

# Documentação Técnica do Sistema SIGA

## Sistema Integrado de Gestão de Almoxarifado - CBM MT

**Versão:** 6.0

**Data:** 17 de junho de 2025

**Autor:** Documentação gerada com base na modelagem PostgreSQL v6.0

---

## Sumário

1. [Introdução](#)
  2. [Visão Geral do Sistema](#)
  3. [Diagramas de Estrutura e Relacionamentos](#)
  4. 3.1. [Diagrama Entidade-Relacionamento \(ERD\)](#)
  5. 3.2. [Diagrama de Classes UML](#)
  6. 3.3. [Diagrama de Componentes/Módulos](#)
  7. [Diagramas de Processo e Fluxo](#)
  8. 4.1. [Diagrama de Fluxo de Dados \(DFD\)](#)
  9. 4.2. [Diagrama de Swimlanes de Processo](#)
  10. 4.3. [Diagrama de Sequência UML](#)
  11. [Diagramas Técnicos Especializados](#)
  12. 5.1. [Diagrama de Particionamento](#)
  13. 5.2. [Diagrama de Índices e Otimização](#)
  14. 5.3. [Diagrama de Implantação \(Deployment\)](#)
  15. [Diagramas de Casos de Uso](#)
  16. 6.1. [Diagrama de Caso de Uso \(Use Case\)](#)
  17. [Considerações Finais](#)
- 

## 1. Introdução

O Sistema Integrado de Gestão de Almoxarifado (SIGA) é uma solução completa desenvolvida para o Corpo de Bombeiros Militar de Mato Grosso (CBM-MT), destinada ao controle e gerenciamento eficiente de materiais, equipamentos e viaturas.

Este documento apresenta uma visão abrangente da arquitetura do sistema através de diagramas Mermaid organizados por domínios funcionais, facilitando a compreensão, manutenção e evolução do sistema.

## Objetivos da Documentação

- Fornecer visão clara da estrutura de dados e relacionamentos
  - Documentar processos de negócio e fluxos operacionais
  - Apresentar aspectos técnicos de implementação
  - Facilitar a manutenção e evolução do sistema
  - Servir como referência para desenvolvedores e administradores
- 

## 2. Visão Geral do Sistema

O SIGA é estruturado em módulos funcionais que atendem às necessidades específicas de gestão de almoxarifado militar:

### Módulos Principais

**Configuração e Segurança** - Gestão de unidades do CBM com hierarquia organizacional - Cadastro e controle de militares por posto/graduação - Sistema de permissões e responsabilidades

**Gestão de Almoxarifado** - Controle de almoxarifados por unidade - Definição de localizações específicas para armazenamento - Monitoramento de estoque atual em tempo real

**Catálogo de Materiais** - Categorização dinâmica e hierárquica de materiais - Gestão especializada por tipo (consumo, permanente, equipamento, viatura) - Controle de especificações técnicas e documentação

**Movimentações e Operações** - Registro de todas as operações de movimentação - Controle de entradas, saídas, transferências e cautelas - Rastreabilidade completa com histórico detalhado

**Histórico e Auditoria** - Registro histórico de todas as movimentações - Sistema de auditoria com logs detalhados - Relatórios e consultas históricas

**Notificações** - Sistema automatizado de notificações - Alertas de estoque crítico e vencimentos - Comunicação integrada via LISTEN/NOTIFY

---

## 3. Diagramas de Estrutura e Relacionamentos

### 3.1. Diagrama Entidade-Relacionamento (ERD)

O diagrama a seguir apresenta a estrutura completa do banco de dados SIGA, mostrando todas as entidades principais e seus relacionamentos. Para melhor legibilidade, o diagrama está organizado por módulos funcionais.

#### ERD Completo - Módulo Principal

```
erDiagram
    %% Configuração e Segurança
    unidade_cbm {
        int id_unidade PK
        varchar nome_unidade
        varchar sigla_unidade UK
        text endereco
        varchar telefone
        varchar email
        int comandante_id FK
        int unidade_superior_id FK
        boolean ativa
        jsonb informacoes_complementares
        timestamp data_criacao
        timestamp data_atualizacao
    }

    militar {
        int id_militar PK
        varchar matricula UK
        varchar nome_completo
        varchar nome_guerra
        varchar cpf UK
        varchar posto_graduacao
        int unidade_id FK
        enum situacao
        date data_nascimento
        varchar telefone
        varchar email
        text endereco
        date data_incorporacao
        jsonb informacoes_complementares
        boolean ativo
        timestamp data_criacao
        timestamp data_atualizacao
    }

    %% Gestão de Almoxarifado
    almoxarifado {
```

```

    int id_almoxarifado PK
    varchar nome_almoxarifado
    text descricao
    int unidade_id FK
    int responsavel_id FK
    text endereco
    numeric capacidade_maxima
    numeric area_total
    jsonb configuracoes_especificas
    boolean ativo
    timestamp data_criacao
    timestamp data_atualizacao
}

localizacao {
    int id_localizacao PK
    int almoxarifado_id FK
    varchar codigo_localizacao UK
    varchar descricao_localizacao
    varchar setor
    varchar prateleira
    varchar nivel
    varchar posicao
    numeric capacidade_maxima
    jsonb restricoes_especiais
    boolean ativa
    timestamp data_criacao
}

estoque_atual {
    int id_estoque PK
    int material_id FK
    int almoxarifado_id FK
    int localizacao_id FK
    int quantidade_disponivel
    int quantidade_reservada
    int quantidade_em_manutencao
    numeric valor_total
    timestamp data_ultima_movimentacao
    timestamp data_atualizacao
}

%% Catálogo de Materiais
categoria_material {
    int id_categoria PK
    varchar nome_categoria UK
    text descricao_categoria
    int categoria_pai_id FK
    int nivel_hierarquia
    varchar codigo_categoria UK
    jsonb configuracoes_especificas
    boolean ativa

```

```

        timestamp data_criacao
        timestamp data_atualizacao
    }

    tipo_viatura {
        int id_tipo_viatura PK
        varchar nome_tipo UK
        text descricao_tipo
        varchar categoria_principal
        varchar subcategoria
        jsonb especificacoes_tecnicas
        jsonb requisitos_operacionais
        boolean ativo
        timestamp data_criacao
        timestamp data_atualizacao
    }

```

```

material_base {
    int id_material PK
    varchar codigo_material UK
    varchar nome_material
    text descricao_material
    int categoria_id FK
    varchar unidade_medida
    numeric valor_unitario
    int estoque_minimo
    int estoque_maximo
    enum situacao
    int localizacao_padrao_id FK
    text observacoes
    jsonb atributos_adicionais
    boolean ativo
    timestamp data_criacao
    timestamp data_atualizacao
}

```

%% Relacionamentos Principais

```

unidade_cbm ||--o{ militar : "lotacao"
unidade_cbm ||--o{ unidade_cbm : "hierarquia"
unidade_cbm ||--o{ almoxarifado : "possui"
militar ||--o{ almoxarifado : "responsavel"
almoxarifado ||--o{ localizacao : "contem"
categoria_material ||--o{ categoria_material : "hierarquia"
categoria_material ||--o{ material_base : "categoriza"
material_base ||--o{ estoque_atual : "estocado"
almoxarifado ||--o{ estoque_atual : "armazena"
localizacao ||--o{ estoque_atual : "localiza"
localizacao ||--o{ material_base : "padrao"

```

## ERD - Especialização de Materiais

```
erDiagram
    material_base {
        int id_material PK
        varchar codigo_material UK
        varchar nome_material
        text descricao_material
        int categoria_id FK
        varchar unidade_medida
        numeric valor_unitario
        int estoque_minimo
        int estoque_maximo
        enum situacao
        int localizacao_padrao_id FK
        text observacoes
        jsonb atributos_adicionais
        boolean ativo
        timestamp data_criacao
        timestamp data_atualizacao
    }

    material_consumo {
        int id_material_consumo PK
        int material_base_id FK
        date data_validade
        varchar lote
        varchar fabricante
        date data_fabricacao
        text instrucoes_uso
        text restricoes_armazenamento
    }

    material_permanente {
        int id_material_permanente PK
        int material_base_id FK
        varchar numero_patrimonio UK
        date data_aquisicao
        numeric valor_aquisicao
        int vida_util_anos
        varchar estado_conservacao
        boolean necessita_manutencao
        date proxima_manutencao
        jsonb historico_manutencao
    }

    equipamento {
        int id_equipamento PK
        int material_base_id FK
        varchar numeroserie
        varchar modelo
    }
```

```

        varchar fabricante
        int ano_fabricacao
        jsonb especificacoes_tecnicas
        text manual_operacao
        jsonb certificacoes
    }

    viatura {
        int id_viatura PK
        int material_base_id FK
        int tipo_viatura_id FK
        varchar placa UK
        varchar chassi UK
        varchar renavam
        int ano_fabricacao
        int ano_modelo
        varchar cor
        varchar combustivel
        int quilometragem
        numeric capacidade_tanque
        boolean documentacao_regular
        date proxima_revisao
        boolean seguro_vigente
        date vencimento_seguro
        text observacoes_viatura
    }

    %% Relacionamentos de Especialização
    material_base ||--o| material_consumo : "especializa"
    material_base ||--o| material_permanente : "especializa"
    material_base ||--o| equipamento : "especializa"
    material_base ||--o| viatura : "especializa"
    tipo_viatura ||--o{ viatura : "classifica"

```

## ERD - Operações e Movimentações

```

erDiagram
    operacao {
        bigint id_operacao PK
        varchar numero_operacao
        enum tipo_movimentacao
        int material_id FK
        int almoxarifado_origem_id FK
        int almoxarifado_destino_id FK
        int militar_responsavel_id FK
        int militar_recebedor_id FK
        int quantidade
        numeric valor_unitario
        numeric valor_total
        timestamp data_operacao
    }

```

```

    date data_prevista_devolucao
    enum status
    enum documento_tipo
    varchar numero_documento
    text observacoes
    jsonb detalhes_adicionais
    timestamp data_criacao
    timestamp data_atualizacao
}

entrada_detalhe {
    int id_entrada_detalhe PK
    bigint operacao_id FK
    varchar fornecedor
    varchar nota_fiscal
    date data_nota_fiscal
    numeric valor_nota_fiscal
    text condicoes_armazenamento
    date data_validade
    varchar lote
    varchar certificado_qualidade
    jsonb detalhes_adicionais_entrada
}

saida_detalhe {
    int id_saida_detalhe PK
    bigint operacao_id FK
    varchar finalidade_uso
    varchar projeto_operacao
    text justificativa
    date data_prevista_retorno
    varchar responsavel_externo
    varchar contato_externo
    jsonb detalhes_adicionais_saida
}

transferencia_detalhe {
    int id_transferencia_detalhe PK
    bigint operacao_id FK
    varchar motivo_transferencia
    varchar autorizacao_superior
    date data_autorizacao
    varchar responsavel_transporte
    text condicoes_transporte
    jsonb detalhes_adicionais_transferencia
}

cautela_detalhe {
    int id_cautela_detalhe PK
    bigint operacao_id FK
    varchar numero_cautela
    date data_cautela

```



```

    date data_prevista_devolucao
    text condicoes_uso
    text responsabilidades
    varchar testemunha_1
    varchar testemunha_2
    jsonb detalhes_adicionais_cautela
}

```

```

devolucao_detalhe {
    int id_devolucao_detalhe PK
    bigint operacao_id FK
    bigint operacao_origem_id FK
    varchar condicoes_devolucao
    text avaliacao_estado
    boolean necessita_manutencao
    text observacoes_devolucao
    jsonb detalhes_adicionais_devolucao
}

```

```

historico_estoque {
    bigint id_historico PK
    int material_id FK
    int almoxarifado_id FK
    int localizacao_id FK
    bigint operacao_id FK
    int quantidade_anterior
    int quantidade_movimentada
    int quantidade_posterior
    numeric valor_unitario
    numeric valor_total
    timestamp data_movimentacao
    enum tipo_movimentacao
    varchar observacoes
    jsonb detalhes_historico
}

```

%% Relacionamentos das Operações

```

material_base ||--o{ operacao : "movimenta"
almoxarifado ||--o{ operacao : "origem"
almoxarifado ||--o{ operacao : "destino"
militar ||--o{ operacao : "responsavel"
militar ||--o{ operacao : "recebedor"

```

```

operacao ||--o| entrada_detalhe : "detalha"
operacao ||--o| saida_detalhe : "detalha"
operacao ||--o| transferencia_detalhe : "detalha"
operacao ||--o| cautela_detalhe : "detalha"
operacao ||--o| devolucao_detalhe : "detalha"

```

```

operacao ||--o{ historico_estoque : "gera"
material_base ||--o{ historico_estoque : "historico"

```

```
almoxarifado ||--o{ historico_estoque : "local"
localizacao ||--o{ historico_estoque : "posicao"
```

## ERD - Sistema de Notificações e Auditoria

```
erDiagram
    notificacao {
        bigint id_notificacao PK
        varchar titulo
        text mensagem
        enum prioridade
        varchar tipo_notificacao
        int destinatario_id FK
        int remetente_id FK
        boolean lida
        timestamp data_envio
        timestamp data_leitura
        jsonb dados_contexto
        varchar canal_notificacao
        boolean ativa
    }

    log_auditoria {
        bigint id_log PK
        varchar tabela_afetada
        varchar operacao
        int usuario_id FK
        jsonb dados_anteriores
        jsonb dados_posteriores
        varchar ip_origem
        varchar user_agent
        timestamp data_operacao
        text observacoes
        varchar sessao_id
    }

    %% Relacionamentos de Notificação e Auditoria
    militar ||--o{ notificacao : "destinatario"
    militar ||--o{ notificacao : "remetente"
    militar ||--o{ log_auditoria : "usuario"
```

## Descrição dos Relacionamentos Principais

**Hierarquia Organizacional** - Unidades CBM possuem estrutura hierárquica com unidade superior - Militares são lotados em unidades específicas - Comandantes são militares responsáveis por unidades

**Gestão de Almoxarifado** - Cada unidade pode ter múltiplos almoxarifados - Almoxarifados possuem localizações específicas para organização - Militares são responsáveis por almoxarifados

**Categorização de Materiais** - Categorias possuem estrutura hierárquica (categoria pai/filha) - Materiais são classificados por categoria - Especialização por tipo (consumo, permanente, equipamento, viatura)

**Controle de Estoque** - Estoque atual vincula material, almoxarifado e localização - Operações registram todas as movimentações - Histórico mantém rastreabilidade completa

**Sistema de Detalhamento** - Cada tipo de operação possui tabela de detalhes específica - Permite flexibilidade para informações particulares de cada tipo - Mantém integridade referencial com operação principal

### 3.2. Diagrama de Classes UML

O diagrama de classes apresenta a estrutura do sistema SIGA em uma perspectiva orientada a objetos, mostrando as classes principais, seus atributos, métodos e relacionamentos de herança e associação.

#### Classes Principais - Módulo de Configuração

```
classDiagram
    class UnidadeCBM {
        -int idUnidade
        -string nomeUnidade
        -string siglaUnidade
        -string endereco
        -string telefone
        -string email
        -int comandanteId
        -int unidadeSuperiorId
        -boolean ativa
        -JSON informacoesComplementares
        -DateTime dataCriacao
        -DateTime dataAtualizacao
        +criarUnidade()
        +atualizarInformacoes()
        +definirComandante()
        +ativarDesativar()
        +obterSubunidades()
    }

    class Militar {
        -int idMilitar
```

```

-String matricula
-String nomeCompleto
-String nomeGuerra
-String cpf
-String postoGraduacao
-int unidadeId
-SituacaoMilitar situacao
-Date dataNascimento
-String telefone
-String email
-String endereco
-Date dataIncorporacao
-JSON informacoesComplementares
-boolean ativo
+cadastrarMilitar()
+atualizarDados()
+alterarSituacao()
+transferirUnidade()
+obterHistorico()
}

class SituacaoMilitar {
  <<enumeration>>
  ATIVO
  LICENCIADO
  AFASTADO
  RESERVISTA
  REFORMADO
}

UnidadeCBM ||--o{ Militar : "lotacao"
UnidadeCBM ||--o{ UnidadeCBM : "hierarquia"
Militar ||-- SituacaoMilitar : "possui"

```

## Classes de Almoxarifado e Localização

```

classDiagram
  class Almoxarifado {
    -int idAlmoxarifado
    -String nomeAlmoxarifado
    -String descricao
    -int unidadeId
    -int responsavelId
    -String endereco
    -decimal capacidadeMaxima
    -decimal areaTotal
    -JSON configuracoesEspecificas
    -boolean ativo
    +criarAlmoxarifado()
    +definirResponsavel()
  }

```

```

+configurarEspecificacoes()
+calcularOcupacao()
+gerarRelatorioEstoque()
}

class Localizacao {
    -int idLocalizacao
    -int almoxarifadoId
    -string codigoLocalizacao
    -string descricaoLocalizacao
    -string setor
    -string prateleira
    -string nivel
    -string posicao
    -decimal capacidadeMaxima
    -JSON restricoesEspeciais
    -boolean ativa
    +definirLocalizacao()
    +verificarCapacidade()
    +aplicarRestricoes()
    +obterMateriais()
}

class EstoqueAtual {
    -int idEstoque
    -int materialId
    -int almoxarifadoId
    -int localizacaoId
    -int quantidadeDisponivel
    -int quantidadeReservada
    -int quantidadeEmManutencao
    -decimal valorTotal
    -DateTime dataUltimaMovimentacao
    +atualizarQuantidade()
    +reservarMaterial()
    +liberarReserva()
    +calcularValorTotal()
    +verificarEstoqueMinimo()
}

Almoxarifado ||--o{ Localizacao : "contem"
Almoxarifado ||--o{ EstoqueAtual : "armazena"
Localizacao ||--o{ EstoqueAtual : "localiza"

```

## Hierarquia de Classes de Material

```

classDiagram
    class MaterialBase {
        <<abstract>>
        -int idMaterial
    }

```

```

        -string codigoMaterial
        -string nomeMaterial
        -string descricaoMaterial
        -int categoriaId
        -string unidadeMedida
        -decimal valorUnitario
        -int estoqueMinimo
        -int estoqueMaximo
        -SituacaoMaterial situacao
        -int localizacaoPadraoId
        -string observacoes
        -JSON atributosAdicionais
        -boolean ativo
        +cadastrarMaterial()*
        +atualizarInformacoes()*
        +alterarSituacao()
        +definirEstoqueLimites()
        +calcularValorEstoque()
    }

    class MaterialConsumo {
        -int idMaterialConsumo
        -Date dataValidade
        -string lote
        -string fabricante
        -Date dataFabricacao
        -string instrucoesUso
        -string restricoesArmazenamento
        +verificarValidade()
        +controlarLote()
        +definirInstrucoesUso()
        +aplicarRestricoes()
    }

    class MaterialPermanente {
        -int idMaterialPermanente
        -string numeroPatrimonio
        -Date dataAquisicao
        -decimal valorAquisicao
        -int vidaUtilAnos
        -string estadoConservacao
        -boolean necessitaManutencao
        -Date proximaManutencao
        -JSON historicoManutencao
        +patrimoniar()
        +avaliarEstado()
        +programarManutencao()
        +registrarManutencao()
        +calcularDepreciacao()
    }

    class Equipamento {

```

```

    -int idEquipamento
    -string numeroSerie
    -string modelo
    -string fabricante
    -int anoFabricacao
    -JSON especificacoesTecnicas
    -string manualOperacao
    -JSON certificacoes
    +registrarEspecificacoes()
    +anexarManual()
    +validarCertificacoes()
    +programarCalibracao()
}

```

```

class Viatura {
    -int idViatura
    -int tipoViaturaId
    -string placa
    -string chassi
    -string renavam
    -int anoFabricacao
    -int anoModelo
    -string cor
    -string combustivel
    -int quilometragem
    -decimal capacidadeTanque
    -boolean documentacaoRegular
    -Date proximaRevisao
    -boolean seguroVigente
    -Date vencimentoSeguro
    +registrarDocumentacao()
    +atualizarQuilometragem()
    +programarRevisao()
    +renovarSeguro()
    +gerarRelatorioManutencao()
}

```

```

class SituacaoMaterial {
    <<enumeration>>
    DISPONIVEL
    EM_USO
    MANUTENCAO
    BAIXADO
    RESERVADO
}

```

```

MaterialBase <|-- MaterialConsumo : "herda"
MaterialBase <|-- MaterialPermanente : "herda"
MaterialBase <|-- Equipamento : "herda"
MaterialBase <|-- Viatura : "herda"
MaterialBase ||-- SituacaoMaterial : "possui"

```

## Classes de Operação e Movimentação

```
classDiagram
    class Operacao {
        -bigint idOperacao
        -string numeroOperacao
        -TipoMovimentacao tipoMovimentacao
        -int materialId
        -int almoxarifadoOrigemId
        -int almoxarifadoDestinoId
        -int militarResponsavelId
        -int militarRecebedorId
        -int quantidade
        -decimal valorUnitario
        -decimal valorTotal
        -DateTime dataOperacao
        -Date dataPrevistaDevolucao
        -StatusOperacao status
        -TipoDocumento documentoTipo
        -string numeroDocumento
        -string observacoes
        -JSON detalhesAdicionais
        +criarOperacao()
        +validarOperacao()
        +executarMovimentacao()
        +atualizarStatus()
        +gerarDocumentacao()
        +calcularValorTotal()
    }

    class TipoMovimentacao {
        <<enumeration>>
        ENTRADA
        SAIDA
        TRANSFERENCIA
        DEVOLUCAO
        CONCESSAO
        CAUTELA
        BAIXA
        MANUTENCAO
        RESERVA
    }

    class StatusOperacao {
        <<enumeration>>
        PENDENTE
        EM_ANDAMENTO
        CONCLUIDA
        CANCELADA
        FALHA
    }
```



```

class DetalheOperacao {
    <<abstract>>
    -bigint operacaoId
    +registrarDetalhes()*
    +validarInformacoes()*
}

```

```

class EntradaDetalhe {
    -string fornecedor
    -string notaFiscal
    -Date dataNotaFiscal
    -decimal valorNotaFiscal
    -string condicoesArmazenamento
    -Date dataValidade
    -string lote
    -string certificadoQualidade
    +validarDocumentacao()
    +registrarFornecedor()
    +controlarValidade()
}

```

```

class SaidaDetalhe {
    -string finalidadeUso
    -string projetoOperacao
    -string justificativa
    -Date dataPrevistaRetorno
    -string responsavelExterno
    -string contatoExterno
    +definirFinalidade()
    +registrarResponsavel()
    +programarRetorno()
}

```

```

class CautelaDetalhe {
    -string numeroCautela
    -Date dataCautela
    -Date dataPrevistaDevolucao
    -string condicoesUso
    -string responsabilidades
    -string testemunha1
    -string testemunha2
    +gerarTermoCautela()
    +definirResponsabilidades()
    +registrarTestemunhas()
    +programarDevolucao()
}

```

```

Operacao ||-- TipoMovimentacao : "possui"
Operacao ||-- StatusOperacao : "possui"
DetalheOperacao <|-- EntradaDetalhe : "especializa"
DetalheOperacao <|-- SaidaDetalhe : "especializa"

```

```
DetalheOperacao <|-- CautelaDetalhe : "especializa"  
Operacao ||--o| DetalheOperacao : "detalha"
```

### 3.3. Diagrama de Componentes/Módulos

Este diagrama apresenta a arquitetura modular do sistema SIGA, mostrando como os diferentes componentes interagem entre si e suas dependências.

```
graph TB  
    subgraph "SISTEMA SIGA"  
        subgraph "Camada de Apresentação"  
            UI[Interface do Usuário]  
            API[API REST]  
            WEB[Interface Web]  
        end  
  
        subgraph "Camada de Negócio"  
            subgraph "Módulo de Configuração"  
                MC_UNIDADE[Gestão de Unidades]  
                MC_MILITAR[Gestão de Militares]  
                MC_SEGURANCA[Controle de Acesso]  
            end  
  
            subgraph "Módulo de Almoxarifado"  
                MA_ALMOX[Gestão de Almoxarifados]  
                MA_LOCAL[Controle de Localizações]  
                MA_ESTOQUE[Controle de Estoque]  
            end  
  
            subgraph "Módulo de Materiais"  
                MM_CATALOGO[Catálogo de Materiais]  
                MM_CATEGORIA[Gestão de Categorias]  
                MM_ESPECIALIZACAO[Especialização de Tipos]  
            end  
  
            subgraph "Módulo de Operações"  
                MO_MOVIMENTACAO[Controle de Movimentações]  
                MO_ENTRADA[Processamento de Entradas]  
                MO_SAIDA[Processamento de Saídas]  
                MO_TRANSFERENCIA[Gestão de Transferências]  
                MO_CAUTELA[Controle de Cautelas]  
            end  
  
            subgraph "Módulo de Auditoria"  
                MAU_HISTORICO[Histórico de Movimentações]  
                MAU_LOG[Sistema de Logs]  
                MAU_RELATORIO[Geração de Relatórios]  
            end  
        end
```

```

    subgraph "Módulo de Notificações"
        MN_ALERTAS[Sistema de Alertas]
        MN_NOTIFICACAO[Gestão de Notificações]
        MN_COMUNICACAO[Comunicação Integrada]
    end
end

subgraph "Camada de Dados"
    subgraph "Banco de Dados PostgreSQL"
        DB_CONFIG[Tabelas de Configuração]
        DB_ALMOX[Tabelas de Almoxarifado]
        DB_MATERIAL[Tabelas de Materiais]
        DB_OPERACAO[Tabelas de Operações]
        DB_HISTORICO[Tabelas de Histórico]
        DB_NOTIF[Tabelas de Notificações]
    end
end

    subgraph "Recursos Técnicos"
        PARTITION[Sistema de Particionamento]
        TRIGGER[Triggers Automatizados]
        INDEX[Índices Otimizados]
        LISTEN_NOTIFY[LISTEN/NOTIFY]
    end
end

subgraph "Camada de Infraestrutura"
    BACKUP[Sistema de Backup]
    MONITOR[Monitoramento]
    SECURITY[Segurança]
    PERFORMANCE[Otimização de Performance]
end
end

%% Relacionamentos entre Módulos
UI --> API
WEB --> API
API --> MC_UNIDADE
API --> MC_MILITAR
API --> MA_ALMOX
API --> MM_CATALOGO
API --> MO_MOVIMENTACAO

MC_UNIDADE --> DB_CONFIG
MC_MILITAR --> DB_CONFIG
MA_ALMOX --> DB_ALMOX
MA_ESTOQUE --> DB_ALMOX
MM_CATALOGO --> DB_MATERIAL
MO_MOVIMENTACAO --> DB_OPERACAO
MAU_HISTORICO --> DB_HISTORICO
MN_NOTIFICACAO --> DB_NOTIF

MO_MOVIMENTACAO --> TRIGGER

```

```
MAU_HISTORICO --> PARTITION
MN_COMUNICACAO --> LISTEN_NOTIFY
```

```
DB_OPERACAO --> PARTITION
DB_HISTORICO --> PARTITION
DB_NOTIF --> PARTITION
```

%% Dependências entre Módulos de Negócio

```
MA_ALMOX --> MC_UNIDADE
MA_ALMOX --> MC_MILITAR
MM_CATALOGO --> MM_CATEGORIA
MO_MOVIMENTACAO --> MA_ESTOQUE
MO_MOVIMENTACAO --> MM_CATALOGO
MO_MOVIMENTACAO --> MC_MILITAR
MAU_HISTORICO --> MO_MOVIMENTACAO
MN_ALERTAS --> MA_ESTOQUE
```

%% Estilos

```
classDef modulo fill:#e1f5fe,stroke:#01579b,stroke-width:2px
classDef dados fill:#f3e5f5,stroke:#4a148c,stroke-width:2px
classDef infra fill:#e8f5e8,stroke:#1b5e20,stroke-width:2px
```

**class**

```
MC_UNIDADE,MC_MILITAR,MC_SEGURANCA,MA_ALMOX,MA_LOCAL,MA_ESTOQUE,MM_CATALOGO
modulo
```

**class**

```
DB_CONFIG,DB_ALMOX,DB_MATERIAL,DB_OPERACAO,DB_HISTORICO,DB_NOTIF,PARTITION
dados
```

**class** BACKUP,MONITOR,SECURITY,PERFORMANCE infra

## Descrição dos Componentes

**Camada de Apresentação - Interface do Usuário:** Interface principal para interação com militares - **API REST:** Serviços web para integração e comunicação - **Interface Web:** Portal web para acesso remoto e relatórios

### Camada de Negócio - Módulos Funcionais

Módulo de Configuração - Gestão hierárquica de unidades CBM - Cadastro e controle de militares - Sistema de permissões e segurança

Módulo de Almoxarifado - Gestão de almoxarifados por unidade - Controle detalhado de localizações - Monitoramento de estoque em tempo real

Módulo de Materiais - Catálogo centralizado de materiais - Sistema de categorização dinâmica - Especialização por tipos específicos

Módulo de Operações - Controle de todas as movimentações - Processamento especializado por tipo - Validação e autorização de operações

Módulo de Auditoria - Histórico completo de movimentações - Sistema de logs detalhado - Geração de relatórios gerenciais

Módulo de Notificações - Alertas automáticos de estoque crítico - Sistema de notificações personalizadas - Comunicação integrada via PostgreSQL

**Camada de Dados** - Estrutura relacional otimizada - Sistema de particionamento por data - Triggers automatizados para integridade - Índices especializados para performance

**Camada de Infraestrutura** - Backup automatizado - Monitoramento de performance - Segurança integrada - Otimização contínua

## 4. Diagramas de Processo e Fluxo

### 4.1. Diagrama de Fluxo de Dados (DFD)

O Diagrama de Fluxo de Dados apresenta como as informações fluem através do sistema SIGA, mostrando os processos principais, armazenamentos de dados e entidades externas.

#### DFD Nível 0 - Visão Geral do Sistema

```
graph TD
  %% Entidades Externas
  MILITAR[👤 Militar]
  COMANDANTE[👤 Comandante]
  FORNECEDOR[🏢 Fornecedor]
  SISTEMA_EXTERNO[🔗 Sistema Externo]

  %% Processo Principal
  SIGA[📦 Sistema SIGA<br/>Gestão de Almoxarifado]

  %% Fluxos de Dados
  MILITAR -->|Solicitação de Material| SIGA
  MILITAR -->|Registro de Operação| SIGA
  MILITAR -->|Consulta de Estoque| SIGA

  COMANDANTE -->|Autorização de Operação| SIGA
  COMANDANTE -->|Solicitação de Relatório| SIGA

  FORNECEDOR -->|Nota Fiscal| SIGA
  FORNECEDOR -->|Documentação de Entrada| SIGA
```

```

SIGA -->|Relatório de Estoque| MILITAR
SIGA -->|Notificação de Status| MILITAR
SIGA -->|Termo de Cautela| MILITAR

SIGA -->|Relatório Gerencial| COMANDANTE
SIGA -->|Alertas de Estoque Crítico| COMANDANTE

SIGA -->|Confirmação de Recebimento| FORNECEDOR

SIGA -->|Dados de Integração| SISTEMA_EXTERNO
SISTEMA_EXTERNO -->|Informações Complementares| SIGA

```

#### %% Estilos

```

classDef entidade fill:#ffeb3b,stroke:#f57f17,stroke-width:
2px
classDef processo fill:#4caf50,stroke:#1b5e20,stroke-width:
3px

```

```

class MILITAR,COMANDANTE,FORNECEDOR,SISTEMA_EXTERNO entidade
class SIGA processo

```

## DFD Nível 1 - Processos Detalhados

```
graph TD
```

#### %% Entidades Externas

```

MILITAR[👤 Militar]
COMANDANTE[👤 Comandante]
FORNECEDOR[🏢 Fornecedor]

```

#### %% Processos Principais

```

P1[1.0<br/>Gestão de<br/>Configuração]
P2[2.0<br/>Controle de<br/>Almoxarifado]
P3[3.0<br/>Gestão de<br/>Materiais]
P4[4.0<br/>Processamento<br/>de Operações]
P5[5.0<br/>Controle de<br/>Estoque]
P6[6.0<br/>Sistema de<br/>Notificações]
P7[7.0<br/>Auditoria e<br/>Relatórios]

```

#### %% Armazenamentos de Dados

```

D1[(D1: Unidades e Militares)]
D2[(D2: Almoxarifados)]
D3[(D3: Catálogo de Materiais)]
D4[(D4: Operações)]
D5[(D5: Estoque Atual)]
D6[(D6: Histórico)]
D7[(D7: Notificações)]

```

#### %% Fluxos de Entrada

```

MILITAR -->|Dados Pessoais| P1

```

MILITAR -->|Solicitação de Material| P4

MILITAR -->|Consulta de Estoque| P5

COMANDANTE -->|Autorização| P4

COMANDANTE -->|Solicitação de Relatório| P7

FORNECEDOR -->|Nota Fiscal| P4

FORNECEDOR -->|Documentação| P4

## %% Fluxos entre Processos

P1 --> D1

P2 --> D2

P3 --> D3

P4 --> D4

P4 --> P5

P5 --> D5

P4 --> D6

P5 --> P6

P6 --> D7

## %% Consultas entre Processos

P4 -->|Consulta Militar| D1

P4 -->|Consulta Almoxarifado| D2

P4 -->|Consulta Material| D3

P5 -->|Consulta Operação| D4

P6 -->|Consulta Estoque| D5

P7 -->|Consulta Histórico| D6

## %% Fluxos de Saída

P5 -->|Status de Estoque| MILITAR

P6 -->|Notificações| MILITAR

P4 -->|Termo de Cautela| MILITAR

P7 -->|Relatórios| COMANDANTE

P6 -->|Alertas Críticos| COMANDANTE

P4 -->|Confirmação| FORNECEDOR

## %% Estilos

classDef entidade fill:#ffeb3b,stroke:#f57f17,stroke-width:  
2px

classDef processo fill:#4caf50,stroke:#1b5e20,stroke-width:  
2px

classDef dados fill:#2196f3,stroke:#0d47a1,stroke-width:2px

class MILITAR,COMANDANTE,FORNECEDOR entidade

class P1,P2,P3,P4,P5,P6,P7 processo

class D1,D2,D3,D4,D5,D6,D7 dados

## DFD Nível 2 - Processo de Operações Detalhado

graph TD

%% Entidades

MILITAR[  Militar]

FORNECEDOR[  Fornecedor]

%% Subprocessos de Operações

P41[4.1<br/>Validar<br/>Solicitação]

P42[4.2<br/>Processar<br/>Entrada]

P43[4.3<br/>Processar<br/>Saída]

P44[4.4<br/>Processar<br/>Transferência]

P45[4.5<br/>Processar<br/>Cautela]

P46[4.6<br/>Atualizar<br/>Estoque]

P47[4.7<br/>Gerar<br/>Documentação]

%% Armazenamentos

D1[(D1: Militares)]

D2[(D2: Almoxarifados)]

D3[(D3: Materiais)]

D4[(D4: Operações)]

D5[(D5: Estoque)]

D6[(D6: Histórico)]

%% Fluxos de Entrada

MILITAR -->|Solicitação| P41

FORNECEDOR -->|Nota Fiscal| P42

%% Fluxos entre Subprocessos

P41 -->|Solicitação Validada| P43

P41 -->|Solicitação Validada| P44

P41 -->|Solicitação Validada| P45

P42 -->|Dados de Entrada| P46

P43 -->|Dados de Saída| P46

P44 -->|Dados de Transferência| P46

P45 -->|Dados de Cautela| P46

P42 --> P47

P43 --> P47

P44 --> P47

P45 --> P47

%% Armazenamento de Dados

P41 --> D4

P42 --> D4

P43 --> D4

P44 --> D4

P45 --> D4

P46 --> D5

P46 --> D6



```
P47 --> D4
```

#### %% Consultas

```
P41 -->|Verifica Militar| D1
```

```
P41 -->|Verifica Almoxarifado| D2
```

```
P41 -->|Verifica Material| D3
```

```
P46 -->|Consulta Estoque Atual| D5
```

#### %% Fluxos de Saída

```
P47 -->|Documentos| MILITAR
```

```
P47 -->|Confirmação| FORNECEDOR
```

#### %% Estilos

```
classDef entidade fill:#ffeb3b,stroke:#f57f17,stroke-width:
```

```
2px
```

```
classDef processo fill:#4caf50,stroke:#1b5e20,stroke-width:
```

```
2px
```

```
classDef dados fill:#2196f3,stroke:#0d47a1,stroke-width:2px
```

```
class MILITAR,FORNECEDOR entidade
```

```
class P41,P42,P43,P44,P45,P46,P47 processo
```

```
class D1,D2,D3,D4,D5,D6 dados
```

## 4.2. Diagrama de Swimlanes de Processo

Este diagrama detalha o processo de movimentação de material dividindo as responsabilidades por "lanes" (raias), mostrando claramente quem faz o quê em cada etapa do processo.

### Processo de Entrada de Material

```
graph TD
    subgraph "Fornecedor"
        F1[Emitir Nota Fiscal]
        F2[Entregar Material]
        F3[Aguardar Confirmação]
    end

    subgraph "Militar Recebedor"
        M1[Receber Solicitação<br/>de Entrada]
        M2[Verificar Documentação]
        M3[Inspeccionar Material]
        M4[Registrar Entrada<br/>no Sistema]
        M5[Definir Localização]
        M6[Confirmar Recebimento]
    end

    subgraph "Sistema SIGA"
        S1[Validar Dados]
    end
```

```

        S2[Gerar Número<br/>de Operação]
        S3[Atualizar Estoque]
        S4[Registrar Histórico]
        S5[Gerar Notificações]
        S6[Emitir Comprovante]
    end

    subgraph "Almoxarifado"
        A1[Verificar Capacidade]
        A2[Preparar Localização]
        A3[Armazenar Material]
        A4[Atualizar Ocupação]
    end

    %% Fluxo do Processo
    F1 --> M1
    F2 --> M2
    M2 --> M3
    M3 --> M4
    M4 --> S1
    S1 --> S2
    S2 --> M5
    M5 --> A1
    A1 --> A2
    A2 --> A3
    A3 --> S3
    S3 --> S4
    S4 --> S5
    S5 --> S6
    S6 --> M6
    M6 --> F3
    A3 --> A4

    %% Estilos
    classDef fornecedor fill:#ffcdd2,stroke:#c62828,stroke-
width:2px
    classDef militar fill:#c8e6c9,stroke:#2e7d32,stroke-width:
2px
    classDef sistema fill:#bbdefb,stroke:#1565c0,stroke-width:
2px
    classDef almoxarifado fill:#fff3e0,stroke:#ef6c00,stroke-
width:2px

    class F1,F2,F3 fornecedor
    class M1,M2,M3,M4,M5,M6 militar
    class S1,S2,S3,S4,S5,S6 sistema
    class A1,A2,A3,A4 almoxarifado

```

## Processo de Saída de Material (Cautela)

```
graph TD
    subgraph "Militar Solicitante"
        MS1[Solicitar Material]
        MS2[Preencher Justificativa]
        MS3[Aguardar Autorização]
        MS4[Receber Material]
        MS5[Assinar Termo<br/>de Cautela]
    end

    subgraph "Militar Responsável Almojarifado"
        MR1[Receber Solicitação]
        MR2[Verificar Disponibilidade]
        MR3[Validar Justificativa]
        MR4[Solicitar Autorização]
        MR5[Preparar Material]
        MR6[Entregar Material]
        MR7[Registrar Saída]
    end

    subgraph "Comandante/Autoridade"
        C1[Analisar Solicitação]
        C2[Autorizar ou Negar]
        C3[Registrar Decisão]
    end

    subgraph "Sistema SIGA"
        SS1[Validar Solicitação]
        SS2[Verificar Estoque]
        SS3[Gerar Termo<br/>de Cautela]
        SS4[Registrar Operação]
        SS5[Atualizar Estoque]
        SS6[Programar Alerta<br/>de Devolução]
        SS7[Gerar Notificações]
    end

    subgraph "Almojarifado"
        AS1[Localizar Material]
        AS2[Verificar Estado]
        AS3[Separar Material]
        AS4[Atualizar Localização]
    end

    %% Fluxo do Processo
    MS1 --> MR1
    MR1 --> SS1
    SS1 --> SS2
    SS2 --> MR2
    MR2 --> MR3
    MR3 --> MR4
```

```

MR4 --> C1
C1 --> C2
C2 --> C3
C3 --> MR5
MR5 --> AS1
AS1 --> AS2
AS2 --> AS3
AS3 --> SS3
SS3 --> MR6
MR6 --> MS4
MS4 --> MS5
MS5 --> MR7
MR7 --> SS4
SS4 --> SS5
SS5 --> AS4
AS4 --> SS6
SS6 --> SS7

```

```

%% Fluxos de Decisão
C2 -->|Negado| MS3
SS2 -->|Sem Estoque| MR2

```

```

%% Estilos

```

```

classDef solicitante fill:#e1f5fe,stroke:#01579b,stroke-
width:2px
classDef responsavel fill:#c8e6c9,stroke:#2e7d32,stroke-
width:2px
classDef comandante fill:#fce4ec,stroke:#ad1457,stroke-
width:2px
classDef sistema fill:#bbdefb,stroke:#1565c0,stroke-width:
2px
classDef almoxarifado fill:#fff3e0,stroke:#ef6c00,stroke-
width:2px

```

```

class MS1,MS2,MS3,MS4,MS5 solicitante
class MR1,MR2,MR3,MR4,MR5,MR6,MR7 responsavel
class C1,C2,C3 comandante
class SS1,SS2,SS3,SS4,SS5,SS6,SS7 sistema
class AS1,AS2,AS3,AS4 almoxarifado

```

### 4.3. Diagrama de Sequência UML

Os diagramas de sequência mostram a interação temporal entre os diferentes componentes do sistema SIGA, incluindo triggers, funções e notificações automáticas.

#### Sequência: Registro de Operação de Entrada

```

sequenceDiagram
    participant M as Militar

```

participant UI as Interface  
participant API as API SIGA  
participant DB as Banco de Dados  
participant TG as Triggers  
participant NT as Sistema Notificação

M->>UI: Solicita registro de entrada  
UI->>API: POST /operacao/entrada  
API->>DB: Validar militar e almoxarifado  
DB->>API: Dados validados

API->>DB: INSERT INTO operacao  
activate TG  
DB->>TG: Trigger particionamento  
TG->>DB: Inserir na partição correta

DB->>TG: Trigger atualizar\_estoque  
TG->>DB: UPDATE estoque\_atual  
TG->>DB: INSERT INTO historico\_estoque

DB->>TG: Trigger verificar\_estoque\_critico  
TG->>DB: Verificar limites mínimos

alt Estoque abaixo do mínimo  
    TG->>NT: NOTIFY estoque\_critico  
    NT->>DB: INSERT INTO notificacao  
    NT->>M: Enviar alerta

end

deactivate TG  
DB->>API: Operação registrada  
API->>UI: Confirmação de sucesso  
UI->>M: Exibir comprovante

Note over DB,NT: Sistema automatizado<br/>de notificações

## Sequência: Processo de Cautela com Autorização

sequenceDiagram

participant MS as Militar Solicitante  
participant MR as Militar Responsável  
participant CMD as Comandante  
participant API as API SIGA  
participant DB as Banco de Dados  
participant DOC as Gerador Documentos  
participant NT as Notificações

MS->>API: Solicitar cautela de material  
API->>DB: Verificar disponibilidade  
DB->>API: Material disponível

```

API->>DB: Criar operação PENDENTE
DB-->>API: Operação criada

API->>NT: Notificar responsável
NT->>MR: Solicitação pendente

MR->>API: Analisar solicitação
API->>DB: Buscar detalhes
DB-->>API: Dados da solicitação

MR->>API: Encaminhar para autorização
API->>NT: Notificar comandante
NT->>CMD: Solicitação para autorização

CMD->>API: Autorizar operação
API->>DB: UPDATE status = 'AUTORIZADA'

API->>NT: Notificar responsável
NT->>MR: Operação autorizada

MR->>API: Processar cautela
API->>DB: Verificar estoque atual
DB-->>API: Estoque confirmado

API->>DB: UPDATE estoque (reservar)
API->>DOC: Gerar termo de cautela
DOC-->>API: Documento gerado

API->>DB: UPDATE status = 'EM_ANDAMENTO'
API->>NT: Notificar solicitante
NT->>MS: Cautela pronta para retirada

MS->>API: Confirmar retirada
API->>DB: UPDATE status = 'CONCLUIDA'
API->>DB: Atualizar estoque final

API->>NT: Programar lembrete devolução
NT->>DB: Agendar notificação futura

```

## Sequência: Sistema de Particionamento Automático

```

sequenceDiagram
    participant APP as Aplicação
    participant DB as PostgreSQL
    participant PT as Trigger Particionamento
    participant JOB as Job Scheduler
    participant ADM as Administrador

    Note over DB,JOB: Processo de criação<br/>automática de

```

## partições

```
JOB->>DB: Executar job mensal
DB->>PT: Verificar próximas partições
PT->>DB: Consultar partições existentes
DB->>PT: Lista de partições
```

```
PT->>PT: Calcular próximas datas
```

```
loop Para cada tabela particionada
  PT->>DB: CREATE TABLE operacao_p2025_07
  PT->>DB: ALTER TABLE ATTACH PARTITION
  PT->>DB: CREATE INDEX ON partition
end
```

```
PT->>DB: INSERT INTO log_particionamento
DB->>JOB: Partições criadas
```

```
JOB->>ADM: Relatório de particionamento
```

Note **over** APP,DB: Inserção automática<br/>em partição correta

```
APP->>DB: INSERT INTO operacao
DB->>PT: Trigger before insert
PT->>PT: Calcular partição por data
PT->>DB: Redirecionar para partição
DB->>APP: Registro inserido
```

## Sequência: Sistema LISTEN/NOTIFY para Notificações

### sequenceDiagram

```
participant OP as Operação
participant TG as Trigger
participant PG as PostgreSQL
participant LS as Listener Service
participant NT as Notificação Service
participant USR as Usuário

OP->>PG: INSERT/UPDATE operacao
PG->>TG: Executar trigger

TG->>TG: Verificar condições

alt Estoque crítico detectado
  TG->>PG: NOTIFY 'estoque_critico'
  PG->>LS: Evento recebido
  LS->>NT: Processar notificação
  NT->>PG: INSERT INTO notificacao
  NT->>USR: Enviar alerta (email/SMS)
```

end

```
alt Material vencendo
  TG->>PG: NOTIFY 'material_vencimento'
  PG->>LS: Evento recebido
  LS->>NT: Processar alerta
  NT->>USR: Notificar vencimento
```

end

```
alt Cautela vencida
  TG->>PG: NOTIFY 'cautela_vencida'
  PG->>LS: Evento recebido
  LS->>NT: Processar cobrança
  NT->>USR: Cobrar devolução
```

end

Note over PG,USR: Sistema assíncrono<br/>de notificações em tempo real

## Descrição dos Fluxos de Processo

**Fluxo de Dados Principal** O sistema SIGA processa informações em múltiplas camadas, desde a entrada de dados pelos usuários até o armazenamento e geração de relatórios. Os dados fluem através de processos de validação, transformação e persistência.

**Processo de Operações** Cada tipo de operação (entrada, saída, transferência, cautela) segue um fluxo específico com validações apropriadas, mas todos convergem para a atualização do estoque e geração de histórico.

**Sistema de Autorização** Operações críticas passam por um fluxo de autorização hierárquica, garantindo que apenas pessoal autorizado possa aprovar movimentações importantes.

**Automação de Processos** O sistema utiliza triggers e jobs automatizados para: -  
Particionamento automático de tabelas por data - Atualização de estoque em tempo real  
- Geração de notificações baseadas em eventos - Manutenção de histórico completo

**Comunicação Assíncrona** O sistema PostgreSQL LISTEN/NOTIFY permite comunicação em tempo real entre componentes, garantindo que notificações sejam enviadas imediatamente quando eventos críticos ocorrem.



## 5. Diagramas Técnicos Especializados

### 5.1. Diagrama de Particionamento

O sistema SIGA utiliza particionamento por faixa de datas para otimizar performance e facilitar manutenção das tabelas com grande volume de dados. Este diagrama mostra a estrutura de particionamento implementada.

#### Estrutura de Particionamento por Tabela

```
graph TD
    subgraph "Tabela OPERACAO (Particionada por data_operacao)"
        OP_MASTER[operacao<br/>Tabela Master<br/>PARTITION BY RANGE data_operacao]
        OP_2025_06[operacao_p2025_06<br/>2025-06-01 a 2025-06-30<br/>Partição Mensal]
        OP_2025_07[operacao_p2025_07<br/>2025-07-01 a 2025-07-31<br/>Partição Mensal]
        OP_2025_08[operacao_p2025_08<br/>2025-08-01 a 2025-08-31<br/>Partição Mensal]
        OP_2025_09[operacao_p2025_09<br/>2025-09-01 a 2025-09-30<br/>Partição Mensal]
        OP_DEFAULT[operacao_default<br/>Dados fora do range<br/>Partição Padrão]
        OP_MASTER --> OP_2025_06
        OP_MASTER --> OP_2025_07
        OP_MASTER --> OP_2025_08
        OP_MASTER --> OP_2025_09
        OP_MASTER --> OP_DEFAULT
    end

    subgraph "Tabela HISTORICO_ESTOQUE (Particionada por data_movimentacao)"
        HE_MASTER[historico_estoque<br/>Tabela Master<br/>PARTITION BY RANGE data_movimentacao]
        HE_2025_Q2[historico_estoque_p2025_q2<br/>2025-04-01 a 2025-06-30<br/>Partição Trimestral]
        HE_2025_Q3[historico_estoque_p2025_q3<br/>2025-07-01 a 2025-09-30<br/>Partição Trimestral]
        HE_2025_Q4[historico_estoque_p2025_q4<br/>2025-10-01 a 2025-12-31<br/>Partição Trimestral]
        HE_2026_Q1[historico_estoque_p2026_q1<br/>2026-01-01 a 2026-03-31<br/>Partição Trimestral]
        HE_MASTER --> HE_2025_Q2
        HE_MASTER --> HE_2025_Q3
```

```

HE_MASTER --> HE_2025_Q4
HE_MASTER --> HE_2026_Q1
end

subgraph "Tabela LOG_AUDITORIA (Particionada por
data_operacao)"
    LA_MASTER[log_auditoria<br/>Tabela Master<br/>PARTITION
BY RANGE data_operacao]

    LA_2025_06[log_auditoria_p2025_06<br/>2025-06-01 a
2025-06-30<br/>Partição Mensal]
    LA_2025_07[log_auditoria_p2025_07<br/>2025-07-01 a
2025-07-31<br/>Partição Mensal]
    LA_2025_08[log_auditoria_p2025_08<br/>2025-08-01 a
2025-08-31<br/>Partição Mensal]

    LA_MASTER --> LA_2025_06
    LA_MASTER --> LA_2025_07
    LA_MASTER --> LA_2025_08
end

subgraph "Tabela NOTIFICACAO (Particionada por data_envio)"
    NT_MASTER[notificacao<br/>Tabela Master<br/>PARTITION
BY RANGE data_envio]

    NT_2025_S1[notificacao_p2025_s1<br/>2025-01-01 a
2025-06-30<br/>Partição Semestral]
    NT_2025_S2[notificacao_p2025_s2<br/>2025-07-01 a
2025-12-31<br/>Partição Semestral]
    NT_2026_S1[notificacao_p2026_s1<br/>2026-01-01 a
2026-06-30<br/>Partição Semestral]

    NT_MASTER --> NT_2025_S1
    NT_MASTER --> NT_2025_S2
    NT_MASTER --> NT_2026_S1
end

%% Estilos
classDef master fill:#ffcd22,stroke:#c62828,stroke-width:3px
classDef partition fill:#c8e6c9,stroke:#2e7d32,stroke-width:
2px
classDef default fill:#fff3e0,stroke:#ef6c00,stroke-width:
2px

class OP_MASTER,HE_MASTER,LA_MASTER,NT_MASTER master
class
OP_2025_06,OP_2025_07,OP_2025_08,OP_2025_09,HE_2025_Q2,HE_2025_Q3,HE_2025_Q4
partition
class OP_DEFAULT default

```

## Fluxo de Criação Automática de Partições

```
graph TD
    START[Início do Processo]

    subgraph "Job Scheduler Mensal"
        JOB[Job de Manutenção<br/>Executado todo dia 1º]
        CHECK[Verificar Próximas<br/>Partições Necessárias]
        CALC[Calcular Datas<br/>dos Próximos Períodos]
    end

    subgraph "Criação de Partições"
        CREATE_OP[Criar Partições<br/>OPERACAO]
        CREATE_HE[Criar Partições<br/>HISTORICO_ESTOQUE]
        CREATE_LA[Criar Partições<br/>LOG_AUDITORIA]
        CREATE_NT[Criar Partições<br/>NOTIFICACAO]
    end

    subgraph "Configuração de Índices"
        IDX_OP[Criar Índices<br/>Partições OPERACAO]
        IDX_HE[Criar Índices<br/>Partições HISTORICO]
        IDX_LA[Criar Índices<br/>Partições LOG]
        IDX_NT[Criar Índices<br/>Partições NOTIFICACAO]
    end

    subgraph "Manutenção"
        CLEAN[Limpar Partições<br/>Antigas (> 5 anos)]
        LOG[Registrar Log<br/>de Manutenção]
        NOTIFY[Notificar<br/>Administradores]
    end

    START --> JOB
    JOB --> CHECK
    CHECK --> CALC
    CALC --> CREATE_OP
    CREATE_OP --> CREATE_HE
    CREATE_HE --> CREATE_LA
    CREATE_LA --> CREATE_NT

    CREATE_NT --> IDX_OP
    IDX_OP --> IDX_HE
    IDX_HE --> IDX_LA
    IDX_LA --> IDX_NT

    IDX_NT --> CLEAN
    CLEAN --> LOG
    LOG --> NOTIFY

%% Estilos
classDef job fill:#e3f2fd,stroke:#0277bd,stroke-width:2px
classDef create fill:#e8f5e8,stroke:#2e7d32,stroke-width:2px
```

```

classDef index fill:#fff3e0,stroke:#f57c00,stroke-width:2px
classDef maint fill:#fce4ec,stroke:#c2185b,stroke-width:2px

class JOB,CHECK,CALC job
class CREATE_OP,CREATE_HE,CREATE_LA,CREATE_NT create
class IDX_OP,IDX_HE,IDX_LA,IDX_NT index
class CLEAN,LOG,NOTIFY maint

```

## Estratégia de Particionamento por Tabela

```

graph LR
    subgraph "Critérios de Particionamento"
        VOLUME[Volume de Dados]
        ACESSO[Padrão de Acesso]
        MANUTENCAO[Facilidade de Manutenção]
        PERFORMANCE[Performance de Consulta]
    end

    subgraph "Tabelas e Estratégias"
        OP[OP<br/>OPERACAO<br/>📊 Alto Volume<br/>🔍 Consulta por Data<br/>📅 Partição Mensal]
        HE[HE<br/>HISTORICO_ESTOQUE<br/>📊 Volume Muito Alto<br/>🔍 Consulta Histórica<br/>📅 Partição Trimestral]
        LA[LA<br/>LOG_AUDITORIA<br/>📊 Volume Alto<br/>🔍 Auditoria por Período<br/>📅 Partição Mensal]
        NT[NT<br/>NOTIFICACAO<br/>📊 Volume Médio<br/>🔍 Consulta Recente<br/>📅 Partição Semestral]
        EST[EST<br/>ESTOQUE_ATUAL<br/>📊 Volume Baixo<br/>🔍 Consulta Frequente<br/>❌ Não Particionada]
        MAT[MAT<br/>MATERIAL_BASE<br/>📊 Volume Baixo<br/>🔍 Consulta Constante<br/>❌ Não Particionada]
    end

    VOLUME --> OP
    VOLUME --> HE
    ACESSO --> LA
    ACESSO --> NT
    PERFORMANCE --> EST
    PERFORMANCE --> MAT

    %% Estilos
    classDef criteria fill:#elf5fe,stroke:#01579b,stroke-width:2px
    classDef partitioned fill:#c8e6c9,stroke:#2e7d32,stroke-width:2px

```

```

classDef nonpartitioned fill:#ffecb3,stroke:#f57c00,stroke-
width:2px

class VOLUME,ACESSO,MANUTENCAO,PERFORMANCE criteria
class OP,HE,LA,NT partitioned
class EST,MAT nonpartitioned

```

## 5.2. Diagrama de Índices e Otimização

Este diagrama apresenta a estratégia de indexação implementada no sistema SIGA para otimizar consultas e garantir performance adequada.

### Mapa de Índices por Tabela

```

graph TD
    subgraph "Tabela OPERACAO"
        OP_TABLE[operacao]
        OP_PK[🔑 PK: id_operacao<br/>B-tree Index]
        OP_DATA[📅 17 idx_operacao_data<br/>B-tree: data_operacao]
        OP_MATERIAL[📦 idx_operacao_material<br/>B-tree:
material_id]
        OP_MILITAR[👤 idx_operacao_militar<br/>B-tree:
militar_responsavel_id]
        OP_STATUS[📊 idx_operacao_status<br/>B-tree: status]
        OP_TIPO[🔄 idx_operacao_tipo<br/>B-tree:
tipo_movimentacao]
        OP_COMPOSTO[🔗 idx_operacao_composto<br/>B-tree:
(material_id, data_operacao)]
        OP_JSONB[📋 idx_operacao_detalhes<br/>GIN:
detalhes_adicionais]

        OP_TABLE --> OP_PK
        OP_TABLE --> OP_DATA
        OP_TABLE --> OP_MATERIAL
        OP_TABLE --> OP_MILITAR
        OP_TABLE --> OP_STATUS
        OP_TABLE --> OP_TIPO
        OP_TABLE --> OP_COMPOSTO
        OP_TABLE --> OP_JSONB
    end

    subgraph "Tabela ESTOQUE_ATUAL"
        EST_TABLE[estoque_atual]
        EST_PK[🔑 PK: id_estoque<br/>B-tree Index]
        EST_UNIQUE[🔒 UK: (material_id, almoxarifado_id,
localizacao_id)<br/>Unique B-tree]
        EST_MATERIAL[📦 idx_estoque_material<br/>B-tree:
material_id]
        EST_ALMOX[🏠 idx_estoque_almoxarifado<br/>B-tree:

```

```

almoxarifado_id]
    EST_QTD[📊 idx_estoque_quantidade<br/>B-tree:
quantidade_disponivel]
    EST_CRITICO[⚠️ idx_estoque_critico<br/>Partial: WHERE
quantidade_disponivel < estoque_minimo]

    EST_TABLE --> EST_PK
    EST_TABLE --> EST_UNIQUE
    EST_TABLE --> EST_MATERIAL
    EST_TABLE --> EST_ALMOX
    EST_TABLE --> EST_QTD
    EST_TABLE --> EST_CRITICO
end

subgraph "Tabela MATERIAL_BASE"
    MAT_TABLE[material_base]
    MAT_PK[🔑 PK: id_material<br/>B-tree Index]
    MAT_CODIGO[📝 UK: codigo_material<br/>Unique B-tree]
    MAT_NOME[🔥 idx_material_nome<br/>B-tree: nome_material]
    MAT_CATEGORIA[📁 idx_material_categoria<br/>B-tree:
categoria_id]
    MAT_SITUACAO[📊 idx_material_situacao<br/>B-tree:
situacao]

    MAT_ATIVO[✅ idx_material_ativo<br/>Partial: WHERE ativo = true]
    MAT_TEXTO[🔍 idx_material_busca<br/>GIN:
to_tsvector(nome_material || ' ' || descricao_material)]
    MAT_JSONB[📋 idx_material_atributos<br/>GIN:
atributos_adicionais]

    MAT_TABLE --> MAT_PK
    MAT_TABLE --> MAT_CODIGO
    MAT_TABLE --> MAT_NOME
    MAT_TABLE --> MAT_CATEGORIA
    MAT_TABLE --> MAT_SITUACAO
    MAT_TABLE --> MAT_ATIVO
    MAT_TABLE --> MAT_TEXTO
    MAT_TABLE --> MAT_JSONB
end

subgraph "Tabela HISTORICO_ESTOQUE"
    HIST_TABLE[historico_estoque]
    HIST_PK[🔑 PK: id_historico<br/>B-tree Index]
    HIST_DATA[📅 17 idx_historico_data<br/>B-tree:
data_movimentacao]
    HIST_MATERIAL[📦 idx_historico_material<br/>B-tree:
material_id]
    HIST_OPERACAO[🔄 idx_historico_operacao<br/>B-tree:
operacao_id]
    HIST_COMPOSTO[🔗 idx_historico_composto<br/>B-tree:
(material_id, data_movimentacao)]

```

```

HIST_TABLE --> HIST_PK
HIST_TABLE --> HIST_DATA
HIST_TABLE --> HIST_MATERIAL
HIST_TABLE --> HIST_OPERACAO
HIST_TABLE --> HIST_COMPOSTO
end

%% Estilos
classDef table fill:#e3f2fd,stroke:#1565c0,stroke-width:3px
classDef pk fill:#ffcdd2,stroke:#c62828,stroke-width:2px
classDef unique fill:#f8bbd9,stroke:#ad1457,stroke-width:2px
classDef btree fill:#c8e6c9,stroke:#2e7d32,stroke-width:2px
classDef gin fill:#fff3e0,stroke:#f57c00,stroke-width:2px
classDef partial fill:#e1f5fe,stroke:#0277bd,stroke-width:
2px

class OP_TABLE,EST_TABLE,MAT_TABLE,HIST_TABLE table
class OP_PK,EST_PK,MAT_PK,HIST_PK pk
class EST_UNIQUE,MAT_CODIGO unique
class
OP_DATA,OP_MATERIAL,OP_MILITAR,OP_STATUS,OP_TIPO,OP_COMPOSTO,EST_MATERIAL,
btree
class OP_JSONB,MAT_TEXTO,MAT_JSONB gin
class EST_CRITICO,MAT_ATIVO partial

```

## Estratégia de Otimização de Consultas

```

graph TD
  subgraph "Consultas Frequentes"
    Q1[Consultar Estoque<br/>por Material]
    Q2[Histórico de<br/>Movimentações]
    Q3[Operações por<br/>Período]
    Q4[Busca Textual<br/>de Materiais]
    Q5[Alertas de<br/>Estoque Crítico]
    Q6[Relatórios<br/>Gerenciais]
  end

  subgraph "Índices Otimizados"
    I1[idx_estoque_material<br/>B-tree: material_id]
    I2[idx_historico_composto<br/>B-tree: (material_id,
data)]
    I3[idx_operacao_data<br/>B-tree: data_operacao]
    I4[idx_material_busca<br/>GIN: texto completo]
    I5[idx_estoque_critico<br/>Partial: quantidade < mínimo]
    I6[idx_operacao_composto<br/>B-tree: (material_id,
data)]
  end

  subgraph "Técnicas de Otimização"
    T1[Particionamento<br/>por Data]
  end

```

```

T2[Índices Parciais<br/>para Filtros Comuns]
T3[Índices GIN<br/>para JSONB]
T4[Índices Compostos<br/>para Consultas Complexas]
T5[Estatísticas<br/>Atualizadas]
T6[Vacuum e Analyze<br/>Automatizados]
end

Q1 --> I1
Q2 --> I2
Q3 --> I3
Q4 --> I4
Q5 --> I5
Q6 --> I6

I1 --> T4
I2 --> T1
I3 --> T1
I4 --> T3
I5 --> T2
I6 --> T4

T1 --> T5
T2 --> T5
T3 --> T6
T4 --> T6

%% Estilos
classDef query fill:#e8f5e8,stroke:#2e7d32,stroke-width:2px
classDef index fill:#fff3e0,stroke:#f57c00,stroke-width:2px
classDef technique fill:#e3f2fd,stroke:#1565c0,stroke-width:
2px

class Q1,Q2,Q3,Q4,Q5,Q6 query
class I1,I2,I3,I4,I5,I6 index
class T1,T2,T3,T4,T5,T6 technique

```

### 5.3. Diagrama de Implantação (Deployment)

Este diagrama representa a topologia do ambiente de implantação do sistema SIGA, incluindo componentes de infraestrutura, segurança e integração.

#### Arquitetura de Implantação Completa

```

graph TB
    subgraph "Zona DMZ"
        LB[Load Balancer<br/>HAProxy/Nginx<br/>SSL Termination]
        FW[Firewall<br/>Regras de Acesso<br/>WAF]
    end
end

```



```

subgraph "Camada de Aplicação"
  subgraph "Cluster Web Servers"
    WEB1[Web Server 1<br/>Apache/Nginx<br/>Interface
SIGA]
    WEB2[Web Server 2<br/>Apache/Nginx<br/>Interface
SIGA]
  end

  subgraph "Cluster API Servers"
    API1[API Server 1<br/>Python/Flask<br/>REST API]
    API2[API Server 2<br/>Python/Flask<br/>REST API]
  end
end

subgraph "Camada de Dados"
  subgraph "PostgreSQL Cluster"
    DB_MASTER[PostgreSQL Master<br/>Servidor
Principal<br/>Read/Write]
    DB_REPLICA1[PostgreSQL Replica 1<br/>Servidor
Secundário<br/>Read Only]
    DB_REPLICA2[PostgreSQL Replica 2<br/>Servidor
Secundário<br/>Read Only]
  end

  subgraph "Armazenamento"
    STORAGE[Storage SAN/NAS<br/>Dados Persistentes<br/>
>Backup Automático]
  end

  subgraph "Serviços de Apoio"
    subgraph "Monitoramento"
      MONITOR[Prometheus/Grafana<br/>Monitoramento<br/>
>Métricas e Alertas]
      LOG[ELK Stack<br/>Centralização de Logs<br/>
>Elasticsearch/Logstash/Kibana]
    end

    subgraph "Backup e Segurança"
      BACKUP[Sistema de Backup<br/>Backup Incremental<br/>
>Retenção 7 anos]
      VAULT[HashiCorp Vault<br/>Gerenciamento<br/>de
Secrets]
    end

    subgraph "Notificações"
      NOTIFY[Serviço de Notificação<br/>Email/SMS
Gateway<br/>LISTEN/NOTIFY Handler]
      QUEUE[Message Queue<br/>Redis/RabbitMQ<br/>
>Processamento Assíncrono]
    end
  end
end

```

```

subgraph "Integração Externa"
    LDAP[Active Directory/LDAP<br/>Autenticação<br/>Centralizada]
    ERP[Sistema ERP<br/>Integração Financeira<br/>API REST]
    SIAFI[SIAFI<br/>Sistema Governamental<br/>Web Services]
end

subgraph "Clientes"
    DESKTOP[Estações de Trabalho<br/>CBM Units<br/>Browsers]
    MOBILE[Dispositivos Móveis<br/>Tablets/Smartphones<br/>App Mobile]
    KIOSK[Terminais de Consulta<br/>Almoxarifados<br/>Interface Simplificada]
end

%% Conexões de Rede
DESKTOP --> FW
MOBILE --> FW
KIOSK --> FW

FW --> LB
LB --> WEB1
LB --> WEB2
LB --> API1
LB --> API2

WEB1 --> API1
WEB2 --> API2
API1 --> DB_MASTER
API2 --> DB_MASTER

DB_MASTER -->|Replicação| DB_REPLICA1
DB_MASTER -->|Replicação| DB_REPLICA2

DB_MASTER --> STORAGE
DB_REPLICA1 --> STORAGE
DB_REPLICA2 --> STORAGE

API1 --> LDAP
API2 --> LDAP
API1 --> VAULT
API2 --> VAULT

DB_MASTER --> NOTIFY
NOTIFY --> QUEUE

API1 --> ERP
API2 --> SIAFI

MONITOR --> WEB1
MONITOR --> WEB2

```

```
MONITOR --> API1
MONITOR --> API2
MONITOR --> DB_MASTER
```

```
LOG --> WEB1
LOG --> WEB2
LOG --> API1
LOG --> API2
```

```
BACKUP --> STORAGE
```

%% Estilos

```
classDef dmz fill:#ffcdd2,stroke:#c62828,stroke-width:2px
classDef app fill:#c8e6c9,stroke:#2e7d32,stroke-width:2px
classDef data fill:#bbdefb,stroke:#1565c0,stroke-width:2px
classDef support fill:#fff3e0,stroke:#f57c00,stroke-width:
2px
classDef external fill:#f3e5f5,stroke:#7b1fa2,stroke-width:
2px
classDef client fill:#e8f5e8,stroke:#388e3c,stroke-width:2px

class LB,FW dmz
class WEB1,WEB2,API1,API2 app
class DB_MASTER,DB_REPLICA1,DB_REPLICA2,STORAGE data
class MONITOR,LOG,BACKUP,VAULT,NOTIFY,QUEUE support
class LDAP,ERP,SIAFI external
class DESKTOP,MOBILE,KIOSK client
```

## Fluxo de Dados e Comunicação

```
graph LR
    subgraph "Fluxo de Autenticação"
        USER[Usuário]
        AUTH[Autenticação]
        LDAP_AUTH[LDAP/AD]
        TOKEN[Token JWT]

        USER --> AUTH
        AUTH --> LDAP_AUTH
        LDAP_AUTH --> TOKEN
        TOKEN --> USER
    end

    subgraph "Fluxo de Operação"
        REQUEST[Requisição]
        VALIDATE[Validação]
        BUSINESS[Lógica de Negócio]
        DATABASE[Banco de Dados]
        RESPONSE[Resposta]
    end
```

```

    REQUEST --> VALIDATE
    VALIDATE --> BUSINESS
    BUSINESS --> DATABASE
    DATABASE --> RESPONSE
end

subgraph "Fluxo de Notificação"
    TRIGGER[Trigger DB]
    LISTEN[LISTEN/NOTIFY]
    HANDLER[Handler Service]
    NOTIFICATION[Envio Notificação]

    TRIGGER --> LISTEN
    LISTEN --> HANDLER
    HANDLER --> NOTIFICATION
end

subgraph "Fluxo de Backup"
    SCHEDULE[Agendamento]
    BACKUP_EXEC[Execução Backup]
    COMPRESS[Compressão]
    STORAGE_BACKUP[Armazenamento]

    SCHEDULE --> BACKUP_EXEC
    BACKUP_EXEC --> COMPRESS
    COMPRESS --> STORAGE_BACKUP
end

%% Estilos
classDef auth fill:#e1f5fe,stroke:#01579b,stroke-width:2px
classDef operation fill:#e8f5e8,stroke:#2e7d32,stroke-width:
2px
classDef notification fill:#fff3e0,stroke:#f57c00,stroke-
width:2px
classDef backup fill:#fce4ec,stroke:#c2185b,stroke-width:2px

class USER,AUTH,LDAP_AUTH,TOKEN auth
class REQUEST,VALIDATE,BUSINESS,DATABASE,RESPONSE operation
class TRIGGER,LISTEN,HANDLER,NOTIFICATION notification
class SCHEDULE,BACKUP_EXEC,COMPRESS,STORAGE_BACKUP backup

```

## Características Técnicas da Implantação

**Alta Disponibilidade** - Load balancer com failover automático - Cluster de servidores web e API - Replicação de banco de dados master-slave - Armazenamento redundante

**Segurança** - Firewall com WAF (Web Application Firewall) - SSL/TLS em todas as comunicações - Autenticação centralizada via LDAP/AD - Gerenciamento seguro de secrets com Vault

**Monitoramento e Observabilidade** - Métricas de performance com Prometheus/ Grafana - Centralização de logs com ELK Stack - Alertas automáticos para eventos críticos - Dashboards em tempo real

**Backup e Recuperação** - Backup incremental automatizado - Retenção de dados por 7 anos - Testes regulares de recuperação - Armazenamento offsite

**Integração** - APIs REST para integração com sistemas externos - Conectores para ERP e SIAFI - Suporte a dispositivos móveis - Interface adaptável para diferentes tipos de terminal






**Performance e Escalabilidade** - Particionamento de dados por data - Índices otimizados para consultas frequentes - Cache distribuído com Redis - Processamento assíncrono com filas de mensagem

## 6. Diagramas de Casos de Uso

### 6.1. Diagrama de Caso de Uso (Use Case)

Os diagramas de casos de uso apresentam as funcionalidades do sistema SIGA do ponto de vista dos usuários, mostrando os atores envolvidos e suas interações com o sistema.

#### Casos de Uso Principais - Visão Geral

```
graph TD
  %% Atores
  MILITAR[ Militar]
  RESPONSAVEL[ Responsável Almoxxarifado]
  COMANDANTE[ Comandante]
  ADMINISTRADOR[ Administrador Sistema]
  FORNECEDOR[ Fornecedor]

  %% Sistema
  subgraph "Sistema SIGA"
    %% Casos de Uso de Configuração
    UC01[UC01: Gerenciar Unidades]
    UC02[UC02: Cadastrar Militares]
    UC03[UC03: Configurar Almoxxarifados]
    UC04[UC04: Definir Localizações]

    %% Casos de Uso de Materiais
    UC05[UC05: Cadastrar Materiais]
    UC06[UC06: Gerenciar Categorias]
    UC07[UC07: Consultar Catálogo]

    %% Casos de Uso de Operações
```

UC08[UC08: Registrar Entrada]  
UC09[UC09: Processar Saída]  
UC10[UC10: Executar Transferência]  
UC11[UC11: Controlar Cautela]  
UC12[UC12: Processar Devolução]

#### %% Casos de Uso de Consultas

UC13[UC13: Consultar Estoque]  
UC14[UC14: Visualizar Histórico]  
UC15[UC15: Gerar Relatórios]

#### %% Casos de Uso de Notificações

UC16[UC16: Receber Notificações]  
UC17[UC17: Configurar Alertas]

#### %% Casos de Uso de Administração

UC18[UC18: Gerenciar Usuários]  
UC19[UC19: Configurar Sistema]  
UC20[UC20: Auditar Operações]

**end**

#### %% Relacionamentos

MILITAR --> UC07  
MILITAR --> UC09  
MILITAR --> UC11  
MILITAR --> UC12  
MILITAR --> UC13  
MILITAR --> UC14  
MILITAR --> UC16

RESPONSAVEL --> UC03  
RESPONSAVEL --> UC04  
RESPONSAVEL --> UC05  
RESPONSAVEL --> UC08  
RESPONSAVEL --> UC09  
RESPONSAVEL --> UC10  
RESPONSAVEL --> UC11  
RESPONSAVEL --> UC13  
RESPONSAVEL --> UC14  
RESPONSAVEL --> UC15  
RESPONSAVEL --> UC16  
RESPONSAVEL --> UC17

COMANDANTE --> UC01  
COMANDANTE --> UC02  
COMANDANTE --> UC15  
COMANDANTE --> UC16  
COMANDANTE --> UC20

ADMINISTRADOR --> UC01  
ADMINISTRADOR --> UC02  
ADMINISTRADOR --> UC06

```
ADMINISTRADOR --> UC18
ADMINISTRADOR --> UC19
ADMINISTRADOR --> UC20
```

```
FORNECEDOR --> UC08
```

```
%% Estilos
```

```
classDef actor fill:#ffeb3b,stroke:#f57f17,stroke-width:2px
```

```
classDef usecase fill:#4caf50,stroke:#1b5e20,stroke-width:
2px
```

```
classDef system fill:#e3f2fd,stroke:#1565c0,stroke-width:3px
```

```
class
```

```
MILITAR,RESPONSAVEL,COMANDANTE,ADMINISTRADOR,FORNECEDOR actor
```

```
class
```


```
UC01,UC02,UC03,UC04,UC05,UC06,UC07,UC08,UC09,UC10,UC11,UC12,UC13,UC14,UC15
usecase
```

## Casos de Uso Detalhados - Módulo de Operações

```
graph TD
```

```
%% Atores
```

```
MILITAR[ Militar]
```

```
RESPONSAVEL[ Responsável Almoхарifado]
```

```
COMANDANTE[ Comandante]
```

```
FORNECEDOR[ Fornecedor]
```

```
subgraph "Módulo de Operações"
```

```
%% Caso de Uso Principal
```

```
UC_OPERACA0[Gerenciar Operações<br/>de Movimentação]
```

```
%% Casos de Uso Específicos
```

```
UC_ENTRADA[Registrar Entrada<br/>de Material]
```

```
UC_SAIDA[Processar Saída<br/>de Material]
```

```
UC_TRANSFERENCIA[Executar Transferência<br/>entre
Almoхарifados]
```

```
UC_CAUTELA[Controlar Cautela<br/>de Material]
```

```
UC_DEVOLUCA0[Processar Devolução<br/>de Material]
```

```
UC_BAIXA[Registrar Baixa<br/>de Material]
```

```
%% Casos de Uso de Apoio
```

```
UC_VALIDAR[Validar Operação]
```

```
UC_AUTORIZAR[Autorizar Operação]
```

```
UC_DOCUMENTAR[Gerar Documentação]
```

```
UC_NOTIFICAR[Enviar Notificações]
```

```
UC_ATUALIZAR[Atualizar Estoque]
```

```
%% Relacionamentos Include
```

```
UC_ENTRADA -->|include| UC_VALIDAR
```

```
UC_ENTRADA -->|include| UC_DOCUMENTAR
```

```

UC_ENTRADA -->|include| UC_ATUALIZAR

UC_SAIDA -->|include| UC_VALIDAR
UC_SAIDA -->|include| UC_AUTORIZAR
UC_SAIDA -->|include| UC_DOCUMENTAR
UC_SAIDA -->|include| UC_NOTIFICAR
UC_SAIDA -->|include| UC_ATUALIZAR

UC_CAUTELA -->|include| UC_VALIDAR
UC_CAUTELA -->|include| UC_AUTORIZAR
UC_CAUTELA -->|include| UC_DOCUMENTAR
UC_CAUTELA -->|include| UC_NOTIFICAR
UC_CAUTELA -->|include| UC_ATUALIZAR

UC_TRANSFERENCIA -->|include| UC_VALIDAR
UC_TRANSFERENCIA -->|include| UC_AUTORIZAR
UC_TRANSFERENCIA -->|include| UC_DOCUMENTAR
UC_TRANSFERENCIA -->|include| UC_ATUALIZAR

UC_DEVOLUCAO -->|include| UC_VALIDAR
UC_DEVOLUCAO -->|include| UC_DOCUMENTAR
UC_DEVOLUCAO -->|include| UC_ATUALIZAR

UC_BAIXA -->|include| UC_VALIDAR
UC_BAIXA -->|include| UC_AUTORIZAR
UC_BAIXA -->|include| UC_DOCUMENTAR
UC_BAIXA -->|include| UC_ATUALIZAR

%% Relacionamentos Extend
UC_OPERACAO <-->|extend| UC_ENTRADA
UC_OPERACAO <-->|extend| UC_SAIDA
UC_OPERACAO <-->|extend| UC_TRANSFERENCIA
UC_OPERACAO <-->|extend| UC_CAUTELA
UC_OPERACAO <-->|extend| UC_DEVOLUCAO
UC_OPERACAO <-->|extend| UC_BAIXA
end

%% Relacionamentos com Atores
FORNECEDOR --> UC_ENTRADA
RESPONSAVEL --> UC_ENTRADA
RESPONSAVEL --> UC_SAIDA
RESPONSAVEL --> UC_TRANSFERENCIA
RESPONSAVEL --> UC_CAUTELA
RESPONSAVEL --> UC_DEVOLUCAO
RESPONSAVEL --> UC_BAIXA
MILITAR --> UC_SAIDA
MILITAR --> UC_CAUTELA
MILITAR --> UC_DEVOLUCAO
COMANDANTE --> UC_AUTORIZAR

%% Estilos
classDef actor fill:#ffeb3b,stroke:#f57f17,stroke-width:2px

```



```

classDef mainuc fill:#f44336,stroke:#b71c1c,stroke-width:3px
classDef usecase fill:#4caf50,stroke:#1b5e20,stroke-width:
2px
classDef support fill:#2196f3,stroke:#0d47a1,stroke-width:
2px

class MILITAR,RESPONSAVEL,COMANDANTE,FORNECEDOR actor
class UC_OPERACAO mainuc
class
UC_ENTRADA,UC_SAIDA,UC_TRANSFERENCIA,UC_CAUTELA,UC_DEVOLUCAO,UC_BAIXA
usecase
class
UC_VALIDAR,UC_AUTORIZAR,UC_DOCUMENTAR,UC_NOTIFICAR,UC_ATUALIZAR
support

```

## Casos de Uso - Módulo de Consultas e Relatórios

```

graph TD
  %% Atores
  MILITAR[👤 Militar]
  RESPONSAVEL[👤 Responsável Almoxxarifado]
  COMANDANTE[👤 Comandante]
  ADMINISTRADOR[👤 Administrador]

  subgraph "Módulo de Consultas e Relatórios"
    %% Casos de Uso de Consulta
    UC_CONSULTA_ESTOQUE[Consultar Estoque<br/>Atual]
    UC_CONSULTA_MATERIAL[Consultar Material<br/>Específico]
    UC_CONSULTA_HISTORICO[Consultar Histórico<br/>de
Movimentações]
    UC_CONSULTA_OPERACAO[Consultar Operação<br/>Específica]

    %% Casos de Uso de Relatório
    UC_REL_ESTOQUE[Gerar Relatório<br/>de Estoque]
    UC_REL_MOVIMENTACAO[Gerar Relatório<br/>de
Movimentações]
    UC_REL_CAUTELA[Gerar Relatório<br/>de Cautelas]
    UC_REL_VENCIMENTO[Gerar Relatório<br/>de Vencimentos]
    UC_REL_CRITICO[Gerar Relatório<br/>de Estoque Crítico]
    UC_REL_AUDITORIA[Gerar Relatório<br/>de Auditoria]

    %% Casos de Uso de Apoio
    UC_FILTRAR[Aplicar Filtros<br/>de Consulta]
    UC_EXPORTAR[Exportar Dados<br/>(PDF/Excel)]
    UC_AGENDAR[Agendar Relatório<br/>Automático]

    %% Relacionamentos Include
    UC_CONSULTA_ESTOQUE -.->|include| UC_FILTRAR
    UC_CONSULTA_MATERIAL -.->|include| UC_FILTRAR
    UC_CONSULTA_HISTORICO -.->|include| UC_FILTRAR
  end

```

```
UC_CONSULTA_OPERACAO -->|include| UC_FILTRAR
```

```
UC_REL_ESTOQUE -->|include| UC_FILTRAR
```

```
UC_REL_ESTOQUE -->|include| UC_EXPORTAR
```

```
UC_REL_MOVIMENTACAO -->|include| UC_FILTRAR
```

```
UC_REL_MOVIMENTACAO -->|include| UC_EXPORTAR
```

```
UC_REL_CAUTELA -->|include| UC_FILTRAR
```

```
UC_REL_CAUTELA -->|include| UC_EXPORTAR
```

```
UC_REL_VENCIMENTO -->|include| UC_EXPORTAR
```

```
UC_REL_CRITICO -->|include| UC_EXPORTAR
```

```
UC_REL_AUDITORIA -->|include| UC_FILTRAR
```

```
UC_REL_AUDITORIA -->|include| UC_EXPORTAR
```

```
%% Relacionamentos Extend
```

```
UC_REL_ESTOQUE <-->|extend| UC_AGENDAR
```

```
UC_REL_MOVIMENTACAO <-->|extend| UC_AGENDAR
```

```
UC_REL_VENCIMENTO <-->|extend| UC_AGENDAR
```

```
UC_REL_CRITICO <-->|extend| UC_AGENDAR
```

```
end
```

```
%% Relacionamentos com Atores
```

```
MILITAR --> UC_CONSULTA_ESTOQUE
```

```
MILITAR --> UC_CONSULTA_MATERIAL
```

```
MILITAR --> UC_CONSULTA_HISTORICO
```

```
MILITAR --> UC_CONSULTA_OPERACAO
```

```
RESPONSAVEL --> UC_CONSULTA_ESTOQUE
```

```
RESPONSAVEL --> UC_CONSULTA_MATERIAL
```

```
RESPONSAVEL --> UC_CONSULTA_HISTORICO
```

```
RESPONSAVEL --> UC_CONSULTA_OPERACAO
```

```
RESPONSAVEL --> UC_REL_ESTOQUE
```

```
RESPONSAVEL --> UC_REL_MOVIMENTACAO
```

```
RESPONSAVEL --> UC_REL_CAUTELA
```

```
RESPONSAVEL --> UC_REL_VENCIMENTO
```

```
RESPONSAVEL --> UC_REL_CRITICO
```

```
COMANDANTE --> UC_REL_ESTOQUE
```

```
COMANDANTE --> UC_REL_MOVIMENTACAO
```

```
COMANDANTE --> UC_REL_CAUTELA
```

```
COMANDANTE --> UC_REL_AUDITORIA
```

```
COMANDANTE --> UC_AGENDAR
```

```
ADMINISTRADOR --> UC_REL_AUDITORIA
```

```
ADMINISTRADOR --> UC_AGENDAR
```

```
%% Estilos
```

```
classDef actor fill:#ffeb3b,stroke:#f57f17,stroke-width:2px
```

```
classDef consulta fill:#4caf50,stroke:#1b5e20,stroke-width:  
2px
```

```
classDef relatorio fill:#ff9800,stroke:#e65100,stroke-width:  
2px
```

```
classDef support fill:#2196f3,stroke:#0d47a1,stroke-width:
```

2px

```
class MILITAR,RESPONSAVEL,COMANDANTE,ADMINISTRADOR actor
class
UC_CONSULTA_ESTOQUE,UC_CONSULTA_MATERIAL,UC_CONSULTA_HISTORICO,UC_CONSULTA
consulta
class
UC_REL_ESTOQUE,UC_REL_MOVIMENTACAO,UC_REL_CAUTELA,UC_REL_VENCIMENTO,UC_REL
relatorio
class UC_FILTRAR,UC_EXPORTAR,UC_AGENDAR support
```

## Especificação de Casos de Uso Críticos

```
graph TD
    subgraph "UC11: Controlar Cautela de Material"
        subgraph "Pré-condições"
            PRE1[Militar autenticado]
            PRE2[Material disponível]
            PRE3[Almoxarifado ativo]
        end

        subgraph "Fluxo Principal"
            F1[1. Militar solicita cautela]
            F2[2. Sistema valida disponibilidade]
            F3[3. Responsável analisa solicitação]
            F4[4. Sistema solicita autorização]
            F5[5. Comandante autoriza]
            F6[6. Sistema gera termo de cautela]
            F7[7. Militar assina termo]
            F8[8. Sistema atualiza estoque]
            F9[9. Sistema programa alerta devolução]
        end

        subgraph "Fluxos Alternativos"
            A1[2a. Material indisponível<br/>- Notificar indisponibilidade]
            A2[5a. Comandante nega<br/>- Notificar negativa]
            A3[7a. Militar não comparece<br/>- Cancelar operação]
        end

        subgraph "Pós-condições"
            POS1[Estoque atualizado]
            POS2[Termo de cautela gerado]
            POS3[Alerta de devolução programado]
            POS4[Histórico registrado]
        end

        F1 --> F2
        F2 --> F3
    end
```

```

F2 -->|Material indisponível| A1
F3 --> F4
F4 --> F5
F5 --> F6
F5 -->|Negado| A2
F6 --> F7
F7 --> F8
F7 -->|Não comparece| A3
F8 --> F9
F9 --> POS1
F9 --> POS2
F9 --> POS3
F9 --> POS4
end

%% Estilos
classDef pre fill:#e8f5e8,stroke:#2e7d32,stroke-width:2px
classDef main fill:#bbdefb,stroke:#1565c0,stroke-width:2px
classDef alt fill:#ffe0b3,stroke:#f57c00,stroke-width:2px
classDef pos fill:#f3e5f5,stroke:#7b1fa2,stroke-width:2px

class PRE1,PRE2,PRE3 pre
class F1,F2,F3,F4,F5,F6,F7,F8,F9 main
class A1,A2,A3 alt
class POS1,POS2,POS3,POS4 pos

```

## Descrição dos Casos de Uso

### Atores do Sistema

Militar - Usuário básico do sistema - Pode consultar estoque e solicitar materiais - Recebe notificações sobre suas operações

Responsável de Almojarifado - Gerencia operações do almoxarifado - Processa entradas, saídas e transferências - Gera relatórios operacionais

Comandante - Autoriza operações críticas - Acessa relatórios gerenciais - Configura políticas do sistema

Administrador do Sistema - Gerencia configurações técnicas - Controla usuários e permissões - Monitora auditoria e performance

Fornecedor - Interage para registro de entradas - Fornece documentação necessária

### Casos de Uso Principais

Gestão de Operações - Controle completo de movimentações - Validação automática de regras de negócio - Geração automática de documentação - Sistema de autorização hierárquica

Consultas e Relatórios - Consultas em tempo real do estoque - Relatórios personalizáveis - Exportação em múltiplos formatos - Agendamento de relatórios automáticos

Sistema de Notificações - Alertas automáticos de estoque crítico - Notificações de vencimento - Lembretes de devolução de cautelas - Comunicação em tempo real

### **Características dos Casos de Uso**

Validação e Segurança - Todos os casos de uso incluem validação de dados - Sistema de autorização baseado em perfis - Auditoria completa de todas as operações - Controle de acesso granular

Integração e Automação - Atualização automática de estoque - Geração automática de documentos - Integração com sistemas externos - Processamento assíncrono de notificações

## **7. Considerações Finais**

### **7.1. Resumo da Arquitetura**

O Sistema Integrado de Gestão de Almoxarifado (SIGA) representa uma solução completa e robusta para o gerenciamento de materiais do Corpo de Bombeiros Militar de Mato Grosso. A arquitetura apresentada nesta documentação demonstra:

**Estrutura Modular** - Organização clara em módulos funcionais independentes - Separação de responsabilidades bem definida - Facilidade de manutenção e evolução - Reutilização de componentes

**Escalabilidade e Performance** - Sistema de particionamento automático por data - Índices otimizados para consultas frequentes - Arquitetura distribuída com alta disponibilidade - Processamento assíncrono para operações críticas

**Segurança e Auditoria** - Controle de acesso baseado em perfis hierárquicos - Auditoria completa de todas as operações - Rastreabilidade total das movimentações - Integração com sistemas de autenticação corporativa

**Automação Inteligente** - Triggers automatizados para manutenção de integridade - Sistema de notificações em tempo real - Geração automática de documentação - Alertas proativos de estoque crítico

## 7.2. Benefícios da Implementação

**Operacionais** - Redução significativa do tempo de processamento de operações - Eliminação de erros manuais através de validação automática - Visibilidade completa do estoque em tempo real - Otimização do uso de recursos e espaço físico

**Gerenciais** - Relatórios gerenciais precisos e atualizados - Indicadores de performance em tempo real - Controle efetivo de custos e desperdícios - Suporte à tomada de decisão baseada em dados

**Técnicos** - Arquitetura moderna e escalável - Facilidade de integração com sistemas externos - Manutenibilidade e extensibilidade - Performance otimizada para grandes volumes

**Conformidade** - Atendimento às normas de controle interno - Rastreabilidade completa para auditoria - Documentação automática de processos - Conformidade com regulamentações governamentais

## 7.3. Recomendações para Implementação

**Fase de Planejamento** - Realizar treinamento completo das equipes - Definir cronograma de migração gradual - Estabelecer procedimentos de backup e recuperação - Configurar ambiente de homologação

**Fase de Implantação** - Implementar módulos de forma incremental - Realizar testes extensivos em ambiente controlado - Executar migração de dados com validação - Monitorar performance durante período inicial

**Fase de Operação** - Estabelecer rotinas de manutenção preventiva - Implementar monitoramento contínuo - Realizar backups regulares e testes de recuperação - Manter documentação atualizada

## 7.4. Evolução Futura

**Melhorias Tecnológicas** - Implementação de inteligência artificial para previsão de demanda - Integração com IoT para monitoramento automático - Desenvolvimento de aplicativo mobile nativo - Implementação de blockchain para auditoria

**Expansão Funcional** - Módulo de manutenção preventiva de equipamentos - Sistema de gestão de contratos e fornecedores - Integração com sistemas de logística externa - Portal de autoatendimento para militares

**Otimizações** - Implementação de cache distribuído - Otimização de consultas com machine learning - Compressão automática de dados históricos - Implementação de data warehouse para analytics

## 7.5. Conclusão

A documentação apresentada fornece uma visão abrangente e detalhada do Sistema SIGA através de diagramas Mermaid organizados por domínios funcionais. Os diagramas cobrem desde a estrutura básica de dados até aspectos avançados de implantação e casos de uso, facilitando:

- **Compreensão:** Visão clara da arquitetura e funcionamento
- **Manutenção:** Facilidade para localizar e modificar componentes
- **Evolução:** Base sólida para futuras expansões
- **Treinamento:** Material didático para capacitação de equipes
- **Auditoria:** Documentação completa para verificações

O sistema SIGA, conforme modelado, atende plenamente às necessidades de gestão de almoxarifado militar, proporcionando eficiência operacional, controle gerencial e conformidade regulatória. A arquitetura proposta garante escalabilidade para crescimento futuro e flexibilidade para adaptações às mudanças organizacionais.

A implementação bem-sucedida deste sistema resultará em significativa melhoria na gestão de recursos materiais do CBM-MT, contribuindo para a eficiência operacional e a qualidade dos serviços prestados à sociedade.

---

**Documento gerado em:** 17 de junho de 2025

**Versão:** 1.0

**Base:** Modelagem PostgreSQL v6.0 do Sistema SIGA

**Ferramenta:** Diagramas Mermaid

---

Este documento foi elaborado com base nos requisitos especificados e na modelagem de dados fornecida, seguindo as melhores práticas de documentação técnica e arquitetura de sistemas.

# Documentação Técnica - Sistema Integrado de Gestão de Almojarifado (SIGA)

**Corpo de Bombeiros Militar de Mato Grosso (CBM MT)**

**Modelagem de Dados PostgreSQL**

**Data:** 18 de junho de 2025

**Autor:** SD Walingson 4ºBBM Sinop MT

## Sumário Executivo

Esta documentação técnica apresenta a modelagem de dados PostgreSQL para o Sistema Integrado de Gestão de Almojarifado (SIGA) do Corpo de Bombeiros Militar de Mato Grosso. O sistema foi projetado para atender às necessidades específicas de uma organização militar especializada em operações de emergência, combate a incêndios e salvamento, incorporando funcionalidades avançadas de rastreabilidade, gestão de custódia em campo, vinculação a eventos operacionais, avaliação de materiais e registro de ocorrências.

A modelagem apresentada representa uma solução robusta e escalável, capaz de suportar as operações complexas do CBM MT, desde o gerenciamento básico de estoque até o controle detalhado de materiais em operações de campo de longa duração. O banco de dados foi estruturado para garantir a integridade dos dados, a eficiência das consultas e a facilidade de manutenção, utilizando as melhores práticas de engenharia de software e administração de banco de dados.

## 1. Arquitetura Geral do Sistema

### 1.1. Visão Arquitetural

O Sistema Integrado de Gestão de Almojarifado foi concebido seguindo uma arquitetura de banco de dados relacional híbrida, que combina a robustez e consistência dos modelos relacionais tradicionais com a flexibilidade dos campos JSONB para extensibilidade futura. Esta abordagem permite que o sistema mantenha a integridade referencial e as garantias ACID (Atomicidade, Consistência, Isolamento e Durabilidade)



enquanto oferece a capacidade de adaptar-se a novos requisitos sem necessidade de alterações estruturais significativas.

A arquitetura é fundamentada em cinco pilares principais: gestão de entidades organizacionais (unidades, militares, almoxarifados), controle de materiais e equipamentos, rastreamento de operações e movimentações, gestão de eventos operacionais e sistema de auditoria e notificações. Cada pilar é implementado através de conjuntos de tabelas especializadas, interconectadas por relacionamentos bem definidos que garantem a consistência dos dados e facilitam a geração de relatórios complexos.

O sistema utiliza particionamento temporal para tabelas de alto volume, como operações, histórico de estoque, logs de auditoria e notificações. Esta estratégia garante que o desempenho das consultas seja mantido mesmo com o crescimento exponencial dos dados ao longo dos anos de operação. As partições são criadas automaticamente através de triggers e funções especializadas, reduzindo a necessidade de intervenção manual e garantindo a continuidade operacional.

## **1.2. Princípios de Design**

O design da modelagem de dados foi guiado por princípios fundamentais que garantem a qualidade, manutenibilidade e evolução do sistema. O primeiro princípio é a normalização controlada, onde as tabelas são normalizadas até a terceira forma normal para eliminar redundâncias, mas com desnormalizações estratégicas em pontos específicos para otimizar consultas frequentes. Esta abordagem equilibra a integridade dos dados com a performance das operações.

O segundo princípio é a extensibilidade através de campos JSONB, que permite a adição de novos atributos sem alterações estruturais no banco de dados. Estes campos são utilizados de forma criteriosa, apenas em situações onde a flexibilidade é mais importante que a estrutura rígida, como em configurações específicas de equipamentos ou detalhes adicionais de operações que podem variar significativamente entre diferentes tipos de atividades.

O terceiro princípio é a rastreabilidade completa, implementada através de um sistema abrangente de auditoria que registra todas as alterações em tabelas críticas. Cada operação é documentada com informações sobre quem realizou a alteração, quando foi feita, quais dados foram modificados e o contexto da operação. Esta funcionalidade é essencial para organizações militares, onde a prestação de contas e a transparência são requisitos fundamentais.

### 1.3. Estratégia de Particionamento

A estratégia de particionamento adotada é baseada em intervalos temporais, adequada para o padrão de acesso aos dados do sistema. As tabelas `operacao`, `historico_estoque`, `log_auditoria` e `notificacao` são particionadas por períodos específicos: operações por trimestre, histórico de estoque por semestre, logs de auditoria por mês e notificações por mês. Esta granularidade foi escolhida com base na análise do volume esperado de dados e nos padrões de consulta típicos.

O particionamento automático é implementado através de funções PL/pgSQL que são executadas por triggers no momento da inserção de novos registros. Quando uma inserção é feita em uma data para a qual não existe partição, o sistema automaticamente cria a nova partição com os índices apropriados. Este mecanismo garante que o sistema continue operando sem interrupções mesmo quando novos períodos são alcançados.

A gestão do ciclo de vida das partições inclui procedimentos para arquivamento e remoção de partições antigas. Partições com mais de cinco anos são automaticamente movidas para um esquema de arquivo, onde permanecem acessíveis para consultas históricas mas não impactam o desempenho das operações correntes. Partições com mais de dez anos podem ser exportadas para armazenamento externo e removidas do banco principal, seguindo políticas de retenção de dados estabelecidas pela organização.

## 2. Estrutura de Entidades Organizacionais

### 2.1. Gestão de Unidades do CBM

A estrutura organizacional do Corpo de Bombeiros Militar de Mato Grosso é representada através da tabela `unidade_cbm`, que modela a hierarquia complexa da organização. Esta tabela suporta uma estrutura hierárquica recursiva, onde cada unidade pode ter uma unidade superior, permitindo a representação de comandos regionais, batalhões, companhias, pelotões e outras subdivisões organizacionais.

A tabela inclui informações essenciais como nome completo da unidade, sigla padronizada, dados de contato e localização física. O campo `comandante_id` estabelece a ligação com o militar responsável pela unidade, permitindo consultas rápidas sobre a cadeia de comando. O campo `unidade_superior_id` implementa a hierarquia organizacional, facilitando consultas que precisam navegar pela estrutura de comando.

O campo `informacoes_complementares` em formato JSONB permite o armazenamento de dados específicos de cada tipo de unidade, como especialidades operacionais, efetivo autorizado, equipamentos principais e outras características que podem variar significativamente entre diferentes tipos de unidades. Esta flexibilidade é crucial para acomodar a diversidade de unidades especializadas dentro do CBM MT.

## 2.2. Cadastro e Gestão de Militares

O sistema de gestão de militares é implementado através da tabela `militar`, que centraliza todas as informações pessoais e funcionais dos membros da corporação. A tabela utiliza domínios personalizados para garantir a consistência de dados críticos como matrícula e CPF, implementando validações automáticas que previnem a inserção de dados malformados.

A estrutura inclui campos para informações básicas como nome completo, nome de guerra (amplamente utilizado no ambiente militar), posto ou graduação, e dados de contato. A ligação com a unidade de lotação é estabelecida através do campo `unidade_id`, permitindo consultas eficientes sobre a distribuição de efetivo e a localização de militares específicos.

O campo `situacao` utiliza um tipo enumerado que define os possíveis estados funcionais de um militar (ativo, licenciado, afastado, reservista, reformado), garantindo consistência nos dados e facilitando consultas estatísticas sobre o efetivo. O campo `informacoes_complementares` em JSONB permite o armazenamento de dados como especialidades, cursos realizados, habilitações especiais e outras informações que podem ser relevantes para a designação de tarefas específicas.

## 2.3. Estrutura de Almoxarifados

A gestão de almoxarifados é implementada através de duas tabelas principais: `almoxarifado` e `localizacao`. A tabela `almoxarifado` define as características gerais de cada depósito, incluindo capacidade, área total, responsável e configurações específicas. Esta estrutura permite que cada unidade do CBM tenha múltiplos almoxarifados especializados, como almoxarifados de materiais de combate a incêndio, equipamentos de resgate, materiais administrativos, entre outros.

A tabela `localizacao` implementa um sistema de endereçamento interno detalhado, permitindo a localização precisa de materiais dentro de cada almoxarifado. O sistema suporta uma estrutura hierárquica de setores, prateleiras, níveis e posições, facilitando tanto o armazenamento organizado quanto a localização rápida de itens específicos. Cada localização pode ter restrições especiais definidas em JSONB, como limitações de peso, tipos de materiais permitidos ou requisitos de acesso restrito.

A integração entre almoxarifados e localizações permite consultas complexas sobre a distribuição espacial de materiais, otimização de rotas de coleta e identificação de localizações subutilizadas ou sobrecarregadas. Esta funcionalidade é essencial para a eficiência operacional, especialmente em situações de emergência onde a rapidez na localização e retirada de materiais pode ser crítica.

## 3. Sistema de Classificação e Gestão de Materiais

### 3.1. Categorização Dinâmica de Materiais

O sistema de categorização de materiais é implementado através da tabela `categoria_material`, que suporta uma hierarquia flexível e extensível de categorias e subcategorias. Esta estrutura permite a organização lógica de materiais de acordo com critérios funcionais, operacionais ou administrativos, facilitando tanto a gestão quanto a localização de itens específicos.

A hierarquia de categorias é implementada através de uma estrutura recursiva, onde cada categoria pode ter uma categoria pai, permitindo múltiplos níveis de classificação. Por exemplo, uma categoria principal "Equipamentos de Combate a Incêndio" pode ter subcategorias como "Mangueiras", "Esguichos", "Equipamentos de Proteção Individual", e cada uma dessas pode ter subdivisões adicionais baseadas em características técnicas ou aplicações específicas.

O campo `configuracoes_especificas` em JSONB permite que cada categoria tenha regras e parâmetros únicos, como requisitos de treinamento para uso, periodicidade de manutenção, condições especiais de armazenamento ou critérios de substituição. Esta flexibilidade é fundamental para acomodar a diversidade de materiais utilizados pelo CBM MT, desde equipamentos altamente especializados até materiais de consumo básico.

### 3.2. Gestão de Tipos de Viaturas

A tabela `tipo_viatura` implementa um sistema de categorização dinâmica para os diferentes tipos de viaturas utilizadas pelo CBM MT. Esta abordagem permite a adição de novos tipos de viaturas conforme a corporação adquire equipamentos especializados ou desenvolve novas capacidades operacionais, sem necessidade de alterações estruturais no banco de dados.

Cada tipo de viatura é caracterizado por informações como categoria principal (combate a incêndio, resgate, transporte, comando), subcategoria específica (urbano, florestal, aquático), e especificações técnicas detalhadas armazenadas em JSONB. Esta estrutura

permite consultas sofisticadas sobre capacidades operacionais, como identificar todas as viaturas com capacidade de transporte de água superior a determinado volume ou viaturas equipadas com escadas mecânicas.

O campo `requisitos_operacionais` define os pré-requisitos para operação de cada tipo de viatura, incluindo habilitações necessárias, treinamentos obrigatórios e tamanho mínimo da tripulação. Esta informação é crucial para o planejamento operacional e para garantir que as viaturas sejam operadas apenas por militares devidamente qualificados.

### 3.3. Estrutura Base de Materiais

A tabela `material_base` serve como o núcleo central do sistema de gestão de materiais, concentrando informações comuns a todos os tipos de materiais independentemente de sua natureza específica. Esta abordagem permite consultas unificadas sobre o inventário completo enquanto mantém a flexibilidade para especializações através de tabelas relacionadas.

Cada material é identificado por um código único que segue padrões estabelecidos pela organização, facilitando a integração com sistemas externos e a padronização de procedimentos. O sistema inclui campos para informações básicas como nome, descrição, categoria, unidade de medida e valores financeiros, além de parâmetros operacionais como estoques mínimo e máximo.

O campo `atributos_adicionais` em JSONB permite o armazenamento de características específicas que podem variar significativamente entre diferentes tipos de materiais. Por exemplo, materiais químicos podem ter informações sobre toxicidade e prazo de validade, equipamentos eletrônicos podem ter especificações técnicas detalhadas, e materiais têxteis podem ter informações sobre tamanhos e cores disponíveis.

### 3.4. Especialização por Tipo de Material

O sistema implementa especialização através de tabelas dedicadas para diferentes tipos de materiais: `material_consumo`, `material_permanente`, `equipamento` e `viatura`. Esta abordagem permite que cada tipo de material tenha campos específicos para suas características únicas, mantendo a organização lógica dos dados e otimizando as consultas.

A tabela `material_consumo` inclui informações específicas para materiais que são consumidos durante o uso, como data de validade, lote de fabricação, instruções de uso e restrições de armazenamento. Estes campos são essenciais para o controle de

qualidade e para garantir que materiais vencidos ou inadequados não sejam utilizados em operações críticas.

A tabela `material_permanente` foi aprimorada para incluir controle de ocorrências, com campos que indicam se o material possui ocorrências pendentes e um histórico de incidentes em formato JSONB. Esta funcionalidade permite o acompanhamento detalhado do ciclo de vida de materiais permanentes, identificando padrões de problemas e orientando decisões sobre manutenção, substituição ou baixa.

A tabela `equipamento` concentra informações técnicas específicas para equipamentos especializados, incluindo número de série, especificações técnicas detalhadas, manuais de operação e certificações. A tabela `viatura` inclui informações específicas para viaturas, como dados de documentação, quilometragem, manutenção e seguro, essenciais para o controle da frota.

## 4. Sistema de Operações e Movimentações

### 4.1. Estrutura Principal de Operações

A tabela `operacao` constitui o núcleo do sistema de rastreamento de movimentações, registrando todas as transações que envolvem materiais dentro do sistema. A tabela é particionada por data para garantir performance adequada mesmo com grandes volumes de dados, e inclui campos para todos os tipos de movimentação suportados pelo sistema.

Cada operação é identificada por um número sequencial único que facilita a referência e o rastreamento. O campo `tipo_movimentacao` utiliza um tipo enumerado que inclui todas as possibilidades: entrada, saída, transferência, devolução, concessão, cautela, baixa, manutenção, reserva e transferência de custódia. Esta padronização garante consistência nos dados e facilita a geração de relatórios estatísticos.

A estrutura inclui campos para identificar os almoxarifados de origem e destino, os militares responsável e recebedor, quantidades, valores e datas relevantes. O campo `detalhes_adicionais` em JSONB permite o armazenamento de informações específicas para cada tipo de operação, como condições especiais, autorizações necessárias ou observações técnicas.

### 4.2. Gestão de Custódia e Transferências em Campo

Uma das inovações mais significativas do sistema é a capacidade de rastrear transferências de custódia de materiais diretamente em campo, implementada através da tabela `custodia_campo`. Esta funcionalidade atende a uma necessidade crítica das

operações do CBM MT, onde materiais cautelados frequentemente precisam ser transferidos entre militares durante operações de longa duração sem retorno ao almoxarifado.

A tabela registra cada evento de transferência com informações detalhadas sobre quem entregou, quem recebeu, quando e onde a transferência ocorreu, e as condições do material no momento da transferência. Esta rastreabilidade é essencial para manter a cadeia de responsabilidade e para investigações posteriores em caso de problemas com o material.

O sistema automaticamente atualiza a tabela `cautela_detalle` para refletir o novo custodiante, mantendo sempre atualizada a informação sobre quem detém a responsabilidade por cada material cautelado. Esta funcionalidade é implementada através de triggers que garantem a consistência dos dados e reduzem a possibilidade de erros manuais.

### 4.3. Controle de Estoque em Tempo Real

O sistema mantém o controle de estoque em tempo real através da tabela `estoque_atual`, que é automaticamente atualizada sempre que uma operação é concluída. A tabela foi aprimorada para incluir o controle de quantidades em custódia externa, permitindo uma visão completa da distribuição de materiais entre o almoxarifado físico e as operações de campo.

A atualização automática é implementada através de triggers que analisam o tipo de movimentação e ajustam as quantidades apropriadas. Por exemplo, uma operação de cautela reduz a quantidade disponível no almoxarifado e aumenta a quantidade em custódia externa, enquanto uma devolução faz o movimento inverso. Esta automação elimina discrepâncias entre o estoque físico e o registrado no sistema.

O sistema também mantém um histórico detalhado de estoque através da tabela `historico_estoque`, que registra snapshots periódicos das quantidades de cada material. Esta funcionalidade permite análises temporais de consumo, identificação de tendências e planejamento de aquisições futuras baseado em dados históricos concretos.

### 4.4. Detalhamento de Operações Específicas

O sistema inclui tabelas de detalhamento para operações específicas que requerem informações adicionais além das registradas na tabela principal de operações. A tabela `cautela_detalle` foi aprimorada para incluir informações sobre custódia atual,

permitindo o rastreamento completo da cadeia de responsabilidade mesmo quando há transferências em campo.

Outras tabelas de detalhamento incluem `entrada_detalhe`, `saida_detalhe`, `transferencia_detalhe`, `devolucao_detalhe`, `concessao_detalhe`, `baixa_detalhe`, `manutencao_detalhe` e `reserva_detalhe`. Cada uma dessas tabelas inclui campos específicos para o tipo de operação, como fornecedores para entradas, motivos para baixas, ou especificações técnicas para manutenções.

Esta estrutura detalhada permite que o sistema capture todas as nuances de cada tipo de operação, fornecendo informações ricas para análises posteriores e garantindo que todos os aspectos relevantes sejam documentados para fins de auditoria e prestação de contas.

## 5. Gestão de Eventos Operacionais

### 5.1. Estrutura de Eventos e Atividades

O sistema introduz o conceito de eventos operacionais através da tabela `evento_operacional`, que permite agrupar operações de movimentação de materiais sob contextos operacionais específicos. Esta funcionalidade é fundamental para organizações como o CBM MT, onde materiais são frequentemente utilizados em operações de longa duração que envolvem múltiplas movimentações e diversos tipos de recursos.

A tabela suporta diferentes tipos de eventos, incluindo incidentes (como incêndios florestais de grande porte), campanhas (como operações preventivas sazonais), atividades de rotina (como treinamentos regulares), e eventos sociais (como demonstrações públicas). Cada tipo de evento pode ter características específicas em termos de duração, recursos envolvidos e procedimentos operacionais.

O sistema permite que eventos tenham hierarquia temporal, onde um incidente de longa duração pode ser dividido em ciclos operacionais menores, cada um com suas próprias características e recursos. Esta flexibilidade é essencial para operações como combate a incêndios florestais, que podem durar meses e envolver rotação de equipes e equipamentos.

### 5.2. Vinculação de Operações a Eventos

A tabela `operacao_evento` implementa a ligação entre operações de movimentação de materiais e eventos operacionais, permitindo que múltiplas operações sejam associadas a um mesmo evento. Esta funcionalidade possibilita análises consolidadas



de consumo de recursos por evento, facilitando a avaliação de custos operacionais e o planejamento de recursos para eventos similares no futuro.

A vinculação pode ser feita no momento da criação da operação ou posteriormente, permitindo flexibilidade nos procedimentos operacionais. O sistema suporta a associação de uma operação a múltiplos eventos, acomodando situações onde recursos são compartilhados entre diferentes atividades simultâneas.

Esta estrutura permite a geração de relatórios detalhados sobre o consumo de materiais por evento, incluindo análises de custo-benefício, identificação de recursos críticos e avaliação da eficiência operacional. Estas informações são valiosas para o planejamento estratégico e para a justificativa de investimentos em novos equipamentos ou materiais.

### **5.3. Análise de Consumo por Evento**

O sistema inclui views especializadas para análise de consumo de materiais por evento, permitindo consultas rápidas sobre quais materiais foram utilizados, em que quantidades, e qual foi o custo total associado. A view `vw_consumo_material_por_evento` consolida estas informações de forma eficiente, facilitando a geração de relatórios gerenciais.

As análises podem ser segmentadas por tipo de evento, período temporal, unidade responsável ou categoria de material, proporcionando flexibilidade para diferentes necessidades de relatório. O sistema também permite comparações entre eventos similares, identificando variações no consumo que podem indicar oportunidades de otimização ou necessidades de treinamento adicional.

Esta funcionalidade é particularmente valiosa para o planejamento de operações futuras, permitindo estimativas mais precisas de recursos necessários baseadas em dados históricos de eventos similares. Também facilita a prestação de contas para órgãos superiores e a justificativa de custos operacionais.

## **6. Sistema de Avaliação e Feedback**

### **6.1. Avaliação de Materiais pelos Usuários**

O sistema implementa um mecanismo abrangente de avaliação de materiais através da tabela `avaliacao_material`, permitindo que os militares forneçam feedback detalhado sobre a qualidade, usabilidade e desempenho dos materiais após o uso. Esta funcionalidade é crucial para a melhoria contínua do inventário e para decisões informadas sobre aquisições futuras.

Cada avaliação inclui uma nota geral de 1 a 5 estrelas, complementada por campos de texto livre para comentários detalhados, pontos positivos, pontos negativos e sugestões de melhoria. O sistema também registra as condições de uso do material, permitindo correlacionar o desempenho com fatores ambientais ou operacionais específicos.

As avaliações podem ser vinculadas a operações específicas ou eventos operacionais, proporcionando contexto adicional para a análise do feedback. Esta vinculação permite identificar se problemas com materiais são sistemáticos ou relacionados a condições específicas de uso, orientando decisões sobre treinamento, procedimentos operacionais ou especificações técnicas.

## **6.2. Análise de Desempenho e Ranking**

O sistema calcula automaticamente médias de avaliação para cada material, permitindo a criação de rankings de desempenho que orientam decisões de aquisição e substituição. Materiais com avaliações consistentemente baixas podem ser identificados para revisão de especificações ou substituição por alternativas superiores.

A análise de feedback também permite identificar padrões de problemas que podem indicar necessidades de treinamento adicional, melhorias em procedimentos operacionais ou ajustes nas especificações técnicas de materiais. Por exemplo, se múltiplos usuários reportam dificuldades com um equipamento específico, isso pode indicar a necessidade de treinamento mais detalhado ou a inadequação do equipamento para as condições de uso locais.

O sistema inclui funcionalidades para análise temporal das avaliações, permitindo identificar tendências de melhoria ou deterioração no desempenho de materiais ao longo do tempo. Esta informação é valiosa para decisões sobre ciclos de substituição e para avaliação da qualidade de fornecedores.

## **6.3. Integração com Decisões de Aquisição**

As informações de avaliação são integradas às views principais do sistema, permitindo que gestores tenham acesso imediato ao feedback dos usuários ao consultar informações sobre materiais. Esta integração facilita decisões informadas sobre reposição de estoque, especificação de novos materiais e avaliação de fornecedores.

O sistema permite a geração de relatórios consolidados de avaliação por categoria de material, fornecedor ou período temporal, facilitando análises estratégicas sobre a qualidade do inventário. Estes relatórios podem ser utilizados em negociações com fornecedores, especificação de novos contratos e planejamento de investimentos em equipamentos.

A funcionalidade de avaliação também suporta a criação de especificações técnicas mais precisas para futuras aquisições, baseadas no feedback real dos usuários finais. Esta abordagem centrada no usuário garante que os materiais adquiridos atendam efetivamente às necessidades operacionais da organização.

## **7. Sistema de Registro de Ocorrências**

### **7.1. Documentação de Incidentes com Materiais**

A tabela `ocorrencia_material` implementa um sistema abrangente para registro e acompanhamento de incidentes envolvendo materiais, desde avarias leves até perdas totais. Esta funcionalidade é essencial para organizações militares, onde a prestação de contas sobre o patrimônio público é um requisito fundamental e onde a análise de incidentes pode revelar oportunidades de melhoria em procedimentos ou treinamento.

Cada ocorrência é classificada por tipo (avaria leve, moderada ou grave, perda total, extravio, defeito de fabricação, desgaste natural), permitindo análises estatísticas sobre padrões de problemas. O sistema registra informações detalhadas sobre as circunstâncias da ocorrência, incluindo local, condições operacionais, causa provável e impacto no material.

O registro inclui campos para documentar as providências imediatas tomadas após a ocorrência e o status do processo de apuração. Esta estrutura garante que todos os incidentes sejam adequadamente investigados e que as responsabilidades sejam claramente estabelecidas, protegendo tanto a organização quanto os militares envolvidos.

### **7.2. Processo de Apuração e Resolução**

O sistema implementa um fluxo completo de apuração de ocorrências, com campos para designação de responsável pela investigação, datas de início e fim da apuração, e parecer final. Este processo estruturado garante que todas as ocorrências sejam adequadamente investigadas e que as conclusões sejam documentadas para referência futura.

O status da ocorrência evolui através de estados bem definidos: registrada, em análise, aguardando reparo, reparado, aguardando substituição, substituído, baixado após ocorrência, resolvida sem ação, ou arquivada. Esta progressão permite o acompanhamento do processo de resolução e garante que nenhuma ocorrência seja esquecida ou deixada sem resolução.

O sistema automaticamente atualiza o status dos materiais permanentes quando ocorrências são registradas, marcando-os como tendo ocorrências pendentes e atualizando o histórico de incidentes. Esta funcionalidade permite consultas rápidas sobre materiais com problemas e facilita decisões sobre manutenção, substituição ou baixa.

### **7.3. Análise de Padrões e Prevenção**

O sistema permite análises estatísticas sobre ocorrências, identificando padrões que podem indicar problemas sistemáticos com materiais específicos, fornecedores, ou procedimentos operacionais. Por exemplo, se um tipo específico de equipamento apresenta alta incidência de avarias em determinadas condições, isso pode indicar inadequação do equipamento para essas condições ou necessidade de treinamento adicional.

As análises podem ser segmentadas por tipo de material, unidade usuária, tipo de operação, ou condições ambientais, proporcionando insights valiosos para a melhoria contínua dos processos. O sistema também permite correlacionar ocorrências com avaliações de materiais, identificando situações onde problemas técnicos impactam a satisfação dos usuários.

Esta funcionalidade de análise é fundamental para a gestão proativa de riscos, permitindo a identificação e correção de problemas antes que se tornem críticos. Também fornece dados objetivos para negociações com fornecedores e para especificação de requisitos técnicos em futuras aquisições.

## **8. Sistema de Auditoria e Rastreabilidade**

### **8.1. Log Abrangente de Auditoria**

O sistema implementa um mecanismo abrangente de auditoria através da tabela `log_auditoria`, que registra automaticamente todas as operações críticas realizadas no sistema. Esta funcionalidade é implementada através de triggers que capturam inserções, atualizações e exclusões em tabelas sensíveis, garantindo que todas as alterações sejam documentadas com informações sobre quem, quando, o que e por que foi alterado.

Cada entrada no log inclui informações sobre a tabela afetada, o tipo de operação, o ID do registro modificado, o usuário responsável, os dados antes e depois da alteração, e informações de contexto como endereço IP e user agent. Esta riqueza de informações

permite investigações detalhadas sobre alterações específicas e facilita a identificação de padrões de uso ou problemas operacionais.

O log de auditoria é particionado por mês para garantir performance adequada mesmo com grandes volumes de dados. Partições antigas são automaticamente arquivadas seguindo políticas de retenção estabelecidas, mantendo o histórico acessível para investigações enquanto otimiza o desempenho das operações correntes.

## **8.2. Rastreabilidade Completa de Materiais**

O sistema oferece rastreabilidade completa de materiais desde a entrada no sistema até sua baixa ou devolução. Esta rastreabilidade é implementada através da combinação de múltiplas tabelas: operações principais, detalhes específicos por tipo de operação, transferências de custódia em campo, e registros de ocorrências.

A view `vw_rastreabilidade_custodia_campo` consolida informações sobre a cadeia de custódia de materiais cautelados, mostrando todas as transferências realizadas em campo e identificando o custodiante atual. Esta funcionalidade é crucial para operações de longa duração onde materiais podem passar por múltiplas transferências antes de retornar ao almoxarifado.

O sistema também mantém histórico detalhado de todas as movimentações de cada material, permitindo consultas sobre onde um material específico esteve, quem foi responsável por ele em cada momento, e quais operações ou eventos estiveram envolvidos. Esta informação é valiosa tanto para investigações específicas quanto para análises de padrões de uso.

## **8.3. Integridade e Consistência de Dados**

O sistema implementa múltiplas camadas de validação para garantir a integridade e consistência dos dados. Domínios personalizados validam formatos de dados críticos como CPF e placas de viaturas. Tipos enumerados garantem que apenas valores válidos sejam inseridos em campos de status e classificação. Constraints de chave estrangeira mantêm a integridade referencial entre tabelas relacionadas.

Triggers especializados implementam regras de negócio complexas, como a atualização automática de estoques após operações e a verificação de disponibilidade antes de permitir saídas. Estas validações automáticas reduzem significativamente a possibilidade de erros humanos e garantem que o sistema mantenha sempre um estado consistente.

O sistema também inclui procedimentos para verificação periódica da integridade dos dados, identificando e reportando inconsistências que possam ter surgido devido a

falhas de sistema ou operações manuais diretas no banco de dados. Estas verificações são executadas automaticamente e geram alertas para os administradores quando problemas são detectados.

## **9. Sistema de Notificações e Alertas**

### **9.1. Notificações em Tempo Real**

O sistema implementa um mecanismo robusto de notificações em tempo real utilizando a funcionalidade LISTEN/NOTIFY do PostgreSQL. Esta abordagem permite que o sistema gere alertas automáticos para situações críticas como estoque baixo, documentação vencida, ou ocorrências graves com materiais, sem necessidade de polling constante do banco de dados.

As notificações são categorizadas por tipo e prioridade, permitindo que diferentes tipos de alertas sejam tratados de forma apropriada. Notificações críticas, como materiais com estoque zerado durante uma operação de emergência, podem ser enviadas imediatamente para múltiplos canais (sistema, email, SMS), enquanto notificações de baixa prioridade podem ser agrupadas em relatórios periódicos.

O sistema suporta múltiplos canais de notificação e permite que usuários configurem suas preferências de recebimento. Administradores podem receber todas as notificações críticas, enquanto usuários operacionais podem receber apenas notificações relacionadas aos materiais sob sua responsabilidade.

### **9.2. Alertas Automáticos de Estoque**

O sistema monitora continuamente os níveis de estoque e gera alertas automáticos quando materiais atingem níveis críticos. Estes alertas são configuráveis por material, permitindo que itens críticos tenham limites de alerta mais conservadores que materiais de uso menos frequente.

Os alertas incluem informações contextuais como histórico de consumo, tempo médio de reposição, e operações programadas que podem impactar o estoque. Esta informação adicional permite que gestores tomem decisões informadas sobre a urgência de reposição e sobre quantidades a serem adquiridas.

O sistema também gera alertas preventivos baseados em análises de tendência, identificando materiais que, embora ainda não tenham atingido níveis críticos, estão em trajetória de esgotamento baseada nos padrões de consumo recentes. Esta funcionalidade permite ações proativas que evitam situações de desabastecimento.

### **9.3. Monitoramento de Documentação e Compliance**

O sistema monitora automaticamente datas de vencimento de documentação de viaturas, certificações de equipamentos, e validade de materiais de consumo. Alertas são gerados com antecedência suficiente para permitir renovação ou substituição antes do vencimento, evitando situações onde recursos ficam indisponíveis por problemas documentais.

Para viaturas, o sistema monitora vencimentos de seguro, licenciamento, revisões obrigatórias e inspeções técnicas. Para equipamentos, monitora certificações técnicas, calibrações obrigatórias e manutenções preventivas programadas. Para materiais de consumo, monitora datas de validade e condições de armazenamento.

Estes alertas são essenciais para manter a conformidade com regulamentações e para garantir que todos os recursos estejam sempre disponíveis para uso operacional. O sistema permite configuração de múltiplos alertas para cada tipo de vencimento, com antecedências diferentes para permitir planejamento adequado.

## **10. Performance e Otimização**

### **10.1. Estratégias de Indexação**

O sistema implementa uma estratégia abrangente de indexação que equilibra performance de consultas com overhead de manutenção. Índices B-tree são utilizados para consultas de igualdade e range em campos frequentemente utilizados em filtros e ordenações. Índices compostos são criados para consultas que filtram por múltiplos campos simultaneamente.

Índices GIN especializados são utilizados para campos JSONB, permitindo consultas eficientes em atributos específicos dentro dos documentos JSON. Estes índices são particularmente importantes para consultas em configurações específicas de equipamentos, detalhes adicionais de operações, e outros dados semi-estruturados.

A estratégia de indexação é continuamente monitorada e ajustada baseada em padrões reais de uso. O sistema inclui queries para identificar consultas lentas e índices subutilizados, permitindo otimizações contínuas da performance.

### **10.2. Otimização de Consultas**

O sistema inclui views otimizadas para consultas frequentes, pré-calculando joins complexos e agregações que seriam custosas se executadas repetidamente. Estas views

são especialmente importantes para relatórios gerenciais e dashboards que precisam de resposta rápida.

Consultas complexas que envolvem múltiplas tabelas particionadas são otimizadas através de técnicas como partition pruning, onde o otimizador do PostgreSQL automaticamente elimina partições que não contêm dados relevantes para a consulta. Esta otimização é particularmente efetiva para consultas que filtram por períodos temporais específicos.

O sistema também utiliza materialized views para consultas analíticas complexas que não precisam de dados em tempo real. Estas views são atualizadas periodicamente através de jobs agendados, proporcionando performance excelente para relatórios estatísticos e análises históricas.

### **10.3. Gestão de Recursos e Escalabilidade**

O sistema é projetado para escalar horizontalmente através de técnicas como read replicas para distribuir carga de consultas e particionamento para distribuir dados. A arquitetura suporta a adição de novos servidores de leitura conforme a demanda cresce, sem impacto nas operações de escrita.

A gestão de memória é otimizada através de configurações específicas do PostgreSQL que consideram os padrões de acesso aos dados do sistema. Buffers são dimensionados para manter em memória as partições mais acessadas e os índices mais utilizados, minimizando acessos a disco.

O sistema inclui monitoramento automático de performance com alertas para situações que podem impactar a experiência do usuário. Métricas como tempo de resposta de consultas, utilização de CPU e memória, e taxa de cache hits são continuamente monitoradas e reportadas.

## **Conclusão**

A modelagem de dados apresentada representa uma solução abrangente e robusta para as necessidades de gestão de almoxarifado do Corpo de Bombeiros Militar de Mato Grosso. O sistema combina a solidez de uma arquitetura relacional bem estruturada com a flexibilidade necessária para acomodar as complexidades e variações inerentes às operações de uma organização militar especializada em emergências.

As funcionalidades implementadas atendem não apenas às necessidades básicas de controle de estoque, mas também aos requisitos avançados de rastreabilidade, prestação de contas, análise de performance e gestão de operações complexas. A



capacidade de rastrear materiais em campo, vincular movimentações a eventos operacionais, coletar feedback dos usuários e registrar ocorrências proporciona uma visão completa e detalhada de todos os aspectos relacionados à gestão de materiais.

A arquitetura técnica do sistema, com seu uso criterioso de particionamento, indexação especializada, e automação através de triggers e procedures, garante que o sistema mantenha performance adequada mesmo com o crescimento dos dados ao longo dos anos. A implementação de auditoria abrangente e notificações em tempo real proporciona a transparência e responsividade necessárias para uma organização pública.

O sistema está preparado para evoluir junto com as necessidades da organização, através de sua arquitetura extensível e do uso estratégico de campos JSONB para acomodar novos requisitos. Esta flexibilidade, combinada com a solidez da estrutura base, garante que o investimento no sistema continue proporcionando valor ao longo de muitos anos de operação.

---

**Autor:** Manus AI

**Data:** 18 de junho de 2025

**Sistema:** SIGA - Sistema Integrado de Gestão de Almoxarifado

**Organização:** Corpo de Bombeiros Militar de Mato Grosso

# Plano de Manutenção do Banco de Dados

## - Sistema SIGA

**Corpo de Bombeiros Militar de Mato Grosso (CBM MT)**

**Sistema Integrado de Gestão de Almoxarifado**

**Data:** 18 de junho de 2025

**Autor:** SD Walingson 4ºBBM MT

## Sumário Executivo

Este documento apresenta o plano abrangente de manutenção para o banco de dados PostgreSQL do Sistema Integrado de Gestão de Almoxarifado (SIGA) do Corpo de Bombeiros Militar de Mato Grosso. O plano foi atualizado para incorporar as novas funcionalidades implementadas, incluindo gestão de custódia em campo, vinculação a eventos operacionais, sistema de avaliação de materiais e registro de ocorrências.

O plano de manutenção é estruturado em rotinas diárias, semanais, mensais, trimestrais e anuais, cada uma com objetivos específicos para garantir a performance, integridade, segurança e disponibilidade do sistema. As rotinas foram projetadas considerando as características específicas do ambiente operacional do CBM MT, incluindo picos de demanda durante emergências e a necessidade de disponibilidade contínua para operações críticas.

A implementação deste plano garante que o sistema mantenha performance otimizada mesmo com o crescimento dos dados, que a integridade dos dados seja preservada através de verificações regulares, que backups sejam realizados de forma consistente e confiável, e que potenciais problemas sejam identificados e resolvidos antes de impactar as operações.

## 1. Rotinas de Manutenção Diária

### 1.1. Verificação de Integridade e Consistência

As rotinas diárias de verificação de integridade constituem a primeira linha de defesa contra problemas que podem comprometer a confiabilidade do sistema. Estas verificações são executadas automaticamente durante períodos de baixa atividade, tipicamente entre 02:00 e 04:00, quando o impacto nas operações é mínimo.

A verificação de integridade referencial é executada diariamente para garantir que todas as relações entre tabelas estejam consistentes. Esta verificação é particularmente importante para as novas funcionalidades implementadas, como as ligações entre operações de cautela e transferências de custódia em campo, ou entre operações e eventos operacionais. O sistema verifica se todos os registros de `custodia_campo` referenciam operações de cautela válidas e se todas as associações em `operacao_evento` apontam para eventos existentes.

A verificação de consistência de estoque é executada comparando as quantidades registradas na tabela `estoque_atual` com os somatórios calculados a partir das operações registradas. Esta verificação inclui agora as quantidades em custódia externa, garantindo que o total de materiais cautelados corresponda exatamente aos registros de custódia em campo. Discrepâncias são automaticamente reportadas para investigação imediata.

A verificação de integridade de partições garante que todas as partições necessárias existam e estejam corretamente configuradas. O sistema verifica se existem partições para o mês corrente e os próximos dois meses para todas as tabelas particionadas (`operacao`, `historico_estoque`, `log_auditoria`, `notificacao`), criando automaticamente partições ausentes conforme necessário.

## 1.2. Monitoramento de Performance

O monitoramento diário de performance inclui a coleta e análise de métricas críticas que indicam a saúde geral do sistema. As métricas de tempo de resposta são coletadas para consultas representativas de cada módulo do sistema, incluindo consultas de estoque, rastreabilidade de custódia, análise de eventos e geração de relatórios.

A análise de consultas lentas identifica queries que excedem limites de tempo estabelecidos, permitindo otimizações proativas antes que problemas de performance impactem os usuários. O sistema mantém um histórico de performance que permite identificar tendências de degradação e correlacionar problemas de performance com eventos específicos como picos de operações ou crescimento de dados.

O monitoramento de utilização de recursos inclui CPU, memória, espaço em disco e I/O. Alertas são gerados quando qualquer recurso excede 80% de utilização, permitindo ações preventivas antes que limitações de recursos afetem a disponibilidade do sistema. O monitoramento é especialmente importante para o espaço em disco, considerando o crescimento contínuo das tabelas particionadas.

A verificação de locks e bloqueios identifica situações onde transações estão sendo bloqueadas por períodos excessivos, o que pode indicar problemas de concorrência ou

transações mal otimizadas. Esta verificação é particularmente importante durante períodos de alta atividade, como durante grandes operações de emergência quando múltiplos usuários estão registrando movimentações simultaneamente.

### **1.3. Backup Incremental**

O sistema de backup incremental diário garante que todas as alterações realizadas nas últimas 24 horas sejam preservadas de forma segura. O backup incremental é executado utilizando a funcionalidade de WAL (Write-Ahead Logging) do PostgreSQL, capturando todas as transações desde o último backup completo ou incremental.

Os backups incrementais são armazenados em múltiplas localizações para garantir redundância: armazenamento local para recuperação rápida, armazenamento em rede para proteção contra falhas de hardware local, e armazenamento em nuvem para proteção contra desastres que possam afetar as instalações físicas. Cada backup é verificado automaticamente após a conclusão para garantir sua integridade.

O sistema mantém um histórico de 30 dias de backups incrementais, permitindo recuperação point-in-time para qualquer momento dentro deste período. Esta capacidade é crucial para situações onde problemas são descobertos dias após sua ocorrência, permitindo recuperação precisa sem perda significativa de dados.

A documentação de cada backup inclui informações sobre tamanho, duração, verificação de integridade e localização de armazenamento. Esta documentação é essencial para planejamento de capacidade e para procedimentos de recuperação de desastres.

### **1.4. Verificação de Alertas e Notificações**

A verificação diária do sistema de alertas e notificações garante que todos os mecanismos de comunicação estejam funcionando corretamente. Esta verificação inclui o teste dos canais LISTEN/NOTIFY do PostgreSQL, que são utilizados para notificações em tempo real sobre estoque crítico, documentação irregular e novas ocorrências.

O sistema verifica se todas as notificações geradas nas últimas 24 horas foram adequadamente processadas e entregues aos destinatários apropriados. Notificações não entregues são reprocessadas automaticamente, e falhas persistentes são escaladas para investigação manual.

A verificação inclui também o teste dos alertas automáticos de estoque crítico, simulando condições que deveriam gerar alertas e verificando se as notificações são geradas corretamente. Esta verificação é essencial para garantir que situações críticas sejam comunicadas imediatamente aos responsáveis.

O sistema mantém estatísticas sobre a efetividade das notificações, incluindo tempos de entrega, taxas de leitura e ações tomadas em resposta aos alertas. Estas estatísticas são utilizadas para otimizar continuamente o sistema de notificações e garantir sua relevância e efetividade.

## **2. Rotinas de Manutenção Semanal**

### **2.1. Análise Detalhada de Performance**

A análise semanal de performance proporciona uma visão mais abrangente das tendências de utilização do sistema, permitindo identificar padrões que não são visíveis nas verificações diárias. Esta análise inclui a revisão de todas as métricas coletadas durante a semana, identificando picos de utilização, consultas que apresentaram degradação de performance e recursos que se aproximaram de limites críticos.

A análise de consultas inclui a identificação de queries que, embora não sejam individualmente problemáticas, consomem recursos significativos devido à alta frequência de execução. Estas consultas são candidatas para otimização através de ajustes de índices, reescrita de queries ou criação de views materializadas.

O sistema analisa padrões de acesso às partições, identificando partições que recebem mais acessos que o esperado ou partições antigas que ainda estão sendo consultadas frequentemente. Esta informação orienta decisões sobre estratégias de arquivamento e otimização de índices.

A análise de crescimento de dados projeta tendências futuras baseadas no crescimento observado durante a semana. Esta projeção é essencial para planejamento de capacidade e para identificar quando será necessário expandir recursos de armazenamento ou processamento.

### **2.2. Manutenção de Índices**

A manutenção semanal de índices garante que todos os índices estejam otimizados para as consultas atuais do sistema. Esta manutenção inclui a reconstrução de índices que apresentaram fragmentação significativa, a análise de utilização de índices para identificar índices subutilizados que podem ser removidos, e a identificação de oportunidades para criação de novos índices.

A análise de fragmentação é particularmente importante para índices em tabelas com alta rotatividade de dados, como as tabelas de operações e notificações. Índices fragmentados podem impactar significativamente a performance de consultas, especialmente em operações de range scan que são comuns em consultas por período.

O sistema analisa estatísticas de utilização de índices coletadas pelo PostgreSQL, identificando índices que não foram utilizados durante a semana. Índices não utilizados consomem espaço de armazenamento e recursos durante operações de escrita sem proporcionar benefícios, sendo candidatos para remoção após análise cuidadosa.

A identificação de oportunidades para novos índices é baseada na análise de consultas que apresentaram performance subótima durante a semana. O sistema sugere índices que poderiam melhorar a performance destas consultas, considerando o impacto no overhead de manutenção e no espaço de armazenamento.

### **2.3. Verificação de Backup e Recuperação**

A verificação semanal de backup inclui testes de recuperação que garantem que os backups estão funcionais e que os procedimentos de recuperação estão operacionais. Estes testes são executados em um ambiente separado para não impactar as operações de produção.

O teste de recuperação point-in-time verifica a capacidade de recuperar o banco de dados para um momento específico durante a semana anterior, utilizando a combinação de backup completo e backups incrementais. Este teste valida não apenas a integridade dos backups, mas também a funcionalidade dos procedimentos de recuperação.

A verificação de integridade dos backups arquivados inclui a validação de checksums e a verificação de que todos os arquivos necessários para recuperação estão presentes e acessíveis. Backups corrompidos ou incompletos são identificados e substituídos por cópias de backup secundárias.

O sistema testa também a recuperação de componentes específicos, como tabelas individuais ou partições específicas, validando procedimentos que podem ser necessários em cenários de recuperação parcial. Esta capacidade é importante para situações onde apenas parte dos dados foi afetada por problemas.

### **2.4. Análise de Segurança**

A análise semanal de segurança inclui a revisão de logs de acesso, identificação de padrões de uso anômalos e verificação de que todas as políticas de segurança estão sendo adequadamente aplicadas. Esta análise é crucial para manter a integridade e confidencialidade dos dados do sistema.

A revisão de logs de auditoria identifica atividades suspeitas, como tentativas de acesso não autorizado, alterações de dados fora do horário normal de trabalho, ou padrões de acesso que desviam significativamente do comportamento normal dos usuários.

Atividades suspeitas são investigadas e, se necessário, escaladas para as autoridades competentes.

A verificação de permissões garante que todos os usuários tenham apenas as permissões necessárias para suas funções, seguindo o princípio do menor privilégio. O sistema identifica usuários com permissões excessivas e sugere ajustes para reduzir riscos de segurança.

A análise inclui também a verificação de que todas as conexões ao banco de dados estão utilizando criptografia adequada e que não há conexões não autorizadas ou de origens não reconhecidas. Esta verificação é essencial para manter a confidencialidade dos dados durante a transmissão.

## **3. Rotinas de Manutenção Mensal**

### **3.1. Backup Completo e Verificação**

O backup completo mensal constitui a base do sistema de recuperação de desastres, capturando o estado completo do banco de dados em um momento específico. Este backup é executado durante um período de manutenção programada, quando o sistema pode ser temporariamente colocado em modo de leitura apenas para garantir consistência total.

O processo de backup completo inclui não apenas os dados das tabelas, mas também todos os objetos do banco de dados como índices, views, procedures, triggers e configurações. Esta abordagem garante que uma recuperação completa possa restaurar não apenas os dados, mas toda a funcionalidade do sistema.

A verificação do backup completo inclui a restauração em um ambiente de teste e a execução de uma bateria de testes que validam a integridade dos dados e a funcionalidade de todos os componentes do sistema. Esta verificação é essencial para garantir que o backup seja utilizável em uma situação real de recuperação de desastres.

O backup completo é armazenado em múltiplas localizações geográficas para proteção contra desastres regionais. Cópias são mantidas em instalações locais, em data centers remotos e em serviços de armazenamento em nuvem, garantindo que pelo menos uma cópia esteja sempre acessível mesmo em cenários de desastre extremo.

### **3.2. Análise de Crescimento e Planejamento de Capacidade**

A análise mensal de crescimento proporciona uma visão abrangente das tendências de utilização de recursos e crescimento de dados, permitindo planejamento proativo de

expansão de capacidade. Esta análise inclui projeções baseadas em dados históricos e consideração de fatores sazonais que podem afetar o crescimento.

O crescimento de dados é analisado por tabela e por partição, identificando áreas do sistema que estão crescendo mais rapidamente que o esperado. Esta análise é particularmente importante para as novas funcionalidades como registros de custódia em campo e avaliações de materiais, que podem ter padrões de crescimento diferentes das funcionalidades tradicionais.

A análise de utilização de recursos inclui tendências de CPU, memória, armazenamento e I/O, correlacionando estas métricas com o crescimento de dados e o aumento de usuários. Esta correlação permite projeções mais precisas sobre quando será necessário expandir recursos de hardware.

O planejamento de capacidade inclui recomendações específicas sobre quando e como expandir recursos, considerando fatores como lead time para aquisição de hardware, janelas de manutenção disponíveis e impacto nas operações. As recomendações incluem também estratégias alternativas como otimização de consultas ou arquivamento de dados antigos.

### **3.3. Manutenção de Partições**

A manutenção mensal de partições inclui a criação de partições futuras, o arquivamento de partições antigas e a otimização de partições ativas. Esta manutenção é essencial para manter a performance do sistema conforme os dados crescem ao longo do tempo.

A criação de partições futuras garante que o sistema continue operando sem interrupções mesmo durante períodos de alta atividade quando a criação automática de partições poderia causar contenção. O sistema cria partições para os próximos três meses, garantindo disponibilidade contínua.

O arquivamento de partições antigas move dados históricos para armazenamento de longo prazo, liberando espaço no armazenamento principal e melhorando a performance de consultas que não precisam acessar dados históricos. Partições com mais de dois anos são movidas para armazenamento de arquivo, onde permanecem acessíveis mas não impactam operações correntes.

A otimização de partições ativas inclui a reconstrução de índices fragmentados, a atualização de estatísticas para o otimizador de consultas e a verificação de que todas as partições estão adequadamente configuradas. Esta otimização garante que as partições mais utilizadas mantenham performance ideal.



### **3.4. Análise de Utilização e Otimização**

A análise mensal de utilização examina padrões de uso do sistema, identificando funcionalidades subutilizadas, consultas ineficientes e oportunidades de otimização. Esta análise é baseada em dados coletados durante todo o mês e proporciona insights valiosos para melhorias contínuas.

A análise de funcionalidades identifica módulos do sistema que são utilizados com menos frequência que o esperado, permitindo investigação sobre possíveis problemas de usabilidade ou necessidades de treinamento adicional. Funcionalidades como o sistema de avaliação de materiais podem precisar de promoção adicional para alcançar adoção completa.

A identificação de consultas ineficientes inclui não apenas consultas individuais lentas, mas também padrões de consultas que, em conjunto, consomem recursos significativos. Esta análise pode revelar oportunidades para otimização através de views materializadas, índices especializados ou reestruturação de dados.

A análise de oportunidades de otimização considera também aspectos funcionais, como a identificação de processos manuais que poderiam ser automatizados ou relatórios que poderiam ser pré-calculados para melhorar a experiência do usuário.

## **4. Rotinas de Manutenção Trimestral**

### **4.1. Revisão Abrangente de Segurança**

A revisão trimestral de segurança constitui uma análise aprofundada de todos os aspectos de segurança do sistema, incluindo controles de acesso, auditoria, criptografia e conformidade com políticas organizacionais. Esta revisão é executada por uma equipe especializada e pode incluir consultores externos para garantir objetividade.

A auditoria de controles de acesso verifica que todos os usuários tenham permissões apropriadas para suas funções atuais, identificando contas órfãs de funcionários que mudaram de função ou deixaram a organização. O sistema de Row Level Security (RLS) é testado para garantir que usuários só possam acessar dados apropriados para sua unidade e função.

A análise de logs de auditoria inclui a identificação de padrões que podem indicar tentativas de acesso não autorizado, uso inadequado de privilégios ou outras atividades suspeitas. Esta análise utiliza ferramentas automatizadas para identificar anomalias em grandes volumes de dados de auditoria.

A verificação de conformidade garante que o sistema continue atendendo a todas as regulamentações aplicáveis, incluindo leis de proteção de dados, regulamentos de segurança da informação e políticas internas da organização. Esta verificação inclui a documentação de todos os controles implementados e evidências de sua efetividade.

## **4.2. Teste de Recuperação de Desastres**

O teste trimestral de recuperação de desastres simula cenários realistas de falha para validar que os procedimentos de recuperação são efetivos e que os objetivos de tempo de recuperação (RTO) e ponto de recuperação (RPO) podem ser atendidos. Estes testes são executados em ambiente separado para não impactar as operações de produção.

O teste de recuperação completa simula a perda total do sistema de produção, validando a capacidade de restaurar completamente o sistema a partir de backups. Este teste inclui não apenas a restauração dos dados, mas também a reconfiguração de todos os componentes do sistema e a validação de que todas as funcionalidades estão operacionais.

O teste de recuperação parcial simula cenários onde apenas parte do sistema é afetada, como a corrupção de uma partição específica ou a falha de um subsistema. Estes testes validam procedimentos de recuperação granular que podem ser mais apropriados para certos tipos de problemas.

A validação de procedimentos inclui a verificação de que toda a documentação de recuperação está atualizada e que todos os membros da equipe responsável pela recuperação estão adequadamente treinados. O teste identifica lacunas na documentação ou no treinamento que precisam ser corrigidas.

## **4.3. Análise de Performance e Otimização**

A análise trimestral de performance proporciona uma visão de longo prazo das tendências de utilização do sistema, permitindo identificar padrões sazonais e planejar otimizações estratégicas. Esta análise inclui correlação entre performance e eventos operacionais, como grandes emergências que podem impactar significativamente a utilização do sistema.

A análise de tendências de performance identifica degradações graduais que podem não ser visíveis em análises de curto prazo. Por exemplo, consultas que se tornam progressivamente mais lentas conforme os dados crescem podem ser identificadas e otimizadas antes que se tornem problemáticas.

A avaliação de arquitetura examina se a estrutura atual do banco de dados ainda é adequada para os padrões de uso observados. Esta avaliação pode recomendar

mudanças estruturais como redistribuição de dados, criação de novos índices especializados ou implementação de novas estratégias de particionamento.

A análise inclui também a avaliação de novas funcionalidades do PostgreSQL que podem beneficiar o sistema, como novos tipos de índices, funcionalidades de paralelização ou melhorias no otimizador de consultas. A adoção de novas funcionalidades é planejada cuidadosamente para minimizar riscos.

#### **4.4. Revisão de Procedimentos e Documentação**

A revisão trimestral de procedimentos garante que toda a documentação operacional esteja atualizada e que os procedimentos reflitam as melhores práticas atuais. Esta revisão é especialmente importante após a implementação de novas funcionalidades ou mudanças significativas no sistema.

A atualização de procedimentos operacionais inclui a incorporação de lições aprendidas durante o trimestre, ajustes baseados em mudanças no ambiente operacional e melhorias identificadas através da experiência prática. Procedimentos para as novas funcionalidades como gestão de custódia em campo são refinados baseados no uso real.

A revisão de documentação técnica garante que todas as especificações, diagramas e manuais estejam atualizados e reflitam o estado atual do sistema. Esta documentação é essencial para manutenção efetiva e para treinamento de novos membros da equipe.

A validação de procedimentos de emergência inclui a verificação de que todos os procedimentos para situações críticas estão atualizados e que todos os membros da equipe estão familiarizados com suas responsabilidades. Esta validação pode incluir simulações de cenários de emergência para testar a efetividade dos procedimentos.

### **5. Rotinas de Manutenção Anual**

#### **5.1. Revisão Estratégica Completa**

A revisão anual constitui uma avaliação abrangente de todos os aspectos do sistema, incluindo arquitetura, performance, segurança, conformidade e alinhamento com objetivos organizacionais. Esta revisão é conduzida por uma equipe multidisciplinar que inclui administradores de banco de dados, arquitetos de sistemas, especialistas em segurança e representantes dos usuários finais.

A avaliação de arquitetura examina se a estrutura atual do sistema ainda é adequada para as necessidades da organização, considerando mudanças nos requisitos

operacionais, crescimento de dados e evolução tecnológica. Esta avaliação pode recomendar mudanças significativas na arquitetura para melhorar performance, escalabilidade ou manutenibilidade.

A análise de alinhamento estratégico verifica se o sistema continua atendendo aos objetivos organizacionais e se há oportunidades para melhor suporte às operações do CBM MT. Esta análise considera mudanças na missão da organização, novos requisitos regulamentares e evolução das melhores práticas na área.

A avaliação de tecnologia examina novas tecnologias e funcionalidades que podem beneficiar o sistema, incluindo novas versões do PostgreSQL, ferramentas de monitoramento, soluções de backup e tecnologias de alta disponibilidade. A adoção de novas tecnologias é planejada considerando benefícios, riscos e recursos necessários.

## **5.2. Planejamento de Capacidade de Longo Prazo**

O planejamento anual de capacidade projeta necessidades de recursos para os próximos três a cinco anos, baseado em tendências históricas, planos organizacionais e projeções de crescimento. Este planejamento é essencial para garantir que o sistema possa suportar o crescimento futuro sem degradação de performance.

A projeção de crescimento de dados considera não apenas tendências históricas, mas também mudanças planejadas nas operações da organização, como expansão de unidades, implementação de novas funcionalidades ou mudanças nos procedimentos operacionais. Estas projeções orientam decisões sobre expansão de armazenamento e arquivamento de dados.

O planejamento de recursos de processamento considera o crescimento esperado no número de usuários, aumento na complexidade das consultas e implementação de novas funcionalidades que podem impactar a utilização de CPU e memória. Este planejamento inclui estratégias para escalabilidade horizontal e vertical.

A análise de custo-benefício avalia diferentes estratégias para atender às necessidades futuras de capacidade, considerando fatores como custo de hardware, licenciamento de software, custos operacionais e impacto na disponibilidade do sistema. Esta análise orienta decisões de investimento de longo prazo.

## **5.3. Atualização de Versão e Modernização**

A revisão anual inclui a avaliação de atualizações de versão do PostgreSQL e outros componentes do sistema, considerando benefícios de performance, segurança e funcionalidade versus riscos e esforço de migração. Atualizações de versão são planejadas cuidadosamente para minimizar impacto nas operações.

A avaliação de benefícios de atualização inclui melhorias de performance, novas funcionalidades que podem beneficiar o sistema, correções de segurança e melhor suporte para hardware moderno. Estes benefícios são quantificados sempre que possível para facilitar decisões de investimento.

A análise de riscos de atualização considera compatibilidade de aplicações, necessidade de retreinamento de equipe, potencial impacto na disponibilidade durante a migração e riscos de problemas não identificados durante testes. Estratégias de mitigação são desenvolvidas para todos os riscos identificados.

O planejamento de migração inclui cronograma detalhado, procedimentos de rollback, critérios de sucesso e planos de contingência. A migração é testada extensivamente em ambiente de desenvolvimento antes da implementação em produção.

## **5.4. Revisão de Conformidade e Auditoria**

A auditoria anual constitui uma revisão abrangente de todos os aspectos de conformidade do sistema, incluindo aderência a regulamentações, políticas organizacionais e melhores práticas da indústria. Esta auditoria pode ser conduzida por auditores internos ou externos, dependendo dos requisitos organizacionais.

A verificação de conformidade regulamentária garante que o sistema continue atendendo a todas as leis e regulamentos aplicáveis, incluindo proteção de dados, segurança da informação e requisitos específicos para organizações públicas. Esta verificação inclui documentação de todos os controles implementados.

A auditoria de segurança examina todos os aspectos de segurança do sistema, incluindo controles de acesso, criptografia, auditoria, backup e recuperação. Esta auditoria identifica vulnerabilidades potenciais e recomenda melhorias para fortalecer a postura de segurança.

A revisão de procedimentos operacionais verifica que todos os procedimentos estão sendo seguidos adequadamente e que são efetivos para seus propósitos. Esta revisão pode identificar oportunidades para simplificação de procedimentos ou automação de tarefas manuais.

# **6. Procedimentos de Emergência**

## **6.1. Resposta a Falhas Críticas**

Os procedimentos de emergência definem ações específicas para diferentes tipos de falhas críticas que podem afetar a disponibilidade ou integridade do sistema. Estes

procedimentos são projetados para minimizar o tempo de inatividade e preservar a integridade dos dados durante situações de crise.

Para falhas de hardware que afetam o servidor principal, o procedimento inclui avaliação rápida da extensão do problema, ativação de sistemas de backup se disponíveis, e início do processo de recuperação a partir de backups se necessário. O procedimento define claramente as responsabilidades de cada membro da equipe e os critérios para escalção.

Para corrupção de dados, o procedimento inclui isolamento imediato da área afetada para prevenir propagação do problema, avaliação da extensão da corrupção, e determinação da estratégia de recuperação mais apropriada. O procedimento considera diferentes cenários, desde corrupção de registros individuais até corrupção de partições inteiras.

Para ataques de segurança ou acesso não autorizado, o procedimento inclui isolamento imediato do sistema, preservação de evidências para investigação, avaliação da extensão do comprometimento, e implementação de medidas de contenção. O procedimento define também quando e como comunicar o incidente às autoridades competentes.

## **6.2. Comunicação de Emergência**

Os procedimentos de comunicação de emergência garantem que todas as partes interessadas sejam informadas adequadamente sobre problemas críticos e sobre o progresso das ações de recuperação. Esta comunicação é essencial para manter a confiança dos usuários e para coordenar ações de resposta.

A lista de contatos de emergência inclui todos os membros da equipe técnica, gestores responsáveis, fornecedores críticos e autoridades relevantes. Esta lista é mantida atualizada e inclui múltiplos métodos de contato para garantir que as pessoas possam ser alcançadas mesmo fora do horário normal de trabalho.

Os modelos de comunicação definem mensagens padrão para diferentes tipos de emergência, garantindo que informações essenciais sejam comunicadas de forma clara e consistente. Estes modelos incluem informações sobre a natureza do problema, impacto esperado, ações sendo tomadas e estimativas de tempo para resolução.

O procedimento de escalção define quando e como escalar problemas para níveis superiores de gestão, considerando fatores como duração do problema, impacto nas operações e recursos necessários para resolução. A escalção garante que recursos adequados sejam mobilizados rapidamente para problemas críticos.

## 6.3. Recuperação e Validação

Os procedimentos de recuperação definem passos específicos para restaurar o sistema após diferentes tipos de falhas, garantindo que a recuperação seja executada de forma consistente e que a integridade dos dados seja preservada. Estes procedimentos são testados regularmente para garantir sua efetividade.

Para recuperação a partir de backups, o procedimento inclui seleção do backup apropriado, validação de sua integridade, execução da restauração, e verificação de que todos os dados foram recuperados corretamente. O procedimento considera diferentes cenários de recuperação, desde recuperação completa até recuperação de componentes específicos.

Para recuperação de falhas de hardware, o procedimento inclui instalação e configuração de hardware de substituição, restauração do sistema operacional e software, recuperação dos dados, e validação de que todas as funcionalidades estão operacionais. O procedimento define também como minimizar o tempo de inatividade durante a recuperação.

A validação pós-recuperação inclui testes abrangentes de todas as funcionalidades críticas do sistema, verificação de integridade dos dados, e confirmação de que a performance está dentro de parâmetros aceitáveis. Esta validação garante que o sistema esteja completamente funcional antes de retornar às operações normais.

## 7. Monitoramento e Alertas

### 7.1. Sistema de Monitoramento Contínuo

O sistema de monitoramento contínuo coleta métricas em tempo real sobre todos os aspectos críticos do sistema, incluindo performance, disponibilidade, integridade dos dados e segurança. Este monitoramento é essencial para identificar problemas antes que afetem os usuários e para manter visibilidade sobre a saúde geral do sistema.

As métricas de performance incluem tempo de resposta de consultas, utilização de CPU e memória, throughput de I/O e utilização de conexões. Estas métricas são coletadas continuamente e analisadas para identificar tendências que podem indicar problemas emergentes ou necessidades de otimização.

O monitoramento de disponibilidade verifica continuamente que todos os componentes críticos do sistema estão funcionais e acessíveis. Este monitoramento inclui verificação de conectividade, responsividade de serviços e integridade de funcionalidades essenciais como backup e replicação.

O monitoramento de segurança inclui detecção de tentativas de acesso não autorizado, monitoramento de atividades suspeitas e verificação de que todos os controles de segurança estão funcionando adequadamente. Este monitoramento é integrado com sistemas de resposta a incidentes para garantir resposta rápida a ameaças.

## **7.2. Configuração de Alertas**

O sistema de alertas é configurado para notificar automaticamente a equipe responsável sobre condições que requerem atenção imediata ou que podem indicar problemas emergentes. Os alertas são categorizados por severidade e configurados com diferentes métodos de notificação baseados na urgência.

Alertas críticos são configurados para condições que afetam imediatamente a disponibilidade ou integridade do sistema, como falhas de hardware, corrupção de dados ou ataques de segurança. Estes alertas são enviados imediatamente através de múltiplos canais para garantir resposta rápida.

Alertas de aviso são configurados para condições que podem se tornar problemáticas se não forem endereçadas, como utilização alta de recursos, degradação de performance ou aproximação de limites de capacidade. Estes alertas permitem ação proativa antes que problemas se tornem críticos.

Alertas informativos são configurados para eventos que devem ser registrados mas não requerem ação imediata, como conclusão de backups, criação de novas partições ou execução de manutenção programada. Estes alertas mantêm a equipe informada sobre atividades do sistema.

## **7.3. Dashboards e Relatórios**

Os dashboards de monitoramento proporcionam visibilidade em tempo real sobre o status de todos os componentes críticos do sistema, permitindo que a equipe identifique rapidamente problemas e monitore tendências. Estes dashboards são acessíveis através de interfaces web e podem ser exibidos em monitores dedicados.

O dashboard principal inclui métricas de alto nível sobre performance, disponibilidade e utilização de recursos, proporcionando uma visão geral rápida da saúde do sistema. Este dashboard é projetado para permitir identificação rápida de problemas mesmo por pessoal não técnico.

Dashboards especializados proporcionam visões detalhadas de aspectos específicos do sistema, como performance de consultas, utilização de partições, efetividade de backups e atividade de segurança. Estes dashboards são utilizados pela equipe técnica para análise detalhada e troubleshooting.



Os relatórios automatizados são gerados periodicamente para documentar tendências de longo prazo, efetividade de procedimentos de manutenção e conformidade com objetivos de nível de serviço. Estes relatórios são utilizados para planejamento estratégico e para comunicação com gestão superior.

## Conclusão

O plano de manutenção apresentado proporciona uma estrutura abrangente para garantir que o Sistema Integrado de Gestão de Almoxarifado do CBM MT mantenha performance, integridade, segurança e disponibilidade ideais ao longo de sua vida útil. A implementação consistente deste plano é essencial para o sucesso operacional do sistema e para a confiança dos usuários.

As rotinas de manutenção são projetadas para serem proativas, identificando e resolvendo problemas antes que afetem as operações. A combinação de verificações automatizadas e análises manuais garante que todos os aspectos críticos do sistema sejam adequadamente monitorados e mantidos.

A estrutura de emergência proporciona procedimentos claros para resposta a situações críticas, minimizando o impacto de problemas inesperados e garantindo recuperação rápida. A documentação detalhada e o treinamento regular da equipe são essenciais para a efetividade destes procedimentos.

O sistema de monitoramento contínuo e alertas garante visibilidade constante sobre a saúde do sistema e resposta rápida a condições que requerem atenção. Esta visibilidade é fundamental para manter a confiança dos usuários e para garantir que o sistema continue atendendo às necessidades operacionais da organização.

A implementação bem-sucedida deste plano de manutenção requer comprometimento organizacional, recursos adequados e treinamento contínuo da equipe. Com estes elementos em lugar, o sistema SIGA continuará proporcionando valor significativo para as operações do CBM MT por muitos anos.

---

**Autor:** Manus AI

**Data:** 18 de junho de 2025

**Sistema:** SIGA - Sistema Integrado de Gestão de Almoxarifado

**Organização:** Corpo de Bombeiros Militar de Mato Grosso

# PROJETO DE ENGENHARIA DE SOFTWARE

## Sistema Integrado de Gestão de Almoxarifado - BEA/ CBM MT

**Versão:** 1.1

**Data:** 11 de junho de 2025

**Autor:** SD Walingson 4ºBBM / Equipe de Desenvolvimento BEA/CBM MT

**Classificação:** Uso Interno - CBM MT

---

## SUMÁRIO EXECUTIVO

O presente documento apresenta o projeto de engenharia de software para desenvolvimento do Sistema Integrado de Gestão de Almoxarifado (SIGA) do Batalhão de Emergências Ambientais do Corpo de Bombeiros Militar de Mato Grosso. Este sistema visa modernizar e centralizar o controle de materiais, equipamentos e ativos da corporação, substituindo os métodos manuais atualmente utilizados por uma solução informatizada robusta, segura e escalável.

O projeto foi estruturado considerando as limitações orçamentárias iniciais, a necessidade de entregas incrementais e a complexidade operacional do BEA durante o período crítico de combate a incêndios florestais. A estratégia de desenvolvimento modular permitirá a implementação gradual das funcionalidades, priorizando os processos mais críticos e gerando valor imediato para a instituição.

A solução proposta utilizará tecnologias já consolidadas na instituição (Scriptcase 9.4 e PostgreSQL), garantindo compatibilidade com os sistemas existentes e facilitando a manutenção futura. O desenvolvimento será realizado inicialmente em ambiente controlado, com migração posterior para a infraestrutura oficial do Estado de Mato Grosso.

# **1. ANÁLISE DE REQUISITOS E ESPECIFICAÇÃO DO SISTEMA**

## **1.1 Contexto Organizacional**

O Batalhão de Emergências Ambientais do Corpo de Bombeiros Militar de Mato Grosso representa uma das unidades mais estratégicas da corporação, sendo responsável pela coordenação logística e operacional do combate a incêndios florestais em todo o território estadual. Durante o período crítico, que se estende de junho a dezembro, a unidade gerencia um volume extraordinário de recursos materiais e humanos, coordenando operações que podem envolver centenas de militares distribuídos em diferentes regiões do estado.

A complexidade operacional do BEA transcende a simples gestão de estoque tradicional. A unidade deve coordenar simultaneamente o controle de materiais de consumo rápido, equipamentos permanentes de alto valor, viaturas especializadas, máquinas pesadas e ferramentas específicas para combate a incêndios. Cada categoria de material possui características únicas de controle, manutenção e distribuição, exigindo um sistema capaz de adaptar-se a essa diversidade operacional.

A dinâmica de trabalho durante as operações de combate a incêndios impõe desafios adicionais significativos. Materiais são constantemente emprestados, transferidos ou distribuídos para equipes que operam em campo por períodos de aproximadamente doze dias. Muitos equipamentos de uso coletivo circulam entre diferentes grupos operacionais, nem sempre retornando à unidade de origem, criando lacunas no controle patrimonial que comprometem a eficiência logística e a prestação de contas.

A ausência de um sistema informatizado padronizado resulta em cada quartel desenvolvendo métodos próprios de controle, baseados em registros manuais ou planilhas eletrônicas rudimentares. Esta fragmentação metodológica gera inconsistências nos dados, dificulta a consolidação de informações gerenciais e aumenta significativamente o risco de perdas materiais e extravios patrimoniais.

## **1.2 Problemas Identificados**

### **1.2.1 Falta de Padronização nos Processos**

A inexistência de procedimentos padronizados para gestão de almoxarifado resulta em cada unidade operacional desenvolvendo métodos próprios de controle. Esta fragmentação metodológica compromete a qualidade dos dados, dificulta a consolidação de informações e impede a implementação de controles efetivos de auditoria. A variabilidade nos processos também dificulta a transferência de conhecimento entre unidades e a capacitação de novos militares responsáveis pela gestão de materiais.

### **1.2.2 Baixa Rastreabilidade de Movimentações**

O controle manual das movimentações de materiais resulta em registros incompletos ou inexistentes de transferências, empréstimos, devoluções e manutenções. Esta deficiência compromete gravemente a capacidade de rastreamento patrimonial, dificultando a localização de equipamentos específicos durante operações críticas e impossibilitando a identificação precisa de responsabilidades em casos de perdas ou avarias.

### **1.2.3 Gestão Inadequada de Contratos e Doações**

A diversidade de fontes de aquisição de materiais, incluindo contratos governamentais, doações de entidades privadas e transferências entre órgãos públicos, cria complexidades adicionais no processo de patrimonialização e controle. A falta de um sistema integrado para acompanhar o status desses processos resulta em atrasos na disponibilização de materiais para uso operacional e dificulta o cumprimento de obrigações contratuais e prestação de contas.

### **1.2.4 Controle Deficiente de Manutenção Preventiva**

Viaturas e equipamentos especializados requerem manutenções regulares baseadas em quilometragem, horas de uso ou períodos específicos. A ausência de um sistema automatizado para controlar esses vencimentos resulta em manutenções atrasadas, comprometendo a disponibilidade operacional dos equipamentos e aumentando os custos de reparo devido à falta de prevenção adequada.

### **1.2.5 Ausência de Informações Gerenciais**

A dispersão e inconsistência dos dados impossibilitam a geração de relatórios gerenciais confiáveis, comprometendo a capacidade de tomada de decisões estratégicas. Gestores não conseguem visualizar de forma centralizada a localização, quantidade e estado dos materiais, prejudicando o planejamento logístico e podendo comprometer a efetividade das operações de combate a incêndios.

## **1.3 Requisitos Funcionais**

### **1.3.1 Módulo de Controle de Entrada e Saída**

#### **RF001 - Registro de Entrada de Materiais**

O sistema deve permitir o registro detalhado de entrada de materiais no almoxarifado, incluindo informações sobre fornecedor, número do contrato ou processo de doação, data de recebimento, quantidade, valor unitário, especificações técnicas e responsável pelo recebimento. O sistema deve gerar automaticamente códigos únicos de identificação para cada item e permitir a anexação de documentos digitalizados relacionados à aquisição.

#### **RF002 - Controle de Saída de Materiais**

O sistema deve registrar todas as saídas de materiais do almoxarifado, identificando o solicitante, a finalidade da retirada, a quantidade, a data prevista de devolução (quando aplicável) e o responsável pela autorização. Deve ser possível categorizar as saídas como empréstimo, transferência definitiva, consumo ou baixa patrimonial, cada uma com fluxos específicos de aprovação.

#### **RF003 - Gestão de Devoluções**

O sistema deve controlar o processo de devolução de materiais emprestados, registrando a data de retorno, o estado de conservação do item e eventuais avarias identificadas. Deve gerar alertas automáticos para devoluções em atraso e permitir a aplicação de penalidades ou procedimentos disciplinares quando necessário.

### **1.3.2 Módulo de Inventário de Materiais**

#### **RF004 - Cadastro Abrangente de Materiais**

O sistema deve permitir o cadastro detalhado de todos os tipos de materiais gerenciados pelo BEA, incluindo materiais de consumo, equipamentos permanentes, viaturas, máquinas e ferramentas. Para cada item, deve ser possível registrar especificações técnicas, fornecedor, data de aquisição, valor, localização atual, responsável pela guarda e histórico de movimentações.

#### **RF005 - Controle de Estoque em Tempo Real**

O sistema deve manter atualização automática dos níveis de estoque, considerando entradas, saídas, transferências e consumos. Deve calcular automaticamente pontos de reposição baseados no histórico de consumo e gerar alertas quando os níveis mínimos forem atingidos, facilitando o planejamento de aquisições.

#### **RF006 - Gestão de Localização Física**

O sistema deve permitir o controle detalhado da localização física dos materiais dentro do almoxarifado, incluindo prédio, sala, estante, prateleira e posição específica. Deve facilitar a localização rápida de itens durante operações e permitir a otimização do layout do almoxarifado baseada na frequência de uso dos materiais.

### **1.3.3 Módulo de Gestão de Transferências**

#### **RF007 - Transferências Entre Unidades**

O sistema deve gerenciar transferências de materiais entre diferentes unidades do CBM MT, controlando o processo desde a solicitação até a confirmação de recebimento. Deve manter rastreabilidade completa da movimentação e atualizar automaticamente os estoques das unidades envolvidas.

#### **RF008 - Controle de Transferências Temporárias**

O sistema deve distinguir entre transferências definitivas e temporárias, controlando prazos de retorno para transferências temporárias e gerando alertas para acompanhamento. Deve permitir a extensão de prazos mediante justificativa e aprovação adequada.

### **1.3.4 Módulo de Concessão e Cautela**

#### **RF009 - Gestão de Concessões**

O sistema deve controlar a concessão de materiais para uso pessoal de militares, registrando o responsável, o período de concessão, as condições de uso e as responsabilidades assumidas. Deve gerar termos de responsabilidade automáticos e controlar a devolução dos itens concedidos.

#### **RF010 - Controle de Cautelas**

O sistema deve gerenciar o processo de cautela de equipamentos e materiais, registrando detalhadamente as condições de entrega, as responsabilidades do militar cautelário e os prazos de devolução. Deve permitir inspeções periódicas e registrar o estado de conservação dos itens cautelados.

### **1.3.5 Módulo de Logística**

#### **RF011 - Planejamento Logístico**

O sistema deve auxiliar no planejamento logístico das operações, permitindo a reserva antecipada de materiais para operações específicas, calculando necessidades baseadas no histórico de consumo e facilitando a distribuição eficiente de recursos entre diferentes frentes de trabalho.

#### **RF012 - Controle de Distribuição**

O sistema deve controlar a distribuição de materiais para operações de campo, registrando quais materiais foram enviados para cada operação, quem são os responsáveis no campo e quando os materiais devem retornar à base.

## **1.4 Requisitos Não Funcionais**

### **1.4.1 Performance e Escalabilidade**

#### **RNF001 - Tempo de Resposta**

O sistema deve apresentar tempo de resposta inferior a 3 segundos para consultas simples e inferior a 10 segundos para relatórios complexos, mesmo com o banco de dados contendo mais de 100.000 registros de movimentações. Esta performance deve ser mantida com até 50 usuários simultâneos acessando o sistema.

#### **RNF002 - Capacidade de Armazenamento**

O sistema deve ser capaz de armazenar pelo menos 5 anos de histórico completo de movimentações, incluindo documentos digitalizados anexados aos registros. Deve suportar o crescimento do volume de dados sem degradação significativa da performance.

### **1.4.2 Segurança e Auditoria**

#### **RNF003 - Controle de Acesso**

O sistema deve implementar controle de acesso baseado em perfis hierárquicos, respeitando a estrutura militar da instituição. Deve registrar todas as ações dos usuários em log de auditoria e permitir a rastreabilidade completa de todas as operações realizadas no sistema.

#### **RNF004 - Integridade dos Dados**

O sistema deve garantir a integridade dos dados através de validações rigorosas, backup automático diário e mecanismos de recuperação em caso de falhas. Deve implementar controles que impeçam a exclusão ou alteração indevida de registros históricos.

### **1.4.3 Usabilidade e Acessibilidade**

#### **RNF005 - Interface Intuitiva**

O sistema deve apresentar interface web responsiva, compatível com dispositivos móveis e tablets, facilitando o uso em campo durante operações. A interface deve ser intuitiva, minimizando a necessidade de treinamento extensivo para novos usuários.

#### **RNF006 - Disponibilidade**

O sistema deve apresentar disponibilidade mínima de 99% durante o horário comercial e 95% durante operações noturnas e fins de semana. Deve implementar mecanismos de recuperação automática em caso de falhas menores.

## **1.5 Regras de Negócio**

### **1.5.1 Regras de Movimentação**

#### **RN001 - Autorização de Saídas**

Toda saída de material do almoxarifado deve ser autorizada por militar responsável pelo almoxarifado.

#### **RN002 - Limite de Empréstimos**

Não há limites de empréstimos. Limitado apenas pelo julgamento do responsável pelo almoxarifado. Notificar o administrador do almoxarifado caso tenha pendências do solicitante.

#### **RN003 - Prazo de Devolução**

O prazo padrão para devolução de materiais emprestados é de 15 dias, podendo ser estendido por igual período mediante justificativa aprovada pelo comandante da unidade. Empréstimos superiores a 30 dias devem ser convertidos em transferência temporária.

### **1.5.2 Regras de Controle Patrimonial**

#### **RN004 - Classificação de Materiais**

Materiais permanentes e de alto valor agregado devem ser obrigatoriamente patrimonializados e receber plaqueta de identificação. Materiais de consumo com validade devem ter controle de lote e data de vencimento e número do contrato de aquisição ou termo de doação.

#### **RN005 - Manutenção Preventiva**

Viaturas devem passar por manutenção preventiva a cada 10.000 km ou 6 meses, o que ocorrer primeiro. Equipamentos de combate a incêndio devem ser inspecionados mensalmente durante o período operacional (junho a dezembro).

#### **RN006 - Baixa Patrimonial**

A baixa patrimonial de equipamentos deve ser aprovada por comissão específica e registrada com justificativa detalhada.

## **1.6 Casos de Uso Principais**

### **1.6.1 UC001 - Registrar Entrada de Material**



**Ator Principal:** Responsável pelo Almoxarifado

**Pré-condições:** Usuário autenticado com perfil de almoxarife

**Fluxo Principal:**

1. O almoxarife acessa a funcionalidade de registro de entrada
2. O sistema apresenta formulário para preenchimento dos dados do material
3. O almoxarife informa: tipo de material, descrição, quantidade, fornecedor, número do contrato/processo, valor unitário, data de recebimento
4. O sistema valida os dados informados
5. O sistema gera código único de identificação para o material
6. O sistema registra a entrada no banco de dados
7. O sistema atualiza automaticamente o estoque
8. O sistema gera comprovante de entrada
9. O sistema gera Qrcode com as informações do material

**Fluxos Alternativos:**

- 4a. Dados inválidos: Sistema apresenta mensagens de erro e solicita correção
- 6a. Falha na gravação: Sistema apresenta mensagem de erro e permite nova tentativa

**Pós-condições:** Material registrado no sistema com estoque atualizado

### 1.6.2 UC002 - Processar Saída de Material

**Ator Principal:** Responsável pelo Almoxarifado

**Atores Secundários:** Solicitante, Autorizador

**Pré-condições:** Material disponível em estoque, usuário autenticado

**Fluxo Principal:**

1. O solicitante acessa o sistema e solicita material
2. O sistema verifica disponibilidade em estoque
3. O sistema encaminha solicitação para aprovação
4. O autorizador analisa e aprova a solicitação
5. O almoxarife processa a saída física do material
6. O sistema registra a saída e atualiza o estoque
7. O sistema gera termo de responsabilidade (se aplicável)

**Fluxos Alternativos:**

- 2a. Material indisponível: Sistema informa indisponibilidade e sugere alternativas
- 4a. Solicitação negada: Sistema notifica solicitante com justificativa

**1.6.3 5a. Material não localizado: Sistema**

registra ocorrência para investigação **UC003** -

**Controlar Devolução de Material Ator**

**Principal:** Responsável pelo Almoxarifado

**Pré-condições:** Existe empréstimo ativo do material

**Fluxo Principal:**

1. O responsável pelo material comparece para devolução
2. O almoxarife localiza o registro de empréstimo no sistema
3. O almoxarife inspeciona o estado de conservação do material
4. O sistema registra a devolução com observações sobre o estado
5. O sistema atualiza o estoque e encerra o empréstimo
6. O sistema gera comprovante de devolução

**Fluxos Alternativos:**

- 3a. Material com avarias: Sistema registra avarias e inicia processo de responsabilização

- 4a. Devolução em atraso: Sistema calcula penalidade conforme regulamento

**1.6.4 UC004 - Realizar Inventário**

**Ator Principal:** Comissão de Inventário

**Pré-condições:** Período de inventário autorizado pelo comando

**Fluxo Principal:**

1. A comissão acessa a funcionalidade de inventário
2. O sistema gera lista de materiais para conferência física
3. A comissão realiza contagem física dos materiais
4. O sistema registra as quantidades encontradas
5. O sistema compara com estoque teórico e identifica divergências
6. A comissão investiga e justifica as divergências encontradas
7. O sistema ajusta o estoque conforme aprovação da comissão
8. O sistema gera relatório final do inventário

**1.6.5 UC005 - Gerar Relatórios Gerenciais**

**Ator Principal:** Gestor/Comandante

**Pré-condições:** Usuário com perfil gerencial autenticado

**Fluxo Principal:**

1. O gestor acessa a área de relatórios
2. O sistema apresenta opções de relatórios disponíveis
3. O gestor seleciona tipo de relatório e define parâmetros
4. O sistema processa os dados e gera o relatório
5. O sistema apresenta o relatório na tela
6. O gestor pode exportar o relatório em PDF ou Excel

**Tipos de Relatórios:**

- Posição de estoque por categoria
- Movimentações por período
- Materiais em atraso de devolução
- Manutenções vencidas
- Análise de consumo
- Relatório de inventário

## **1.7 Interfaces e Integrações**

### **1.7.1 Integração com Sistemas Existentes**

O Sistema Integrado de Gestão de Almoxarifado será desenvolvido para integrar-se harmoniosamente com os sistemas já existentes no CBM MT, todos construídos na plataforma Scriptcase. Esta integração permitirá o compartilhamento de informações críticas e evitará a duplicação de dados entre sistemas.

#### **Integração com Sistema de Recursos Humanos**

O SIGA deve integrar-se com o sistema de RH para obter informações atualizadas sobre militares, incluindo patente, unidade de lotação, situação funcional e histórico disciplinar. Esta integração permitirá validações automáticas de autorização para empréstimos e transferências, além de facilitar a identificação de responsáveis por materiais.

#### **Integração com Sistema Financeiro**

A integração com o sistema financeiro permitirá o controle automático de orçamentos destinados à aquisição de materiais, verificação de disponibilidade orçamentária antes de processos de compra e consolidação de informações para prestação de contas aos órgãos de controle.

#### **Integração com Sistema de Manutenção de Viaturas**

O SIGA deve compartilhar informações sobre viaturas com o sistema de manutenção, permitindo o controle integrado de quilometragem, histórico de manutenções e agendamento de serviços preventivos baseados no uso operacional registrado no almoxarifado.

### **1.7.2 Interfaces Externas**

#### **Interface com Fornecedores**

O sistema deve permitir a integração futura com sistemas de fornecedores para automatização de processos de cotação, pedidos de compra e confirmação de entregas. Esta interface facilitará o processo de aquisição e reduzirá o tempo de disponibilização de materiais.

#### **Interface com Órgãos de Controle**

O sistema deve gerar relatórios padronizados para atendimento às demandas dos órgãos de controle interno e externo, incluindo Tribunal de Contas, Controladoria Geral do Estado e auditoria interna do CBM MT.

## **1.8 Restrições Técnicas e Organizacionais**

### **1.8.1 Restrições Tecnológicas**

#### **Plataforma de Desenvolvimento**

O sistema deve ser desenvolvido obrigatoriamente na plataforma Scriptcase 9.4, utilizando banco de dados PostgreSQL, para manter compatibilidade com a infraestrutura tecnológica existente no CBM MT e facilitar a manutenção por equipes internas. Aplicações em Python também pode ser utilizadas para funções auxiliares do sistema como Ciência de dados e Machine Learning por exemplo.

#### **Ambiente de Desenvolvimento Inicial**

Durante a fase inicial de desenvolvimento (30 dias), o sistema será desenvolvido em ambiente pessoal com servidor Scriptcase Host e capacidade de 8GB. Esta restrição impõe limitações temporárias de performance e capacidade que devem ser consideradas no planejamento das entregas.

#### **Migração para Infraestrutura Oficial**

Após a aprovação do projeto piloto, o sistema deve ser migrado para a infraestrutura oficial do Estado de Mato Grosso, que oferece recursos ampliados de processamento, armazenamento e licenciamento para múltiplos desenvolvedores.

### **1.8.2 Restrições Organizacionais**

### **Recursos Humanos Limitados**

O desenvolvimento inicial será realizado por um único desenvolvedor, impondo limitações no escopo das entregas e na velocidade de implementação. Esta restrição exige priorização rigorosa das funcionalidades e estratégia de desenvolvimento modular.

### **Ausência de Orçamento Inicial**

A fase inicial do projeto não conta com orçamento específico, dependendo de recursos pessoais do desenvolvedor. Esta limitação impacta a capacidade de aquisição de ferramentas adicionais, contratação de serviços especializados e implementação de funcionalidades que demandem recursos externos.

### **Necessidade de Aprovação Hierárquica**

Todas as decisões técnicas e funcionais significativas devem ser aprovadas pela cadeia de comando do BEA, incluindo coronel comandante e oficiais. Esta estrutura hierárquica pode impactar os prazos de desenvolvimento e exige documentação detalhada de todas as propostas.

### **1.8.3 Restrições Regulamentares**

#### **Conformidade com Legislação Pública**

O sistema deve atender integralmente à legislação aplicável à administração pública, incluindo Lei de Licitações, Lei de Acesso à Informação, Lei Geral de Proteção de Dados e regulamentos específicos do CBM MT.

#### **Auditabilidade Completa**

Todas as operações do sistema devem ser auditáveis, mantendo logs detalhados de ações, histórico de alterações e trilha de aprovações. Esta exigência é fundamental para atendimento aos órgãos de controle e transparência na gestão pública.

#### **Segurança da Informação**

O sistema deve implementar controles rigorosos de segurança da informação, incluindo criptografia de dados sensíveis, controle de acesso baseado em perfis e backup seguro de informações críticas.

## **2. ARQUITETURA E DESIGN DO SISTEMA**

### **2.1 Visão Geral da Arquitetura**

O Sistema Integrado de Gestão de Almoxarifado (SIGA) foi projetado seguindo

uma arquitetura em camadas que garante escalabilidade, manutenibilidade e segurança. A solução adota o padrão Model-View-Controller (MVC) nativo da plataforma Scriptcase, proporcionando separação clara entre lógica de negócio, apresentação e controle de dados.

A arquitetura proposta considera as limitações iniciais de infraestrutura e a necessidade de migração futura para ambiente de produção mais robusto. O design modular permite

implementação incremental das funcionalidades, facilitando testes, validação e ajustes durante o processo de desenvolvimento.

### **2.1.1 Princípios Arquiteturais**

#### **Modularidade**

O sistema é estruturado em módulos funcionais independentes que podem ser desenvolvidos, testados e implantados separadamente. Esta abordagem permite entregas incrementais de valor e facilita a manutenção evolutiva do sistema.

#### **Escalabilidade**

A arquitetura suporta crescimento horizontal e vertical, permitindo expansão da capacidade de processamento e armazenamento conforme o aumento da demanda operacional. O design considera desde o ambiente inicial limitado até a infraestrutura robusta do Estado de Mato Grosso.

#### **Segurança por Design**

Controles de segurança são implementados em todas as camadas da aplicação, desde a interface de usuário até o banco de dados. A arquitetura incorpora princípios de defesa em profundidade e menor privilégio necessário.

#### **Interoperabilidade**

O sistema é projetado para integrar-se facilmente com outros sistemas do CBM MT, utilizando padrões abertos e interfaces bem definidas que facilitam a troca de informações entre aplicações.

## **2.2 Arquitetura em Camadas**

### **2.2.1 Camada de Apresentação**

A camada de apresentação é responsável pela interface com o usuário, implementada através de aplicações web responsivas geradas pelo Scriptcase. Esta camada inclui formulários de entrada de dados, relatórios, dashboards e interfaces de consulta.

**Características Principais:**

- Interface web responsiva compatível com dispositivos móveis
- Navegação intuitiva baseada na hierarquia militar
- Controles de validação no lado cliente para melhor experiência do usuário
- Suporte a múltiplos navegadores (Chrome, Firefox, Safari, Edge)
- Implementação de Progressive Web App (PWA) para uso offline limitado

**Componentes da Interface:**

- Formulários de cadastro e movimentação de materiais
- Dashboards executivos com indicadores-chave de performance
- Relatórios parametrizáveis com exportação em múltiplos formatos
- Interface de consulta avançada com filtros dinâmicos
- Módulo de configuração de sistema para administradores

**2.2.2 Camada de Controle**

A camada de controle gerencia o fluxo de informações entre a apresentação e a lógica de negócio, implementando validações, controle de acesso e orquestração de processos. Esta camada utiliza os recursos nativos do Scriptcase para controle de sessão e segurança.

**Funcionalidades Principais:**

- Autenticação e autorização baseada em perfis hierárquicos
- Validação de dados de entrada e regras de negócio
- Controle de fluxo de trabalho para aprovações
- Gerenciamento de sessões e controle de concorrência
- Logging de auditoria para todas as operações

**Controles de Segurança:**

- Criptografia de senhas utilizando algoritmos seguros
- Controle de tentativas de acesso inválidas
- Timeout automático de sessões inativas
- Validação de entrada para prevenção de ataques de injeção
- Controle de acesso baseado em funções (RBAC)

**2.2.3 Camada de Lógica de Negócio**

Esta camada implementa todas as regras de negócio específicas do CBM MT, incluindo cálculos de estoque, validações de movimentação, controle de prazos e geração de alertas. A lógica é implementada através de bibliotecas PHP customizadas e procedures do PostgreSQL.

### **Módulos de Negócio:**

- Processamento de entradas e saídas de materiais
- Cálculo automático de pontos de reposição
- Controle de prazos de devolução e manutenção
- Geração de alertas e notificações
- Processamento de relatórios complexos

### **Regras Implementadas:**

- Validação de autorização hierárquica para movimentações
- Controle de limites de empréstimo por militar
- Cálculo automático de multas e penalidades
- Verificação de disponibilidade de estoque
- Controle de integridade patrimonial

#### **2.2.4 Camada de Dados**

A camada de dados é implementada em PostgreSQL, proporcionando robustez, confiabilidade e performance adequada para o volume de transações esperado. O banco de dados é estruturado seguindo princípios de normalização e otimização para consultas frequentes.

#### **Características do Banco de Dados:**

- Estrutura normalizada para evitar redundância de dados
- Índices otimizados para consultas de alta frequência
- Triggers para manutenção de integridade referencial
- Views materializadas para relatórios complexos
- Particionamento de tabelas históricas para melhor performance

#### **Estratégias de Backup:**

- Backup completo diário durante horário de menor movimento
- Backup incremental a cada 4 horas durante período operacional
- Replicação para servidor secundário (quando disponível)
- Testes mensais de recuperação de backup
- Retenção de backups por período mínimo de 2 anos

## **2.3 Modelo de Dados**

### **2.3.1 Entidades Principais**

#### **Materiais**

A entidade central do sistema, representando todos os itens controlados



pelo almoxarifado. Inclui informações detalhadas sobre especificações técnicas, classificação, valor e localização física.

Atributos principais:

- Código único de identificação (gerado automaticamente)
- Descrição detalhada e especificações técnicas
- Categoria e subcategoria de classificação
- Valor unitário e data de aquisição
- Fornecedor e número do contrato/processo
- Localização física no almoxarifado
- Status atual (disponível, emprestado, em manutenção, baixado)

### **Movimentações**

Registra todas as operações realizadas com materiais, mantendo histórico completo e auditável de todas as transações. Esta entidade é fundamental para rastreabilidade e controle patrimonial.

Atributos principais:

- Tipo de movimentação (entrada, saída, transferência, devolução)
- Data e hora da operação
- Material envolvido na movimentação
- Quantidade movimentada
- Responsável pela operação
- Autorizador da movimentação
- Observações e justificativas
- Status da movimentação (pendente, aprovada, cancelada)

### **Militares**

Integra informações dos militares responsáveis por materiais, empréstimos e autorizações. Esta entidade pode ser sincronizada com o sistema de RH existente.

Atributos principais:

- Número de identificação militar
- Nome completo e nome de guerra
- Patente e função atual
- Unidade de lotação
- Situação funcional
- Histórico de responsabilidades por materiais

### **Unidades**

Representa as diferentes unidades organizacionais do CBM MT que podem receber,

emprestar ou transferir materiais.

Atributos principais:

- Código da unidade
- Nome e sigla da unidade
- Endereço e localização geográfica
- Comandante responsável
- Tipo de unidade (quartel, destacamento, base operacional)

### **2.3.2 Relacionamentos Entre Entidades**

O modelo de dados implementa relacionamentos que garantem integridade referencial e permitem consultas eficientes para atendimento aos requisitos funcionais do sistema.

#### **Material-Movimentação (1:N)**

Cada material pode ter múltiplas movimentações ao longo do tempo, mantendo histórico completo de sua utilização. Este relacionamento é fundamental para rastreabilidade patrimonial.

#### **Militar-Movimentação (N:N)**

Militares podem estar envolvidos em movimentações como solicitantes, responsáveis ou autorizadores. O relacionamento permite identificar todas as responsabilidades de cada militar.

#### **Unidade-Material (1:N)**

Materiais estão sempre associados a uma unidade responsável por sua guarda, permitindo controle descentralizado e responsabilização adequada.

#### **Material-Manutenção (1:N)**

Equipamentos e viaturas possuem histórico de manutenções que deve ser controlado para garantir disponibilidade operacional e cumprimento de garantias.

## **2.4 Arquitetura de Segurança**

### **2.4.1 Controle de Acesso**

O sistema implementa controle de acesso baseado em perfis hierárquicos que respeitam a estrutura militar da instituição. Cada usuário possui um perfil que determina suas permissões de acesso às funcionalidades do sistema.

**Perfis de Usuário:**

### **Administrador do Sistema**

- Acesso completo a todas as funcionalidades
- Configuração de parâmetros do sistema
- Gestão de usuários e perfis
- Acesso a logs de auditoria
- Execução de rotinas de manutenção

### **Comandante de Unidade**

- Visualização de relatórios gerenciais da unidade
- Aprovação de movimentações de alto valor
- Acesso a indicadores de performance
- Autorização de baixas patrimoniais
- Configuração de parâmetros da unidade

### **Responsável pelo Almoxarifado**

- Registro de entradas e saídas de materiais
- Processamento de empréstimos e devoluções
- Execução de inventários
- Geração de relatórios operacionais
- Controle de localização física de materiais

### **Usuário Operacional**

- Consulta de disponibilidade de materiais
- Solicitação de empréstimos
- Visualização de materiais sob sua responsabilidade
- Acesso a relatórios básicos
- Atualização de dados pessoais

## **2.4.2 Auditoria e Logs**

O sistema mantém logs detalhados de todas as operações realizadas, garantindo rastreabilidade completa e atendimento aos requisitos de auditoria dos órgãos de controle.

### **Informações Registradas nos Logs:**

- Data e hora da operação
- Usuário responsável pela ação
- Tipo de operação realizada
- Dados alterados (antes e depois)
- Endereço IP de origem
- Resultado da operação (sucesso ou erro)

**Retenção e Proteção dos Logs:**

- Logs mantidos por período mínimo de 5 anos
- Armazenamento em tabelas protegidas contra alteração
- Backup automático dos logs de auditoria
- Acesso restrito a usuários autorizados
- Relatórios de auditoria para órgãos de controle

## **2.5 Arquitetura de Integração**

### **2.5.1 Integração com Sistemas Existentes**

O SIGA foi projetado para integrar-se harmoniosamente com os sistemas já existentes no CBM MT, aproveitando informações disponíveis e evitando duplicação de dados.

**Métodos de Integração:**

- APIs REST para troca de dados em tempo real
- Sincronização batch para grandes volumes de dados
- Triggers de banco de dados para atualizações automáticas
- Web services para consultas pontuais
- Arquivos de intercâmbio para sistemas legados

**Dados Compartilhados:**

- Informações de militares (RH)
- Dados orçamentários (Financeiro)
- Histórico de manutenções (Manutenção de Viaturas)
- Estrutura organizacional (Administração)
- Contratos e fornecedores (Compras)

### **2.5.2 Arquitetura de Comunicação**

A comunicação entre sistemas utiliza protocolos seguros e padrões abertos que garantem interoperabilidade e facilidade de manutenção.

**Protocolos Utilizados:**

- HTTPS para comunicação web segura
- SSL/TLS para criptografia de dados em trânsito
- JSON para formato de intercâmbio de dados
- XML para integração com sistemas legados
- SOAP para web services quando necessário

## **Controles de Segurança na Comunicação:**

- Autenticação mútua entre sistemas
- Criptografia de dados sensíveis
- Validação de integridade de mensagens
- Controle de rate limiting para APIs
- Monitoramento de tentativas de acesso inválidas

## **2.6 Design de Interface do Usuário**

### **2.6.1 Princípios de Design**

O design da interface do SIGA segue princípios de usabilidade e acessibilidade que garantem eficiência operacional e facilidade de uso, considerando o perfil dos usuários militares e a necessidade de operação em diferentes contextos, incluindo situações de emergência.

#### **Simplicidade e Clareza**

A interface prioriza a simplicidade visual e funcional, evitando elementos desnecessários que possam confundir ou retardar as operações. Cada tela possui objetivo claro e navegação intuitiva, permitindo que usuários com diferentes níveis de familiaridade tecnológica possam operar o sistema eficientemente.

#### **Consistência Visual**

Todos os elementos da interface seguem padrões visuais consistentes, incluindo cores, tipografia, iconografia e layout. Esta consistência reduz a curva de aprendizado e aumenta a produtividade dos usuários ao criar expectativas previsíveis sobre o comportamento do sistema.

#### **Responsividade e Adaptabilidade**

A interface adapta-se automaticamente a diferentes tamanhos de tela e dispositivos, mantendo funcionalidade completa em desktops, tablets e smartphones. Esta característica é fundamental para uso em campo durante operações de combate a incêndios.

#### **Acessibilidade**

O design incorpora princípios de acessibilidade universal, incluindo contraste adequado de cores, suporte a leitores de tela, navegação por teclado e textos alternativos para elementos visuais. Estas características garantem que o sistema possa ser utilizado por militares com diferentes necessidades.

### **2.6.2 Estrutura de Navegação**

## **Menu Principal Hierárquico**

A navegação principal reflete a estrutura organizacional militar, com acesso às funcionalidades organizadas por módulos funcionais. O menu adapta-se automaticamente ao perfil do usuário, exibindo apenas as opções para as quais possui autorização.

## **Breadcrumbs e Contexto**

Todas as telas incluem indicadores de localização (breadcrumbs) que permitem aos usuários compreender sua posição na estrutura do sistema e navegar facilmente para níveis superiores. O contexto atual é sempre claramente identificado.

## **Ações Rápidas**

Funcionalidades utilizadas frequentemente são acessíveis através de ações rápidas na tela principal, reduzindo o número de cliques necessários para operações rotineiras como consulta de estoque ou registro de movimentações.

### **2.6.3 Padrões de Interação**

#### **Formulários Inteligentes**

Os formulários de entrada de dados incluem validações em tempo real, preenchimento automático baseado em dados históricos e sugestões contextuais que aceleram a entrada de informações e reduzem erros.

#### **Confirmações e Feedback**

Todas as operações críticas requerem confirmação explícita do usuário, com descrição clara das consequências da ação. O sistema fornece feedback imediato sobre o resultado das operações, incluindo mensagens de sucesso, erro ou advertência.

#### **Busca e Filtros Avançados**

Interfaces de consulta incluem mecanismos de busca inteligente com sugestões automáticas e filtros avançados que permitem localização rápida de informações específicas em grandes volumes de dados.

## **2.7 Estratégia de Modularização**

### **2.7.1 Módulos Prioritários (Fase 1 - 50 dias)**

#### **Módulo de Controle de Entrada e Saída**

Este módulo fundamental implementa as funcionalidades básicas de registro de

entrada e saída de materiais, constituindo a base operacional do sistema. Inclui cadastro de materiais, registro de movimentações e controle básico de estoque.

Funcionalidades incluídas:

- Cadastro simplificado de materiais
- Registro de entrada de materiais
- Processamento de saídas com autorização
- Consulta básica de estoque
- Relatórios operacionais essenciais

### **Módulo de Inventário Básico**

Implementa funcionalidades essenciais para controle de inventário, permitindo cadastro detalhado de materiais e consultas de posição de estoque.

Funcionalidades incluídas:

- Cadastro completo de materiais com especificações
- Controle de localização física
- Consultas de estoque por categoria
- Alertas de estoque mínimo
- Relatório de posição de estoque

## **2.7.2 Módulos Secundários (Fase 2 - Expansão)**

### **Módulo de Gestão de Transferências**

Implementa controle completo de transferências entre unidades, incluindo transferências temporárias e definitivas com rastreabilidade completa.

### **Módulo de Concessão e Cautela**

Gerencia processos de concessão de materiais para uso pessoal e cautela de equipamentos, incluindo controle de prazos e responsabilidades.

### **Módulo de Logística Avançada**

funcionalidades avançadas de planejamento logístico, incluindo reserva de materiais para operações e otimização de distribuição.

## **2.7.3 Módulos de Apoio (Fase 3 - Consolidação)**

### **Módulo de Manutenção**

manutenções preventivas e corretivas de equipamentos e viaturas, incluindo agendamento automático e controle de custos.

## **Módulo de Relatórios Gerenciais**

Implementa relatórios avançados e dashboards executivos para apoio à tomada de decisão estratégica.

## **Módulo de Integração**

Desenvolve interfaces de integração com outros sistemas do CBM MT, incluindo RH, financeiro e manutenção de viaturas.

## **2.8 Tecnologias e Ferramentas**

### **2.8.1 Plataforma de Desenvolvimento**

#### **Scriptcase 9.4**

A escolha do Scriptcase como plataforma de desenvolvimento baseia-se na compatibilidade com a infraestrutura existente no CBM MT e na experiência da equipe de desenvolvimento. Esta plataforma oferece recursos adequados para desenvolvimento rápido de aplicações web robustas.

Vantagens da plataforma:

- Desenvolvimento rápido de aplicações (RAD)
- Geração automática de código otimizado
- Interface administrativa integrada
- Suporte nativo a múltiplos bancos de dados
- Recursos de segurança incorporados
- Facilidade de manutenção e evolução

#### **PostgreSQL**

O banco de dados PostgreSQL oferece robustez, confiabilidade e performance adequadas para o volume de transações esperado, além de recursos avançados de segurança e auditoria.

Características relevantes:

- Conformidade com padrões SQL
- Suporte a transações ACID
- Recursos avançados de indexação
- Replicação e backup automático
- Controles de segurança granulares
- Extensibilidade e customização

### **2.8.2 Infraestrutura de Desenvolvimento**

#### **Ambiente Inicial (50 dias)**



Durante a fase inicial, o desenvolvimento será realizado em servidor Scriptcase Host com 8GB de capacidade, proporcionando ambiente controlado para desenvolvimento e testes iniciais.

Especificações do ambiente:

- Servidor virtual com 8GB de RAM
- Armazenamento SSD para melhor performance
- Backup automático diário
- Acesso via HTTPS com certificado SSL
- Monitoramento básico de performance

### **Ambiente de Produção (Migração)**

Após aprovação do projeto piloto, o sistema será migrado para a infraestrutura oficial do Estado de Mato Grosso, oferecendo recursos ampliados e suporte técnico especializado.

Recursos disponíveis na infraestrutura estadual:

- Servidores dedicados com alta disponibilidade
- Licenciamento para múltiplos desenvolvedores
- Suporte a todas as tecnologias de desenvolvimento
- Backup e recuperação de desastres
- Monitoramento 24/7 e suporte técnico
- Conformidade com padrões de segurança governamentais

### **2.8.3 Ferramentas de Apoio**

#### **Controle de Versão**

Implementação de controle de versão utilizando Git para gerenciamento de código fonte, permitindo rastreabilidade de alterações e colaboração futura entre desenvolvedores.

#### **Documentação Técnica**

Utilização de ferramentas de documentação automática para manutenção de documentação técnica atualizada, incluindo dicionário de dados e manual de APIs.

#### **Testes Automatizados**

Implementação gradual de testes automatizados para garantir qualidade do código e facilitar manutenção evolutiva do sistema.

#### **Monitoramento e Logs**

Implementação de ferramentas de monitoramento de performance e análise de logs para identificação proativa de problemas e otimização contínua do sistema.

## **3. PLANEJAMENTO DE DESENVOLVIMENTO E GESTÃO DE RISCOS**

### **3.1 Metodologia de Desenvolvimento**

#### **3.1.1 Abordagem Ágil Adaptada**

O desenvolvimento do Sistema Integrado de Gestão de Almoxarifado seguirá uma metodologia ágil adaptada às características específicas do ambiente militar e às limitações de recursos identificadas. Esta abordagem combina princípios do Scrum com práticas de desenvolvimento incremental que permitem entregas de valor em ciclos curtos e validação contínua com os usuários finais.

A metodologia proposta reconhece as particularidades do contexto organizacional do CBM MT, incluindo a necessidade de aprovações hierárquicas, a importância da documentação formal e os requisitos rigorosos de segurança e auditoria. Desta forma, o processo de desenvolvimento incorpora checkpoints de aprovação e documentação detalhada sem comprometer a agilidade das entregas.

#### **Princípios Fundamentais da Metodologia:**

##### **Entregas Incrementais de Valor**

Cada ciclo de desenvolvimento produzirá funcionalidades completas e utilizáveis, permitindo que os usuários obtenham benefícios imediatos mesmo durante o processo de desenvolvimento. Esta abordagem reduz riscos de projeto e facilita a identificação precoce de ajustes necessários.

##### **Validação Contínua com Usuários**

Protótipos funcionais serão apresentados regularmente aos usuários finais para validação de requisitos e ajustes de interface. Esta prática garante que o sistema atenda efetivamente às necessidades operacionais e reduz retrabalho.

##### **Documentação Evolutiva**

A documentação técnica e funcional será desenvolvida paralelamente ao código, mantendo-se sempre atualizada e servindo como base para treinamento e manutenção futura do sistema.

##### **Qualidade Integrada**

Práticas de qualidade serão incorporadas em todas as fases do desenvolvimento, incluindo revisões de código, testes unitários e validações de

segurança, garantindo robustez e confiabilidade do sistema.

### **3.1.2 Estrutura de Ciclos de Desenvolvimento**

#### **Ciclo 1 (Dias 1-30): Fundação do Sistema**

O primeiro ciclo focará na implementação da infraestrutura básica do sistema e das funcionalidades fundamentais de controle de entrada e saída de materiais. Este ciclo estabelecerá as bases técnicas e funcionais para os desenvolvimentos subsequentes.

Objetivos específicos do Ciclo 1:

- Configuração do ambiente de desenvolvimento e banco de dados
- Implementação do sistema de autenticação e controle de acesso
- Desenvolvimento do cadastro básico de materiais
- Criação das funcionalidades de entrada de materiais
- Implementação do controle básico de saídas
- Desenvolvimento de consultas simples de estoque
- Criação de relatórios operacionais básicos

#### **Ciclo 2 (Dias 31- 60): Expansão Funcional**

O segundo ciclo expandirá as funcionalidades básicas, implementando controles mais sofisticados e funcionalidades de inventário avançado. Este ciclo consolidará o núcleo operacional do sistema.

Objetivos específicos do Ciclo 2:

- Implementação do controle de devoluções
- Desenvolvimento do sistema de alertas automáticos
- Criação de funcionalidades de inventário
- Implementação de controles de localização física
- Desenvolvimento de relatórios gerenciais básicos
- Criação de interfaces de consulta avançada
- Implementação de validações de negócio complexas

#### **Ciclo 3 (Dias 61- 90): Refinamento e Integração**

O terceiro ciclo focará no refinamento das funcionalidades existentes, implementação de integrações básicas e preparação para migração para ambiente de produção.

Objetivos específicos do Ciclo 3:

- Refinamento de interfaces baseado em feedback dos usuários
- Implementação de integrações com sistemas existentes
- Desenvolvimento de funcionalidades de auditoria

- Criação de documentação de usuário
- Testes de performance e segurança
- Preparação para migração para ambiente de produção
- Treinamento inicial de usuários-chave

### **3.1.3 Práticas de Desenvolvimento**

#### **Desenvolvimento Orientado por Testes**

Embora o Scriptcase não ofereça suporte nativo a frameworks de teste automatizado, serão implementadas práticas de validação sistemática que incluem testes manuais estruturados, validação de dados e verificação de regras de negócio.

#### **Revisão de Código**

Todo código desenvolvido passará por revisão sistemática para garantir qualidade, aderência aos padrões estabelecidos e identificação precoce de possíveis problemas de segurança ou performance.

#### **Integração Contínua**

Será estabelecido processo de integração contínua que inclui backup automático do código fonte, versionamento de alterações e deploy controlado para ambiente de testes.

#### **Documentação Técnica Contínua**

A documentação técnica será mantida atualizada durante todo o processo de desenvolvimento, incluindo dicionário de dados, especificações de interfaces e manuais de instalação e configuração.

## **3.2 Cronograma Detalhado de Desenvolvimento**

### **3.2.1 Cronograma do Ciclo 1 (Dias 1-30)**

#### **Dias 1-6: Preparação do Ambiente**

- Configuração do servidor de desenvolvimento
- Instalação e configuração do Scriptcase 9.4
- Criação da estrutura inicial do banco de dados PostgreSQL
- Configuração de backup automático
- Estabelecimento de controle de versão
- Criação da documentação de ambiente

#### **Dias 7-13: Infraestrutura de Segurança**

- Implementação do sistema de autenticação

- Criação dos perfis de usuário hierárquicos
- Desenvolvimento do controle de acesso baseado em funções
- Implementação de logs de auditoria
- Configuração de criptografia de senhas
- Testes de segurança básicos

#### **Dias 14-19: Cadastro de Materiais**

- Desenvolvimento da interface de cadastro de materiais
- Implementação de validações de entrada
- Criação de categorização automática
- Desenvolvimento de consultas básicas
- Implementação de geração de códigos únicos
- Testes de funcionalidade

#### **Dias 20-25: Controle de Entradas**

- Desenvolvimento da interface de registro de entradas
- Implementação de controle de fornecedores
- Criação de vinculação com contratos/processos
- Desenvolvimento de atualização automática de estoque
- Implementação de geração de comprovantes
- Testes de integração

#### **Dias 26-30: Controle de Saídas**

- Desenvolvimento da interface de registro de saídas
- Implementação de fluxo de aprovação
- Criação de controle de autorização hierárquica
- Desenvolvimento de geração de termos de responsabilidade
- Implementação de atualização de estoque
- Testes finais do ciclo

### **3.2.2 Cronograma do Ciclo 2 (Dias 31-60)**

#### **Dias 31- 37: Controle de Devoluções**

- Desenvolvimento da interface de processamento de devoluções
- Implementação de controle de prazos
- Criação de registro de estado de conservação
- Desenvolvimento de cálculo de multas/penalidades
- Implementação de alertas de atraso
- Testes de funcionalidade

#### **Dias 38-44: Sistema de Alertas**

- Desenvolvimento de alertas de estoque mínimo
- Implementação de notificações de vencimento
- Criação de alertas de manutenção
- Desenvolvimento de dashboard de alertas
- Implementação de configuração de parâmetros
- Testes de notificações

#### **Dias 45-49: Funcionalidades de Inventário**

- Desenvolvimento de interface de inventário
- Implementação de geração de listas de contagem
- Criação de registro de divergências
- Desenvolvimento de ajustes de estoque
- Implementação de relatórios de inventário
- Testes de processo completo

#### **Dias 50-55: Controle de Localização**

- Desenvolvimento de cadastro de localizações físicas
- Implementação de controle de movimentação interna
- Criação de consultas por localização
- Desenvolvimento de otimização de layout
- Implementação de relatórios de localização
- Testes de rastreabilidade

#### **Dias 56-60: Relatórios Gerenciais**

- Desenvolvimento de relatórios de movimentação
- Implementação de análises de consumo
- Criação de indicadores de performance
- Desenvolvimento de exportação para Excel/PDF
- Implementação de agendamento de relatórios
- Testes finais do ciclo

### **3.2.3 Cronograma do Ciclo 3 (Dias 61-90)**

#### **Dias 61-67: Refinamento de Interfaces**

- Ajustes baseados em feedback dos usuários
- Otimização de performance de telas
- Melhoria de usabilidade
- Implementação de atalhos e funcionalidades rápidas
- Ajustes de responsividade para dispositivos móveis
- Testes de usabilidade

### **Dias 68-74: Integrações Básicas**

- Desenvolvimento de integração com sistema de RH
- Implementação de sincronização de dados de militares
- Criação de interfaces de comunicação
- Desenvolvimento de rotinas de sincronização
- Implementação de tratamento de erros
- Testes de integração

### **Dias 75-81: Auditoria e Segurança**

- Implementação de trilhas de auditoria completas
- Desenvolvimento de relatórios de auditoria
- Criação de controles de integridade de dados
- Implementação de backup e recuperação
- Testes de segurança avançados
- Validação de conformidade

### **Dias 82-86: Documentação e Treinamento**

- Criação de manual do usuário
- Desenvolvimento de material de treinamento
- Implementação de help online
- Criação de vídeos tutoriais básicos
- Documentação de procedimentos operacionais
- Preparação de material para apresentação

### **Dias 87-90: Preparação para Produção**

- Testes de performance e carga
- Preparação para migração
- Criação de scripts de instalação
- Documentação de configuração de produção
- Treinamento de usuários-chave
- Apresentação final para comando

## **3.3 Análise de Riscos do Projeto**

### **3.3.1 Identificação de Riscos**

#### **Riscos Técnicos**

#### **RT001 - Limitações da Plataforma Scriptcase**

Probabilidade: Média | Impacto: Alto

O Scriptcase, embora adequado para desenvolvimento rápido, pode apresentar

limitações para implementação de funcionalidades complexas ou integrações avançadas. Estas limitações podem impactar o escopo funcional do sistema ou exigir soluções alternativas que aumentem a complexidade do desenvolvimento.

#### **RT002 - Performance do Ambiente de Desenvolvimento**

Probabilidade: Alta | Impacto: Médio

O servidor de desenvolvimento com 8GB pode apresentar limitações de performance durante testes com volumes maiores de dados ou múltiplos usuários simultâneos, impactando a validação adequada do sistema antes da migração para produção.

#### **RT003 - Complexidade de Integração com Sistemas Existentes**

Probabilidade: Média | Impacto: Alto

A integração com sistemas legados do CBM MT pode apresentar complexidades técnicas não previstas, incluindo incompatibilidades de formato de dados, protocolos de comunicação ou limitações de acesso.

#### **RT004 - Perda de Dados Durante Desenvolvimento**

Probabilidade: Baixa | Impacto: Alto

Falhas no servidor de desenvolvimento ou problemas de backup podem resultar em perda de código fonte ou dados de teste, causando atrasos significativos no cronograma.

### **Riscos Organizacionais**

#### **RO001 - Mudanças de Requisitos Durante Desenvolvimento**

Probabilidade: Alta | Impacto: Médio

A descoberta de novos requisitos ou mudanças nas necessidades operacionais durante o desenvolvimento pode exigir alterações significativas no escopo ou arquitetura do sistema.

#### **RO002 - Disponibilidade Limitada de Usuários para Validação**

Probabilidade: Média | Impacto: Médio

A agenda operacional intensa do BEA, especialmente durante o período de combate a incêndios, pode limitar a disponibilidade de usuários-chave para validação de funcionalidades e feedback sobre o sistema.

#### **RO003 - Resistência à Mudança de Processos**

Probabilidade: Média | Impacto: Alto

Militares acostumados com processos manuais podem apresentar resistência à adoção do novo sistema, impactando a efetividade da implementação e exigindo esforços adicionais de treinamento e mudança cultural.



## **RO004 - Aprovações Hierárquicas Demoradas**

Probabilidade: Média | Impacto: Médio

O processo de aprovação hierárquica para decisões técnicas importantes pode causar atrasos no cronograma de desenvolvimento, especialmente se houver necessidade de ajustes significativos.

## **Riscos de Recursos**

### **RR001 - Sobrecarga do Desenvolvedor Único**

Probabilidade: Alta | Impacto: Alto

A dependência de um único desenvolvedor cria risco significativo de atraso ou interrupção do projeto em caso de problemas pessoais, doença ou sobrecarga de trabalho.

### **RR002 - Limitações Orçamentárias para Ferramentas Adicionais**

Probabilidade: Média | Impacto: Médio

A ausência de orçamento específico pode limitar a aquisição de ferramentas ou serviços que poderiam acelerar o desenvolvimento ou melhorar a qualidade do sistema.

### **RR003 - Capacidade Limitada do Servidor de Desenvolvimento**

Probabilidade: Alta | Impacto: Médio

As limitações de capacidade do servidor podem impactar testes de performance e validação com volumes realistas de dados, comprometendo a qualidade da validação antes da migração.

## **Riscos Externos**

### **RE001 - Mudanças na Infraestrutura do Estado**

Probabilidade: Baixa | Impacto: Alto

Mudanças na infraestrutura tecnológica do Estado de Mato Grosso podem impactar os planos de migração e exigir adaptações significativas na arquitetura do sistema.

### **RE002 - Alterações Regulamentares**

Probabilidade: Baixa | Impacto: Médio

Mudanças na legislação aplicável à gestão patrimonial pública podem exigir ajustes nos requisitos funcionais e regras de negócio do sistema.

### **RE003 - Problemas de Conectividade**

Probabilidade: Média | Impacto: Baixo

Instabilidades na conectividade de internet podem impactar o

desenvolvimento em ambiente cloud e a comunicação com usuários para validação.

### **3.3.2 Estratégias de Mitigação**

#### **Mitigação de Riscos Técnicos**

##### **Para RT001 - Limitações da Plataforma Scriptcase**

- Realizar prova de conceito para funcionalidades complexas antes da implementação completa
- Manter contato com suporte técnico do Scriptcase para orientações especializadas
- Desenvolver soluções alternativas utilizando PHP customizado quando necessário
- Documentar limitações identificadas para consideração em futuras expansões

##### **Para RT002 - Performance do Ambiente de Desenvolvimento**

- Implementar testes de performance com dados reduzidos mas representativos
- Utilizar ferramentas de monitoramento para identificar gargalos
- Planejar migração antecipada para ambiente de produção se necessário
- Otimizar consultas e índices de banco de dados desde o início

##### **Para RT003 - Complexidade de Integração**

- Realizar análise detalhada dos sistemas existentes antes da implementação
- Desenvolver interfaces de integração em fases incrementais
- Manter comunicação constante com responsáveis pelos sistemas legados
- Criar planos de contingência para integração manual temporária

##### **Para RT004 - Perda de Dados**

- Implementar backup automático diário do código fonte e banco de dados
- Utilizar controle de versão distribuído (Git) com repositório remoto
- Manter cópias de segurança em múltiplas localizações
- Testar procedimentos de recuperação regularmente

#### **Mitigação de Riscos Organizacionais**

##### **Para RO001 - Mudanças de Requisitos**

- Implementar processo formal de controle de mudanças
- Priorizar requisitos críticos para implementação inicial
- Manter flexibilidade na arquitetura para acomodar mudanças
- Documentar impactos de mudanças no cronograma e recursos

##### **Para RO002 - Disponibilidade de Usuários**

- Agendar sessões de validação com antecedência
- Utilizar protótipos funcionais para validação assíncrona

- Criar grupos de usuários representativos para feedback
- Implementar canais de comunicação eficientes para coleta de feedback

#### **Para RO003 - Resistência à Mudança**

- Envolver usuários-chave no processo de desenvolvimento
- Demonstrar benefícios tangíveis do novo sistema
- Implementar treinamento gradual e suporte contínuo
- Criar embaixadores do sistema entre os usuários

#### **Para RO004 - Aprovações Hierárquicas**

- Preparar documentação detalhada para facilitar aprovações
- Agendar reuniões de aprovação com antecedência
- Implementar processo de aprovação em etapas
- Manter comunicação proativa sobre necessidades de aprovação

### **Mitigação de Riscos de Recursos**

#### **Para RR001 - Sobrecarga do Desenvolvedor**

- Implementar documentação detalhada para facilitar transferência de conhecimento
- Criar backup de conhecimento através de documentação técnica
- Estabelecer contatos com desenvolvedores Scriptcase externos para emergências
- Planejar carga de trabalho realista com margens de segurança

#### **Para RR002 - Limitações Orçamentárias**

- Priorizar ferramentas gratuitas e open source
- Utilizar recursos disponíveis na infraestrutura existente
- Negociar licenças temporárias ou de avaliação quando necessário
- Documentar necessidades futuras para planejamento orçamentário

#### **Para RR003 - Capacidade do Servidor**

- Otimizar código e consultas para melhor performance
- Implementar cache de dados quando apropriado
- Utilizar técnicas de paginação para grandes volumes de dados
- Planejar migração antecipada se limitações impactarem desenvolvimento

## **3.4 Estimativas de Recursos e Custos**

### **3.4.1 Recursos Humanos**

#### **Desenvolvedor Principal (90 dias)**

- Análise e especificação de requisitos: 60 horas
- Desenvolvimento de código: 440 horas
- Testes e validação: 120 horas

- Documentação: 60 horas
- Treinamento e suporte: 40 horas
- Total: 720 horas

#### **Usuários para Validação (Estimativa)**

- Responsável pelo almoxarifado: 30 horas
- Comandante da unidade: 12 horas
- Usuários operacionais: 24 horas (4 usuários x 4 horas)
- Total: 66 horas de envolvimento de usuários

### **3.4.2 Recursos Tecnológicos**

#### **Infraestrutura de Desenvolvimento (90 dias)**

- Servidor Scriptcase Host (8GB): **Custo pessoal do desenvolvedor**
- Licença Scriptcase 9.4: **Custo pessoal do desenvolvedor**
- Backup e segurança: **Incluído no servidor**
- Monitoramento básico: **Incluído no servidor**

#### **Ferramentas de Apoio**

- Controle de versão (Git): **Gratuito**
- Documentação (Markdown): **Gratuito**
- Testes manuais: **Sem custo adicional**
- Comunicação (email, reuniões): **Sem custo adicional**

### **3.4.3 Custos de Migração para Produção**

#### **Infraestrutura do Estado de MT (Estimativa)**

- Servidor de produção: Sem custo adicional (infraestrutura existente)
- Licenças adicionais Scriptcase: Conforme necessidade futura
- Backup e recuperação: Incluído na infraestrutura estadual
- Suporte técnico: Incluído na infraestrutura estadual

#### **Treinamento e Implantação**

- Material de treinamento: Desenvolvido internamente
- Sessões de treinamento: Recursos internos do CBM MT
- Suporte inicial: Desenvolvedor principal
- Documentação de usuário: Incluída no desenvolvimento

## **3.5 Métricas de Qualidade e Sucesso**

### **3.5.1 Indicadores de Performance Técnica**

### **Tempo de Resposta**

- Consultas simples: < 3 segundos
- Relatórios complexos: < 10 segundos
- Operações de entrada/saída: < 5 segundos
- Login e autenticação: < 2 segundos

### **Disponibilidade do Sistema**

- Horário comercial: > 99%
- Período operacional crítico: > 95%
- Tempo de recuperação após falha: < 30 minutos
- Backup bem-sucedido: 100% dos dias

### **Capacidade e Escalabilidade**

- Usuários simultâneos suportados: 50
- Registros de movimentação: > 100.000
- Tempo de backup completo: < 2 horas
- Crescimento anual de dados: Suporte a 20%

## **3.5.2 Indicadores de Qualidade Funcional**

### **Precisão dos Dados**

- Divergências de inventário: < 2%
- Erros de entrada de dados: < 1%
- Inconsistências de estoque: 0%
- Falhas de integração: < 0,5%

### **Usabilidade**

- Tempo de treinamento para novos usuários: < 4 horas
- Taxa de erro de usuário: < 5%
- Satisfação do usuário: > 80%
- Tempo médio para completar operações rotineiras: Redução de 50%

### **Conformidade e Auditoria**

- Completude dos logs de auditoria: 100%
- Conformidade com regulamentos: 100%
- Rastreabilidade de movimentações: 100%
- Integridade de dados históricos: 100%

## **3.5.3 Indicadores de Sucesso do Projeto**

### **Entrega no Prazo**

- Cumprimento do cronograma de 30 dias: 100%
- Entregas incrementais conforme planejado: 100%
- Marcos de aprovação atendidos: 100%
- Documentação entregue no prazo: 100%

### **Qualidade das Entregas**

- Funcionalidades implementadas conforme especificação: > 95%
- Bugs críticos identificados: 0
- Aprovação dos usuários-chave: > 90%
- Aprovação do comando: Necessária para continuidade

### **Benefícios Operacionais (expectativa)**

- Redução do tempo de processamento de movimentações: > 60%
- Melhoria na precisão do controle de estoque: > 80%
- Redução de perdas e extravios: > 50%
- Melhoria na geração de relatórios: > 90%

## **3.6 Plano de Contingência**

### **3.6.1 Cenários de Contingência**

#### **Cenário 1: Atraso Significativo no Desenvolvimento**

Se o desenvolvimento apresentar atraso superior a 5 dias em relação ao cronograma planejado:

- Priorizar funcionalidades críticas para entrega mínima viável
- Reduzir escopo de funcionalidades secundárias
- Solicitar extensão de prazo com justificativa técnica
- Implementar trabalho em horários estendidos se necessário

#### **Cenário 2: Problemas Técnicos Graves**

Em caso de problemas técnicos que impeçam o desenvolvimento normal:

- Ativar plano de backup e recuperação de dados
- Contactar suporte técnico especializado do Scriptcase
- Implementar soluções alternativas temporárias
- Documentar problemas para resolução futura

#### **Cenário 3: Indisponibilidade do Desenvolvedor Principal**

Se o desenvolvedor principal ficar indisponível por período superior a 3 dias:

- Ativar documentação técnica para transferência de conhecimento
- Contactar desenvolvedores Scriptcase externos para continuidade
- Priorizar funcionalidades já em desenvolvimento
- Replanejar cronograma conforme nova disponibilidade

## **Cenário 4: Mudanças Críticas de Requisitos**

Se mudanças significativas de requisitos forem solicitadas:

- Avaliar impacto no cronograma e recursos
- Repriorizar funcionalidades conforme nova demanda
- Solicitar aprovação formal para mudanças de escopo
- Documentar impactos para futuras fases do projeto

### **3.6.2 Procedimentos de Escalação**

#### **Nível 1: Problemas Técnicos Menores**

- Responsável: Desenvolvedor principal
- Tempo de resolução: 24 horas
- Ações: Análise técnica, implementação de correções, testes

#### **Nível 2: Problemas Técnicos Significativos**

- Responsável: Desenvolvedor + Suporte Scriptcase
- Tempo de resolução: 72 horas
- Ações: Análise especializada, soluções alternativas, escalação para fornecedor

#### **Nível 3: Problemas Críticos de Projeto**

- Responsável: Comando do BEA + Desenvolvedor
- Tempo de resolução: 5 dias úteis
- Ações: Revisão de escopo, realocação de recursos, decisões estratégicas

#### **Nível 4: Problemas de Continuidade do Projeto**

- Responsável: Comando Geral do CBM MT
- Tempo de resolução: Conforme decisão superior
- Ações: Avaliação de viabilidade, decisão de continuidade, planejamento alternativo

## **4. DOCUMENTAÇÃO TÉCNICA E MANUAIS**

### **4.1 Manual de Instalação e Configuração**

#### **4.1.1 Requisitos de Sistema**

##### **Requisitos de Hardware Mínimos**

O Sistema Integrado de Gestão de Almoxarifado foi projetado para operar eficientemente em infraestruturas variadas, desde o ambiente de desenvolvimento inicial até a infraestrutura robusta do Estado de Mato Grosso. Os requisitos mínimos garantem operação adequada mesmo em condições limitadas de recursos.

Para o servidor de aplicação, recomenda-se processador com arquitetura x86\_64 de pelo menos 2 núcleos operando a 2.4 GHz ou superior. A memória RAM mínima necessária é de 4GB, embora se recomende 8GB para operação confortável com múltiplos usuários simultâneos. O armazenamento deve dispor de pelo menos 50GB de espaço livre em disco SSD para garantir performance adequada das operações de banco de dados.

Para o servidor de banco de dados PostgreSQL, quando operando em servidor dedicado, os requisitos incluem processador de pelo menos 4 núcleos, 8GB de RAM e 100GB de armazenamento SSD. Em configurações de alta disponibilidade, recomenda-se duplicar estes recursos para permitir replicação e backup adequados.

### **Requisitos de Software**

O sistema operacional suportado é Linux, preferencialmente distribuições baseadas em Debian ou Red Hat, com versões atualizadas e suporte de segurança ativo. O Scriptcase

9.4 requer Apache HTTP Server versão 2.4 ou superior, PHP versão 7.4 ou 8.0, e PostgreSQL versão 12 ou superior.

Para o ambiente de desenvolvimento, é necessário acesso à internet para download de dependências e atualizações. O servidor deve dispor de certificado SSL válido para garantir comunicação segura, especialmente importante no ambiente militar onde a segurança da informação é prioritária.

### **Requisitos de Rede**

A conectividade de rede deve garantir largura de banda mínima de 10 Mbps para operação adequada com múltiplos usuários. Para integração com sistemas existentes do CBMMT, é necessário acesso à rede interna da corporação com as devidas autorizações de firewall.

#### **4.1.2 Procedimentos de Instalação**

##### **Preparação do Ambiente**

O processo de instalação inicia-se com a preparação do ambiente de servidor, incluindo atualização do sistema operacional, configuração de usuários administrativos e instalação de dependências básicas. É fundamental estabelecer políticas de segurança adequadas desde o início, incluindo configuração de firewall, desabilitação de serviços desnecessários e implementação de políticas de senha robustas.

A configuração do Apache HTTP Server deve incluir módulos necessários para operação do PHP e SSL, com configuração de virtual hosts específicos para o SIGA. A



configuração de SSL deve utilizar certificados válidos e protocolos seguros, desabilitando versões obsoletas que possam comprometer a segurança.

### **Instalação do PostgreSQL**

A instalação do PostgreSQL deve seguir as melhores práticas de segurança, incluindo configuração de autenticação adequada, criação de usuários específicos para a aplicação e configuração de backup automático. O banco de dados deve ser configurado com encoding UTF-8 para suporte adequado a caracteres especiais e acentuação.

A configuração de performance do PostgreSQL deve ser ajustada conforme os recursos disponíveis, incluindo otimização de memória compartilhada, cache de buffer e configurações de checkpoint. Para ambientes de produção, é essencial configurar log de transações (WAL) adequado para permitir recuperação point-in-time em caso de falhas.

### **Instalação do Scriptcase**

A instalação do Scriptcase 9.4 deve seguir os procedimentos padrão do fabricante, incluindo verificação de licença, configuração de conexão com banco de dados e teste de funcionalidades básicas. É importante configurar adequadamente os diretórios de trabalho, cache e upload de arquivos, garantindo permissões apropriadas para o usuário do servidor web.

A configuração inicial deve incluir criação de usuário administrativo, configuração de parâmetros de segurança e teste de conectividade com o banco de dados PostgreSQL. Todos os módulos necessários do PHP devem ser verificados e instalados, incluindo extensões para PostgreSQL, criptografia e manipulação de arquivos.

### **Implantação da Aplicação SIGA**

A implantação da aplicação envolve importação dos arquivos de projeto do Scriptcase, criação da estrutura de banco de dados através dos scripts SQL fornecidos e configuração de parâmetros específicos da aplicação. É fundamental executar os scripts de criação de tabelas na ordem correta para manter integridade referencial.

Após a criação da estrutura básica, devem ser executados os scripts de carga inicial de dados, incluindo cadastro de usuários administrativos, configuração de parâmetros do sistema e criação de dados de referência como categorias de materiais e unidades organizacionais.

#### **4.1.3 Configuração de Segurança**

##### **Controle de Acesso ao Sistema**

A configuração de segurança inicia-se com a definição adequada de perfis de usuário que respeitem a hierarquia militar da instituição. Cada perfil deve ter permissões específicas que seguem o princípio do menor privilégio necessário, garantindo que usuários tenham acesso apenas às funcionalidades essenciais para suas funções.

A autenticação deve ser configurada com políticas de senha robustas, incluindo comprimento mínimo, complexidade de caracteres e expiração periódica. Para ambientes de alta segurança, recomenda-se implementação de autenticação de dois fatores utilizando tokens ou aplicativos móveis.

### **Configuração de Auditoria**

O sistema de auditoria deve ser configurado para registrar todas as operações críticas, incluindo login de usuários, alterações de dados, aprovações de movimentações e acesso a relatórios confidenciais. Os logs de auditoria devem ser armazenados em tabelas protegidas contra alteração e com backup automático.

A configuração deve incluir definição de políticas de retenção de logs, procedimentos de análise periódica e alertas automáticos para atividades suspeitas. É fundamental estabelecer procedimentos para fornecimento de relatórios de auditoria aos órgãos de controle interno e externo.

### **Backup e Recuperação**

A estratégia de backup deve incluir backup completo diário durante horários de menor movimento, backup incremental a cada 4 horas durante período operacional e replicação para servidor secundário quando disponível. Os backups devem ser testados mensalmente para garantir integridade e capacidade de recuperação.

Os procedimentos de recuperação devem ser documentados detalhadamente, incluindo cenários de falha parcial e total, tempos de recuperação esperados e responsabilidades de cada membro da equipe técnica. É essencial manter cópias de backup em localização física separada para proteção contra desastres.

## **4.2 Documentação de APIs e Interfaces**

### **4.2.1 Interfaces de Integração**

#### **API de Sincronização com Sistema de RH**

A interface de integração com o sistema de Recursos Humanos permite sincronização automática de informações de militares, incluindo dados pessoais, patente, unidade de lotação e situação funcional. Esta integração é fundamental para manter consistência de dados e evitar duplicação de informações entre sistemas.

A API utiliza protocolo REST com autenticação baseada em tokens JWT, garantindo segurança na comunicação entre sistemas. Os dados são trocados em formato JSON, facilitando parsing e validação. A sincronização pode ser configurada para execução automática em intervalos definidos ou acionada manualmente quando necessário.

Endpoints principais da API:

- GET /api/militares: Recupera lista de militares ativos
- GET /api/militares/{id}: Recupera dados específicos de um militar
- POST /api/militares/sync: Sincroniza dados alterados
- GET /api/unidades: Recupera estrutura organizacional atualizada

### **API de Integração Financeira**

A integração com o sistema financeiro permite verificação de disponibilidade orçamentária para aquisições, controle de empenhos relacionados a materiais e consolidação de informações para prestação de contas. Esta interface é essencial para garantir conformidade com a legislação de gestão financeira pública.

A comunicação utiliza web services SOAP para compatibilidade com sistemas legados, incluindo validação de esquemas XML e controle de versão de mensagens. Todas as transações são registradas em log específico para auditoria e rastreabilidade.

Principais operações disponíveis:

- ConsultarDisponibilidadeOrcamentaria: Verifica saldo disponível para aquisições
- RegistrarEmpenho: Registra empenho relacionado a material adquirido
- ConsultarHistoricoEmpenhos: Recupera histórico de empenhos por período
- GerarRelatorioConsolidado: Produz relatório para prestação de contas

### **API de Manutenção de Viaturas**

A integração com o sistema de manutenção de viaturas permite compartilhamento de informações sobre quilometragem, histórico de manutenções e agendamento de serviços preventivos. Esta integração otimiza o controle de disponibilidade operacional dos veículos.

A interface utiliza mensageria assíncrona para notificações de eventos importantes, como vencimento de manutenções ou identificação de avarias. O protocolo de comunicação garante entrega confiável de mensagens mesmo em caso de indisponibilidade temporária de um dos sistemas.

Funcionalidades de integração:

- Sincronização de dados de viaturas e equipamentos

- Notificação de vencimentos de manutenção
- Registro de uso operacional para cálculo de intervalos
- Compartilhamento de histórico de avarias e reparos

#### 4.2.2 Especificação de Banco de Dados

##### Estrutura de Tabelas Principais

A estrutura do banco de dados foi projetada seguindo princípios de normalização que garantem integridade referencial e otimização de consultas. As tabelas principais incluem materiais, movimentações, militares, unidades e configurações do sistema.

A tabela de materiais (tb\_materiais) constitui o núcleo central do sistema, armazenando informações detalhadas sobre todos os itens controlados pelo almoxarifado. Inclui campos para código único, descrição, especificações técnicas, categoria, valor, fornecedor, localização física e status atual.

```
CREATE TABLE tb_materiais (
  id_material SERIAL PRIMARY KEY,
  codigo_material VARCHAR(20) UNIQUE NOT NULL,
  descricao TEXT NOT NULL,
  especificacoes_tecnicas TEXT,
  categoria_id INTEGER REFERENCES tb_categorias(id_categoria),
  valor_unitario DECIMAL(10,2),
  fornecedor_id INTEGER REFERENCES tb_fornecedores(id_fornecedor),
  localizacao_fisica VARCHAR(100),
  status_material VARCHAR(20) DEFAULT 'DISPONIVEL',
  data_cadastro TIMESTAMP DEFAULT CURRENT_TIMESTAMP,
  usuario_cadastro INTEGER REFERENCES tb_usuarios(id_usuario)
);
```

```
CREATE TABLE tb_movimentacoes (
  id_movimentacao SERIAL PRIMARY KEY,
  tipo_movimentacao VARCHAR(20) NOT NULL,
  material_id INTEGER REFERENCES tb_materiais(id_material),
```

A tabela de movimentações (tb\_movimentacoes) registra todas as operações realizadas com materiais, mantendo histórico completo e auditável. Inclui tipo de movimentação, datas, quantidades, responsáveis e observações detalhadas.

```

quantidade INTEGER NOT NULL,
data_movimentacao TIMESTAMP DEFAULT CURRENT_TIMESTAMP,
militar_responsavel INTEGER REFERENCES tb_militares(id_militar),
militar_autorizador INTEGER REFERENCES tb_militares(id_militar),
unidade_origem INTEGER REFERENCES tb_unidades(id_unidade),
unidade_destino INTEGER REFERENCES tb_unidades(id_unidade),
observacoes TEXT,
status_movimentacao VARCHAR(20) DEFAULT 'PENDENTE'
);

```

## Índices e Otimizações

A performance do sistema é garantida através de índices estrategicamente posicionados nas colunas mais utilizadas em consultas. Índices compostos são criados para consultas que envolvem múltiplas condições, otimizando operações de busca e relatórios.

```

CREATE INDEX idx_materiais_categoria ON tb_materiais(categoria_id);
CREATE INDEX idx_materiais_status ON tb_materiais(status_material);
CREATE INDEX idx_movimentacoes_data ON
tb_movimentacoes(data_movimentacao);
CREATE INDEX idx_movimentacoes_material ON tb_movimentacoes(material_id);
CREATE INDEX idx_movimentacoes_tipo_data ON
tb_movimentacoes(tipo_movimentacao, data_movimentacao);

```

## Triggers e Procedimentos

Triggers são implementados para manutenção automática de integridade de dados e execução de regras de negócio complexas. Incluem atualização automática de estoque, validação de movimentações e manutenção de logs de auditoria.

```

CREATE OR REPLACE FUNCTION atualizar_estoque()
RETURNS TRIGGER AS $$
BEGIN
  IF NEW.tipo_movimentacao = 'ENTRADA' THEN
    UPDATE tb_materiais
    SET quantidade_estoque = quantidade_estoque + NEW.quantidade
    WHERE id_material = NEW.material_id;
  ELSIF NEW.tipo_movimentacao = 'SAIDA' THEN
    UPDATE tb_materiais
    SET quantidade_estoque = quantidade_estoque - NEW.quantidade
    WHERE id_material = NEW.material_id;
  END IF;
  RETURN NEW;
END;
$$ LANGUAGE plpgsql;

```

```
CREATE TRIGGER trigger_atualizar_estoque  
AFTER INSERT ON tb_movimentacoes  
FOR EACH ROW  
EXECUTE FUNCTION atualizar_estoque();
```

## 4.3 Manual do Usuário

### 4.3.1 Guia de Primeiros Passos

#### Acesso ao Sistema

O acesso ao Sistema Integrado de Gestão de Almoxarifado é realizado através de navegador web, utilizando credenciais fornecidas pelo administrador do sistema. A tela de login apresenta campos para usuário e senha, além de opções para recuperação de senha em caso de esquecimento.

Após o primeiro acesso, o sistema solicitará alteração da senha padrão, seguindo as políticas de segurança estabelecidas. É importante escolher senha robusta que combine letras maiúsculas e minúsculas, números e símbolos especiais, com comprimento mínimo de 8 caracteres.

A interface principal apresenta menu de navegação organizado por módulos funcionais, adaptado automaticamente ao perfil do usuário. Usuários com perfil de almoxarife terão acesso a funcionalidades de movimentação e controle, enquanto gestores visualizarão relatórios e indicadores gerenciais.

#### Navegação Básica

A navegação no sistema segue padrões intuitivos, com menu principal localizado na parte superior da tela e submenu contextual na lateral esquerda. Breadcrumbs na parte superior indicam a localização atual dentro da estrutura do sistema, facilitando a orientação e navegação.

Ícones padronizados identificam diferentes tipos de operações: cadastro (ícone de mais), consulta (ícone de lupa), edição (ícone de lápis) e exclusão (ícone de lixeira). Cores são utilizadas consistentemente para indicar status: verde para aprovado, amarelo para pendente, vermelho para rejeitado ou em atraso.

#### Funcionalidades Básicas

As operações mais comuns incluem cadastro de novos materiais, registro de entradas e saídas, consulta de estoque e geração de relatórios básicos. Cada funcionalidade possui assistente que orienta o usuário através dos passos necessários, incluindo validações em tempo real e mensagens de ajuda contextual.

Formulários incluem campos obrigatórios claramente identificados, validação automática de dados e sugestões baseadas em histórico de uso. Botões de ações são

posicionados consistentemente, com confirmação obrigatória para operações críticas como exclusão de registros ou aprovação de movimentações de alto valor.

#### **4.3.2 Operações Rotineiras**

##### **Cadastro de Materiais**

O cadastro de novos materiais inicia-se através do menu "Materiais > Novo Cadastro", apresentando formulário estruturado em abas para organização lógica das informações. A aba "Dados Básicos" inclui campos para descrição, especificações técnicas, categoria e unidade de medida.

A descrição do material deve ser clara e detalhada, facilitando identificação posterior. Especificações técnicas devem incluir informações relevantes como modelo, marca, dimensões e características específicas. A categoria deve ser selecionada de lista pré- definida, garantindo padronização e facilitando consultas posteriores.

A aba "Dados Financeiros" inclui valor unitário, fornecedor e informações sobre aquisição. O valor deve ser informado com precisão para cálculos corretos de patrimônio e relatórios financeiros. Informações sobre fornecedor e processo de aquisição são importantes para rastreabilidade e prestação de contas.

A aba "Localização" permite definir a posição física do material no almoxarifado, incluindo prédio, sala, estante e prateleira. Esta informação é fundamental para localização rápida durante operações e otimização do layout do almoxarifado.

##### **Registro de Entradas**

O registro de entrada de materiais é acessado através do menu "Movimentações > Entrada de Materiais", apresentando formulário que permite seleção do material através de busca inteligente ou código de barras. O sistema valida automaticamente se o material já existe no cadastro ou se deve ser criado novo registro.

Para materiais existentes, o sistema apresenta informações atuais de estoque e localização, facilitando conferência durante o recebimento. A quantidade recebida deve ser informada com precisão, pois atualizará automaticamente o estoque disponível.

Informações sobre o fornecimento incluem número da nota fiscal, data de recebimento, responsável pelo recebimento e observações sobre o estado dos materiais. Estas informações são importantes para controle de garantia e rastreabilidade de problemas futuros.

O sistema gera automaticamente comprovante de entrada que pode ser impresso ou salvo em PDF, incluindo todas as informações relevantes da operação e código de rastreamento único.

### **Processamento de Saídas**

O processamento de saídas inicia-se com seleção do material através de consulta de estoque, verificando automaticamente disponibilidade e localização. O sistema apresenta informações detalhadas sobre o material, incluindo especificações técnicas e histórico de movimentações recentes.

A solicitação de saída deve incluir identificação do solicitante, finalidade da retirada, quantidade solicitada e data prevista de devolução quando aplicável. O sistema valida automaticamente se o solicitante possui autorização para retirar o tipo de material solicitado e se não possui pendências anteriores.

O fluxo de aprovação é acionado automaticamente conforme o valor e tipo de material, encaminhando a solicitação para o autorizador apropriado. Notificações automáticas são enviadas para todas as partes envolvidas, mantendo transparência no processo.

Após aprovação, o responsável pelo almoxarifado processa a saída física, confirmando a retirada no sistema e gerando termo de responsabilidade quando necessário. O estoque é atualizado automaticamente e alertas são configurados para controle de devolução.

### **4.3.3 Relatórios e Consultas**

#### **Consultas de Estoque**

As consultas de estoque permitem visualização em tempo real da posição de materiais, com filtros por categoria, localização, status e faixa de valores. A interface apresenta resultados em formato de tabela com opções de ordenação e exportação para Excel ou PDF.

Filtros avançados incluem busca por texto livre na descrição, faixa de datas de cadastro, fornecedor e responsável pela guarda. Resultados podem ser agrupados por categoria ou localização, facilitando análises específicas.

Para cada material listado, são apresentadas informações essenciais como código,



descrição, quantidade em estoque, localização física e data da última movimentação. Clicando em um item específico, é possível acessar detalhes completos incluindo histórico de movimentações.

### **Relatórios de Movimentação**

Os relatórios de movimentação apresentam histórico detalhado de todas as operações realizadas em período específico, com filtros por tipo de movimentação, material, responsável e unidade. O relatório inclui totalizações por categoria e análises de tendência de consumo.

Informações apresentadas incluem data e hora da movimentação, tipo de operação, material envolvido, quantidade, responsável, autorizador e observações. Totalizações automáticas facilitam análise de volumes e identificação de padrões de uso.

O relatório pode ser configurado para apresentação sintética ou detalhada, conforme necessidade do usuário. Opções de exportação incluem PDF para impressão formal e Excel para análises adicionais.

### **Relatórios Gerenciais**

Os relatórios gerenciais são direcionados para apoio à tomada de decisão, incluindo análises de consumo, identificação de materiais de baixo giro, previsão de necessidades de reposição e indicadores de performance operacional.

O relatório de análise de consumo apresenta estatísticas de uso por categoria de material, identificando itens de maior e menor rotatividade. Gráficos facilitam visualização de tendências e sazonalidades no consumo.

Indicadores de performance incluem tempo médio de processamento de solicitações, taxa de aprovação de saídas, precisão de inventários e cumprimento de prazos de devolução. Estes indicadores são fundamentais para melhoria contínua dos processos.

## **4.4 Procedimentos de Backup e Recuperação**

### **4.4.1 Estratégia de Backup**

#### **Backup Completo Diário**

O backup completo é executado diariamente durante horário de menor movimento operacional, tipicamente entre 02:00 e 04:00 horas. Este backup inclui todos os dados do sistema, estrutura de banco de dados, arquivos de configuração e logs de auditoria.

O processo utiliza ferramentas nativas do PostgreSQL (pg\_dump) para garantir

consistência dos dados durante a operação. O backup é comprimido automaticamente para otimizar espaço de armazenamento e tempo de transferência.

Verificação de integridade é executada automaticamente após cada backup, incluindo teste de restauração em ambiente isolado para garantir que os dados podem ser recuperados adequadamente. Relatórios de status são gerados automaticamente e enviados para administradores do sistema.

### **Backup Incremental**

Backups incrementais são executados a cada 4 horas durante período operacional, capturando apenas as alterações realizadas desde o último backup. Esta estratégia reduz tempo de processamento e espaço de armazenamento necessário.

O backup incremental utiliza log de transações (WAL) do PostgreSQL para identificar e capturar apenas as modificações realizadas. Arquivos de log são arquivados automaticamente e mantidos por período definido na política de retenção.

Testes de recuperação incremental são realizados semanalmente para garantir que a sequência de backups permite restauração completa do sistema para qualquer ponto no tempo dentro do período de retenção.

### **Replicação em Tempo Real**

Para ambientes de alta disponibilidade, é implementada replicação em tempo real para servidor secundário, mantendo cópia atualizada dos dados que pode ser ativada rapidamente em caso de falha do servidor principal.

A replicação utiliza streaming replication do PostgreSQL, garantindo latência mínima entre servidor principal e secundário. Monitoramento contínuo verifica sincronização e alerta automaticamente sobre problemas de conectividade ou atraso na replicação.

Procedimentos de failover automático podem ser configurados para ativação do servidor secundário em caso de falha do principal, minimizando tempo de indisponibilidade do sistema.

#### **4.4.2 Procedimentos de Recuperação**

##### **Recuperação Completa do Sistema**

Em caso de falha catastrófica que exija recuperação completa do sistema, o procedimento inicia-se com preparação de novo ambiente de servidor, incluindo instalação do sistema operacional, PostgreSQL e Scriptcase conforme especificações

originais.

A restauração dos dados utiliza o backup completo mais recente, seguido pela aplicação sequencial de todos os backups incrementais posteriores. Este processo reconstrói o estado do banco de dados até o ponto mais próximo possível da falha.

Após restauração dos dados, é necessário reconfigurar parâmetros específicos do ambiente, incluindo conexões de rede, certificados SSL e integrações com outros sistemas. Testes completos de funcionalidade devem ser executados antes de liberar o sistema para uso operacional.

### **Recuperação Point-in-Time**

Para situações que exigem recuperação para momento específico no tempo, como correção de erro operacional ou investigação de problemas, o sistema suporta recuperação point-in-time utilizando combinação de backup completo e logs de transação.

O procedimento permite especificar data e hora exatas para recuperação, reconstruindo o estado do banco de dados exatamente como estava no momento solicitado. Esta funcionalidade é valiosa para auditoria e correção de problemas específicos.

A recuperação point-in-time é executada em ambiente isolado para evitar impacto no sistema de produção, permitindo análise detalhada antes de decidir sobre aplicação das correções no ambiente operacional.

### **Recuperação de Dados Específicos**

Para recuperação de registros específicos que foram excluídos acidentalmente ou corrompidos, o sistema oferece procedimentos granulares que permitem restauração seletiva sem impactar outros dados.

O processo utiliza backups e logs de transação para identificar e extrair apenas os dados necessários, aplicando-os ao sistema de produção através de scripts SQL específicos. Esta abordagem minimiza impacto operacional e tempo de indisponibilidade.

Validação cuidadosa é executada antes e após a recuperação para garantir integridade dos dados e funcionamento correto do sistema. Logs detalhados documentam todas as operações realizadas para auditoria e referência futura.

## **4.5 Especificação de Testes e Validação**

#### **4.5.1 Estratégia de Testes**

##### **Testes Unitários**

Embora o Scriptcase não ofereça framework nativo para testes automatizados, são implementados procedimentos sistemáticos de validação que verificam funcionamento correto de cada componente individual do sistema. Estes testes incluem validação de regras de negócio, cálculos automáticos e integridade de dados.

Cada funcionalidade desenvolvida passa por bateria de testes que verifica comportamento esperado em condições normais e excepcionais. Casos de teste incluem validação de entrada de dados, processamento de regras de negócio e geração de saídas corretas.

Documentação detalhada registra todos os casos de teste executados, resultados obtidos e correções implementadas. Esta documentação serve como base para testes de regressão e validação de futuras modificações.

##### **Testes de Integração**

Os testes de integração verificam funcionamento correto das interfaces entre diferentes módulos do sistema e com sistemas externos. Incluem validação de sincronização de dados, comunicação entre componentes e manutenção de consistência.

Cenários de teste simulam operações reais que envolvem múltiplos módulos, como processo completo de entrada de material que inclui cadastro, movimentação, atualização de estoque e geração de relatórios. Validação confirma que dados fluem corretamente entre componentes.

Testes de integração com sistemas externos utilizam ambientes de teste quando disponíveis ou simulação de interfaces para validar protocolos de comunicação e tratamento de erros.

##### **Testes de Performance**

Testes de performance avaliam comportamento do sistema sob diferentes cargas de trabalho, identificando gargalos e validando atendimento aos requisitos de tempo de resposta estabelecidos.

Simulação de múltiplos usuários simultâneos verifica capacidade do sistema de manter performance adequada durante picos de utilização. Testes incluem operações típicas como consultas de estoque, registro de movimentações e geração de relatórios.

Monitoramento de recursos do servidor durante testes identifica utilização de CPU, memória e disco, permitindo otimizações específicas para melhorar performance geral do sistema.

#### **4.5.2 Casos de Teste Críticos**

##### **Teste de Integridade de Estoque**

Este teste crítico verifica se o sistema mantém consistência entre movimentações registradas e saldos de estoque, fundamental para confiabilidade do controle patrimonial.

O teste executa sequência de operações incluindo entradas, saídas, transferências e devoluções, verificando após cada operação se o saldo calculado corresponde ao esperado. Validação inclui verificação de totais por categoria e localização física.

Cenários de erro são testados intencionalmente, incluindo tentativas de saída com estoque insuficiente, movimentações com dados inválidos e operações simultâneas que poderiam causar inconsistências.

##### **Teste de Controle de Acesso**

Validação rigorosa do sistema de controle de acesso verifica se usuários têm acesso apenas às funcionalidades autorizadas para seu perfil, fundamental para segurança e conformidade.

Testes incluem tentativas de acesso a funcionalidades não autorizadas, verificação de timeout de sessão e validação de controles de aprovação hierárquica. Cada perfil de usuário é testado sistematicamente.

Simulação de ataques comuns como tentativas de SQL injection, cross-site scripting e manipulação de URLs verifica robustez dos controles de segurança implementados.

##### **Teste de Backup e Recuperação**

Procedimentos de backup e recuperação são testados regularmente para garantir que dados podem ser restaurados adequadamente em caso de falha.

Testes incluem simulação de falhas de hardware, corrupção de dados e exclusão acidental de informações críticas. Cada cenário é seguido por procedimento de recuperação completo, validando integridade dos dados restaurados.

Tempo de recuperação é medido e comparado com objetivos

estabelecidos, identificando oportunidades de otimização dos procedimentos.

### **4.5.3 Validação com Usuários**

#### **Testes de Aceitação**

Testes de aceitação são conduzidos com usuários finais para validar se o sistema atende adequadamente às necessidades operacionais e expectativas de usabilidade.

Usuários representativos de diferentes perfis executam cenários reais de trabalho, incluindo operações rotineiras e situações excepcionais. Feedback é coletado sistematicamente sobre facilidade de uso, clareza de interfaces e adequação de funcionalidades.

Resultados dos testes de aceitação orientam ajustes finais antes da implantação, garantindo que o sistema seja bem recebido pelos usuários e atenda efetivamente às necessidades operacionais.

#### **Validação de Relatórios**

Relatórios gerados pelo sistema são validados por usuários experientes que verificam precisão dos dados, clareza da apresentação e adequação às necessidades gerenciais.

Comparação com relatórios manuais existentes confirma que informações são apresentadas corretamente e cálculos estão precisos. Validação inclui verificação de totalizações, filtros e opções de exportação.

Sugestões de melhorias na apresentação e conteúdo dos relatórios são coletadas e implementadas quando viáveis, garantindo que produtos finais atendam efetivamente às necessidades dos gestores.

## **4.6 Guia de Manutenção**

### **4.6.1 Manutenção Preventiva**

#### **Rotinas Diárias**

Procedimentos de manutenção diária incluem verificação de logs de sistema para identificação de erros ou alertas, monitoramento de performance do banco de dados e validação de execução correta dos backups automáticos.

Verificação de espaço em disco garante que não há risco de esgotamento de capacidade que poderia impactar operação do sistema. Limpeza automática de arquivos temporários e logs antigos mantém ambiente organizado e otimizado.

Monitoramento de conectividade com sistemas integrados identifica problemas de comunicação que poderiam impactar sincronização de dados ou funcionalidades dependentes.

### **Rotinas Semanais**

Manutenção semanal inclui análise detalhada de logs de auditoria para identificação de padrões anômalos ou tentativas de acesso não autorizado. Relatórios de utilização do sistema orientam otimizações de performance.

Teste de procedimentos de backup e recuperação garante que sistemas de proteção de dados funcionam adequadamente. Validação inclui teste de restauração em ambiente isolado.

Atualização de estatísticas do banco de dados otimiza performance de consultas complexas e relatórios. Reorganização de índices quando necessário mantém eficiência das operações.

### **Rotinas Mensais**

Manutenção mensal inclui análise abrangente de performance do sistema, identificando tendências de crescimento de dados e necessidades de otimização. Planejamento de capacidade orienta decisões sobre expansão de recursos.

Revisão de políticas de segurança e controle de acesso garante que permissões permanecem adequadas conforme mudanças organizacionais. Atualização de senhas de sistema e certificados SSL mantém segurança atualizada.

Validação de integrações com sistemas externos confirma que comunicação permanece funcionando corretamente e dados estão sincronizados adequadamente.

## **4.6.2 Resolução de Problemas**

### **Problemas Comuns e Soluções**

Documentação de problemas comuns e suas soluções facilita resolução rápida de incidentes recorrentes. Inclui sintomas típicos, causas prováveis e procedimentos de correção passo a passo.

Problemas de performance frequentemente relacionam-se com crescimento de volume de dados ou consultas não otimizadas. Soluções incluem otimização de índices, ajuste de parâmetros de configuração e implementação de cache quando apropriado.

Falhas de integração com sistemas externos podem resultar de mudanças de configuração, problemas de rede ou atualizações nos sistemas

integrados.

Procedimentos de diagnóstico orientam identificação e correção rápida.

### **Procedimentos de Escalação**

Estrutura de escalação define responsabilidades e prazos para resolução de problemas conforme severidade. Problemas críticos que impactam operação são escalados imediatamente para suporte especializado.

Documentação detalhada de cada incidente inclui sintomas observados, ações tomadas e resultado obtido. Esta informação orienta melhorias nos procedimentos e prevenção de problemas similares.

Comunicação proativa com usuários durante incidentes mantém transparência sobre status de resolução e impactos esperados na operação.

### **4.6.3 Evolução e Melhorias**

#### **Processo de Mudanças**

Implementação de mudanças no sistema segue processo formal que inclui análise de impacto, aprovação hierárquica e testes adequados antes da implantação em produção.

Solicitações de melhorias são avaliadas considerando benefício operacional, complexidade de implementação e recursos necessários. Priorização considera necessidades críticas da operação e disponibilidade de recursos técnicos.

Documentação de todas as mudanças implementadas mantém histórico de evolução do sistema e facilita manutenção futura. Controle de versão permite reversão de mudanças se necessário.

#### **Planejamento de Expansões**

Crescimento futuro do sistema é planejado considerando evolução das necessidades operacionais, crescimento de volume de dados e disponibilidade de recursos tecnológicos.

Arquitetura modular facilita adição de novas funcionalidades sem impacto nas existentes. Planejamento inclui estimativas de recursos necessários e cronograma de implementação.

Integração com novos sistemas ou expansão das integrações existentes é planejada considerando padrões tecnológicos e políticas de segurança da instituição.