
GET /issues	Filtro unassignedOnly retorna sem assignment
GET /issues	Ordenacao executiva e default
GET /issues	isOverdue calculado corretamente
PATCH /status	Transicao valida funciona
PATCH /status	Transicao invalida retorna 400
PATCH /status	Registra em history
PATCH /resolve	Comentario obrigatorio
PATCH /resolve	Comentario < 10 chars retorna 400
PATCH /resolve	Define resolved_at e resolved_by
PATCH /resolve	Registra em history
POST /assignments	Cria assignment
POST /assignments	Atualiza status para ASSIGNED

POST /assignments	Cancela assignment anterior se existir
POST /assignments	Registra em history
POST /assignments	Agente inativo retorna 400

Checklist Auditoria

[ ] Toda mudanca de status registra history

[ ] Toda mudanca de priority registra history

[ ] Toda atribuicao registra history

[ ] Toda resolucao registra history

[ ] Actor e obrigatorio em todas acoes

Checklist Logs e LGPD

[ ] Logs de actions nao contem dados pessoais

[ ] Logs contem IDs para rastreabilidade

[ ] Logs de erro nao expoem stack traces ao cliente

[ ] Mensagens de erro sao genericas para o cliente

[ ] Logs internos tem nivel adequado (INFO para actions)

Testes Unitarios Minimos

```
GovernanceOverviewServiceTest
    - shouldCalculateOverviewMetrics()
    - shouldCountOverdueCorrectly()
    - shouldCountUnassignedCorrectly()

GovernanceIssueWorkflowServiceTest
    - shouldResolveWithComment()
    - shouldRejectResolutionWithoutComment()
    - shouldValidateStatusTransition()
    - shouldRecordHistory()

GovernanceAssignmentServiceTest
    - shouldCreateAssignment()
    - shouldCancelPreviousAssignment()
    - shouldRejectInactiveAgent()
```

Testes de Integracao Minimos

```
GovernanceApiControllerIntegrationTest
    - testOverviewEndpoint()
    - testSystemsSummaryEndpoint()
    - testIssuesListWithFilters()
    - testResolveIssue()
    - testAssignIssue()

MigrationIntegrationTest
    - testV11OnEmptyDatabase()
    - testV11OnExistingData()
    - testBackfillPriority()
    - testBackfillSla()
```

RESUMO EXECUTIVO

O que muda

1. **Modelo:** kb_governance_issue ganha priority, sla_due_at, resolution_comment
2. **Queries:** Novas queries agregadas com SQL nativo para performance
3. **Endpoints:** 3 novos + 3 atualizados
4. **Validacoes:** Transicao de status, comentario obrigatorio em resolucao
5. **Auditoria:** Novos actions no history

O que reutiliza

1. kb_governance_issue_assignment - continua sendo tabela de atribuicoes
2. kb_governance_issue_history - continua sendo tabela de auditoria
3. KbAgent - responsaveis continuam vindo dessa tabela
4. Enums existentes (GovernanceIssueStatus, GovernanceIssueType, GovernanceSeverity)

Riscos e Mitigacoes

Risco	Probabilidade	Mitigacao
Backfill marca muitos como overdue	Alta	Comunicar stakeholders
Performance em overview	Media	Indices parciais, cache se necessario
Migration falha em prod	Baixa	Testar em staging com dump