

UNIVERSIDADE FEDERAL DE PERNAMBUCO
CENTRO DE INFORMÁTICA
GRADUAÇÃO EM SISTEMAS DE INFORMAÇÃO

Sistema de Gerenciamento de Biblioteca Universitária

Projeto da Disciplina IF976 - Banco de Dados

Felipe Santos
Juliana Serafim
Matheus Dalia
Pedro Balbino

Recife, Dezembro de 2024

Conteúdo

1	Introdução	3
2	Contextualização	3
3	Descrição do Minimundo	3
3.1	Visão Geral	3
3.2	Contexto	3
4	Estruturas Principais	4
4.1	Entidades e Seus Atributos	4
5	Regras de Negócio	6
6	Operações do Sistema	8
7	Esquema Conceitual	9
7.1	Desenvolvimento do Modelo ER	9
7.2	Entidades	9
7.3	Atributos	10
7.3.1	Atributos Identificadores	10
7.3.2	Atributos Compostos	10
7.3.3	Atributos Multivalorados	10
7.3.4	Atributos Derivados	10
7.4	Relacionamentos	11
7.4.1	Relacionamentos Binários	11
7.4.2	Restrições de Participação	11
7.5	Diagrama ER	11
7.5.1	Detalhamento dos Elementos	13
7.6	Relacionamentos Principais	14
7.7	Especializações/Generalizações	15
7.7.1	Especialização de USUÁRIO	15
7.7.2	Justificativa da Especialização	15
8	Esquema Relacional	16
8.1	Processo de Mapeamento	16
8.2	Diagrama do Banco de Dados	16
8.3	Relações Resultantes	17
8.4	Implementação em PostgreSQL	19
8.5	Justificativas do Mapeamento	22
8.5.1	Entidades Principais	22
8.5.2	Especializações	22
8.6	Restrições de Integridade	23
9	Operações e Consultas	23
9.1	Visão Geral	23
9.2	Gestão do Acervo	24
9.3	Gestão de Usuários	26
9.4	Empréstimos e Devoluções	27

9.5	Relatórios e Consultas Analíticas	28
9.6	Consultas de Monitoramento	29
9.7	Indicadores de Desempenho	30
10	Conclusão	30

1 Introdução

Este documento apresenta o projeto de banco de dados para um Sistema de Gerenciamento de Biblioteca Universitária, desenvolvido como parte da disciplina IF976 - Banco de Dados. O projeto está estruturado em quatro entregas principais:

1. **Definição e descrição do minimundo** (09/12/2024)
2. **Esquema conceitual** (11/12/2024)
3. **Esquema relacional** (16/12/2024)
4. **Lista de operações e consultas** (18/12/2024)

Este documento contempla as duas primeiras entregas do projeto, apresentando tanto a descrição detalhada do minimundo quanto o esquema conceitual do banco de dados.

2 Contextualização

Este documento apresenta a primeira etapa do projeto da disciplina IF976 - Banco de Dados, que consiste no desenvolvimento de um sistema de gerenciamento para biblioteca universitária. O sistema proposto visa atender às necessidades específicas do ambiente acadêmico, facilitando o controle e a gestão do acervo bibliográfico.

3 Descrição do Minimundo

3.1 Visão Geral

Objetivo do Sistema
O sistema tem como finalidade gerenciar de forma eficiente todos os aspectos operacionais de uma biblioteca universitária, desde o cadastro e controle do acervo até o gerenciamento de empréstimos e usuários.

3.2 Contexto

A biblioteca universitária necessita de um sistema informatizado para automatizar seus processos internos e melhorar a qualidade dos serviços oferecidos à comunidade acadêmica. O sistema deve contemplar:

- Gerenciamento completo do acervo bibliográfico
- Controle eficiente de empréstimos e devoluções
- Gestão de usuários (alunos, professores e funcionários)
- Sistema de multas e penalidades
- Geração de relatórios gerenciais e estatísticos
- Controle de acesso e segurança

4 Estruturas Principais

4.1 Entidades e Seus Atributos

Livro

- **Identificação:**
 - ISBN (chave primária)
 - Código de tombamento
 - Código de barras
- **Informações Básicas:**
 - Título
 - Subtítulo
 - Autor(es)
 - Editora
 - Edição
 - Ano de publicação
- **Categorização:**
 - Área de conhecimento
 - Palavras-chave
 - Classificação (ex: livro, periódico, tese)
- **Localização:**
 - Seção
 - Prateleira
 - Posição
- **Controle:**
 - Quantidade de exemplares
 - Status de conservação
 - Data de aquisição

Usuário**• Dados Pessoais:**

- ID (chave primária)
- Nome completo
- CPF
- RG
- Data de nascimento
- Endereço completo
- Telefone(s)
- Email

• Dados Acadêmicos:

- Tipo de vínculo (aluno, professor, funcionário)
- Curso/Departamento
- Número de matrícula/SIAPE
- Data de início do vínculo

• Dados do Sistema:

- Login
- Senha
- Status da conta
- Histórico de empréstimos
- Situação de multas

Empréstimo

- **Identificação:**
 - Número do empréstimo (chave primária)
 - ID do usuário (chave estrangeira)
 - ISBN do livro (chave estrangeira)
- **Datas:**
 - Data do empréstimo
 - Data prevista para devolução
 - Data efetiva de devolução
- **Controle:**
 - Status do empréstimo
 - Número de renovações
 - Funcionário responsável
 - Observações

5 Regras de Negócio

RN1. Empréstimos

- Professores podem emprestar até 7 livros por 30 dias
- Alunos podem emprestar até 5 livros por 15 dias
- Funcionários podem emprestar até 3 livros por 15 dias
- Renovações são permitidas até 2 vezes, se não houver reserva
- Um exemplar só pode ser emprestado se não estiver reservado

RN2. Multas e Penalidades

- Multa diária por atraso de R\$ 1,00 por livro
- Bloqueio de novos empréstimos para usuários com multas pendentes
- Após 30 dias de atraso, suspensão do direito de empréstimo por 60 dias
- Sistema deve enviar notificações automáticas de atraso

RN3. Reservas

- Limite de 3 reservas simultâneas por usuário
- Reserva válida por 24 horas após a devolução do livro
- Ordem de prioridade por data de solicitação

RN4. Cadastro e Manutenção

- Todos os livros devem ter ISBN válido
- Usuários devem manter dados cadastrais atualizados

- Baixa de livros só pode ser feita por bibliotecários
- Inventário anual obrigatório

6 Operações do Sistema

Operações Principais

OP1. Gestão de Acervo

- Cadastro de novos livros
- Atualização de informações bibliográficas
- Controle de exemplares
- Baixa de material
- Inventário

OP2. Controle de Empréstimos

- Registro de empréstimos
- Renovações
- Devoluções
- Reservas
- Controle de atrasos

OP3. Gestão de Usuários

- Cadastro de novos usuários
- Atualização de dados cadastrais
- Controle de situação (ativo/inativo)
- Histórico de empréstimos
- Gestão de multas

OP4. Relatórios e Estatísticas

- Relatórios de movimentação
- Estatísticas de uso
- Relatórios de multas
- Histórico de operações
- Indicadores de desempenho

OP5. Administração do Sistema

- Controle de acesso
- Backup de dados
- Configurações do sistema
- Logs de operações
- Manutenção do sistema

7 Esquema Conceitual

7.1 Desenvolvimento do Modelo ER

O esquema conceitual foi desenvolvido seguindo as etapas de modelagem Entidade-Relacionamento, identificando os principais componentes do sistema e suas interações. O processo incluiu:

1. Identificação das entidades regulares e fracas
2. Definição dos atributos (simples, compostos e multivalorados)
3. Estabelecimento dos relacionamentos
4. Determinação das cardinalidades
5. Análise das especializações/generalizações necessárias

7.2 Entidades

Identificação das Entidades

- **Entidades Fortes:**

- LIVRO: representa a obra bibliográfica
- USUÁRIO: representa os usuários do sistema
- FUNCIONÁRIO: representa os funcionários da biblioteca

- **Entidades Fracas:**

- EXEMPLAR: depende da existência de um LIVRO
- EMPRÉSTIMO: depende de USUÁRIO e EXEMPLAR

7.3 Atributos

Tipos de Atributos

7.3.1 Atributos Identificadores

- LIVRO: ISBN (chave primária)
- USUÁRIO: ID, CPF (chaves candidatas)
- EXEMPLAR: Código de Tombamento + ISBN (chave parcial + chave do proprietário)
- EMPRÉSTIMO: ID do empréstimo (chave artificial)

7.3.2 Atributos Compostos

- Endereço (logradouro, número, complemento, bairro, cidade, estado, CEP)
- Nome Completo (primeiro nome, sobrenome)

7.3.3 Atributos Multivalorados

- Telefone(s)
- Autor(es)
- Palavras-chave

7.3.4 Atributos Derivados

- Idade (derivado da data de nascimento)
- Tempo de empréstimo (derivado das datas de empréstimo e devolução)
- Valor da multa (derivado do tempo de atraso)

7.4 Relacionamentos

Tipos de Relacionamentos

7.4.1 Relacionamentos Binários

- **POSSUI** (LIVRO-EXEMPLAR)
 - Tipo: 1:N
 - Participação total do lado EXEMPLAR
 - Atributos: nenhum
- **REALIZA** (USUÁRIO-EMPRÉSTIMO)
 - Tipo: 1:N
 - Participação total do lado EMPRÉSTIMO
 - Atributos: nenhum
- **PARTICIPA** (EXEMPLAR-EMPRÉSTIMO)
 - Tipo: 1:N
 - Participação total do lado EMPRÉSTIMO
 - Atributos: data_empréstimo, data_devolução_prevista
- **CADASTRA** (FUNCIONÁRIO-EMPRÉSTIMO)
 - Tipo: 1:N
 - Participação total do lado EMPRÉSTIMO
 - Atributos: data_cadastro

7.4.2 Restrições de Participação

- EXEMPLAR deve pertencer a um LIVRO (participação total)
- EMPRÉSTIMO deve ter um USUÁRIO e um EXEMPLAR (participação total)
- EMPRÉSTIMO deve ser cadastrado por um FUNCIONÁRIO (participação total)

7.5 Diagrama ER

A seguir, apresentamos o diagrama ER completo do sistema, que representa todas as entidades, relacionamentos e atributos identificados para o Sistema de Gerenciamento de Biblioteca Universitária:

ModeloConceitual.png

7.5.1 Detalhamento dos Elementos

Elementos do Diagrama

1. Entidade Livro

- Chave primária: ISBN
- Atributos descritivos: título, subtítulo, editora, edição
- Atributos de categorização: área de conhecimento, palavras-chave

2. Entidade Exemplar

- Identificação: código de tombamento, código de barras
- Controle: status de conservação, data de aquisição
- Localização: seção, prateleira, posição

3. Entidade Usuário

- Identificadores: ID, CPF, RG
- Dados pessoais: nome completo, data de nascimento, endereço
- Especialização em: Aluno, Professor e Funcionário

4. Entidade Empréstimo

- Controle temporal:
 - Data do empréstimo
 - Data prevista para devolução
 - Data efetiva de devolução
- Gestão:
 - Status do empréstimo
 - Número de renovações
 - Observações
- Rastreamento:
 - Funcionário responsável
 - ID do empréstimo

7.6 Relacionamentos Principais

Relacionamentos do Sistema

R1. Livro-Exemplar (Possui)

- Cardinalidade: 1:N
- Um livro pode ter vários exemplares
- Cada exemplar pertence a exatamente um livro

R2. Exemplar-Empréstimo (Participa)

- Cardinalidade: 1:N
- Um exemplar pode participar de vários empréstimos (em momentos diferentes)
- Cada empréstimo está associado a exatamente um exemplar

R3. Usuário-Empréstimo (Realiza)

- Cardinalidade: 1:N
- Um usuário pode realizar vários empréstimos
- Cada empréstimo está associado a exatamente um usuário

R4. Funcionário-Empréstimo (Cadastrado_por)

- Cardinalidade: 1:N
- Um funcionário pode cadastrar vários empréstimos
- Cada empréstimo é cadastrado por exatamente um funcionário

7.7 Especializações/Generalizações

Hierarquia de Especialização

7.7.1 Especialização de USUÁRIO

- Tipo: Disjunta (d)
- Completude: Total
- Critério: Tipo de vínculo com a instituição
- Subclasses:
 - ALUNO
 - * Atributos específicos: número_matrícula, curso
 - PROFESSOR
 - * Atributos específicos: SIAPE, departamento
 - FUNCIONÁRIO
 - * Atributos específicos: SIAPE, setor

7.7.2 Justificativa da Especialização

A especialização foi escolhida porque:

- Cada tipo de usuário tem atributos específicos
- As regras de empréstimo variam por tipo
- Um usuário não pode pertencer a múltiplas categorias
- Todo usuário deve ser classificado em uma categoria
- **Aluno**
 - Atributos específicos: número de matrícula, curso
 - Limite de empréstimo: 5 livros
 - Prazo de empréstimo: 15 dias
- **Professor**
 - Atributos específicos: SIAPE, departamento
 - Limite de empréstimo: 7 livros
 - Prazo de empréstimo: 30 dias
- **Funcionário**
 - Atributos específicos: SIAPE, departamento
 - Limite de empréstimo: 3 livros
 - Prazo de empréstimo: 15 dias

8 Esquema Relacional

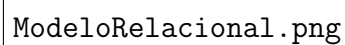
8.1 Processo de Mapeamento

O esquema relacional foi desenvolvido a partir do modelo conceitual (ER) seguindo as regras de mapeamento estabelecidas na literatura. O processo incluiu:

1. Mapeamento de entidades regulares
2. Mapeamento de entidades fracas
3. Mapeamento de relacionamentos
4. Tratamento de especializações
5. Mapeamento de atributos multivalorados
6. Mapeamento de atributos compostos

8.2 Diagrama do Banco de Dados

A Figura 2 apresenta o diagrama do esquema relacional completo, ilustrando todas as tabelas, seus relacionamentos e as chaves primárias e estrangeiras do sistema.



ModeloRelacional.png

Figura 2: Diagrama do Esquema Relacional do Sistema de Biblioteca

8.3 Relações Resultantes

Entidades Principais

- **LIVRO** (isbn, titulo, subtítulo, editora, edicao, ano_publicacao, area_conhecimento, data_cadastro)
- **EXEMPLAR** (codigo_tombamento, isbn[FK], codigo_barras, status_conservacao, data_aquisicao, secao, prateleira, posicao)
- **USUARIO** (id, tipo_usuario, nome, cpf, rg, data_nascimento, email, login, senha, status_conta)

Especializações

- **ALUNO** (id[FK], numero_matricula, curso)
- **PROFESSOR** (id[FK], siape, departamento)
- **FUNCIONARIO** (id[FK], siape, setor)

Relacionamentos

- **EMPRESTIMO** (id, usuario_id[FK], exemplar_codigo[FK], funcionario_id[FK], data_emprestimo, data_devolucao_prevista, data_devolucao_efetiva, status, numero_renovacoes, observacoes)
- **RESERVA** (id, usuario_id[FK], exemplar_codigo[FK], data_reserva, data_validade, status)
- **MULTA** (id, emprestimo_id[FK], valor, status, data_geracao, data_pagamento)

Atributos Multivalorados

- **TELEFONE_USUARIO** (usuario_id[FK], telefone)
- **AUTOR_LIVRO** (isbn[FK], autor)
- **PALAVRACHAVE_LIVRO** (isbn[FK], palavra_chave)

Atributos Compostos

- **ENDERECO_USUARIO** (usuario_id[FK], logradouro, numero, complemento, bairro, cidade, estado, cep)

8.4 Implementação em PostgreSQL

O esquema do banco de dados foi implementado utilizando PostgreSQL, um sistema de gerenciamento de banco de dados relacional robusto e de código aberto. A seguir, apresentamos o código SQL completo para a criação do esquema, dividido em seções lógicas:

1. Entidades Principais

```
-- Criação das tabelas principais
CREATE TABLE "Livro" (
    "isbn" VARCHAR(13) PRIMARY KEY,
    "titulo" VARCHAR(255) NOT NULL,
    "subtitulo" VARCHAR(255),
    "editora" VARCHAR(100) NOT NULL,
    "edicao" VARCHAR(50),
    "ano_publicacao" DATE NOT NULL,
    "area_conhecimento" VARCHAR(100) NOT NULL,
    "data_cadastro" DATE NOT NULL DEFAULT CURRENT_DATE
);

CREATE TABLE "Exemplar" (
    "codigo_tombamento" VARCHAR(20) PRIMARY KEY,
    "isbn" VARCHAR(13) NOT NULL,
    "codigo_barras" VARCHAR(50) UNIQUE,
    "status_conservacao" VARCHAR(20) NOT NULL,
    "data_aquisicao" DATE NOT NULL,
    "secao" VARCHAR(50) NOT NULL,
    "prateleira" VARCHAR(20) NOT NULL,
    "posicao" VARCHAR(20) NOT NULL,
    FOREIGN KEY ("isbn") REFERENCES "Livro" ("isbn")
);
```

2. Usuários e Especializações

```
CREATE TABLE "Usuario" (  
    "id" SERIAL PRIMARY KEY,  
    "tipo_usuario" VARCHAR(20) NOT NULL,  
    "nome" VARCHAR(100) NOT NULL,  
    "cpf" VARCHAR(11) UNIQUE NOT NULL,  
    "rg" VARCHAR(20) NOT NULL,  
    "data_nascimento" DATE NOT NULL,  
    "email" VARCHAR(100) UNIQUE NOT NULL,  
    "login" VARCHAR(50) UNIQUE NOT NULL,  
    "senha" VARCHAR(255) NOT NULL,  
    "status_conta" VARCHAR(20) NOT NULL DEFAULT 'ATIVO',  
    CONSTRAINT "chk_tipo_usuario"  
        CHECK (tipo_usuario IN ('ALUNO', 'PROFESSOR', 'FUNCIONARIO'))  
);  
  
CREATE TABLE "Aluno" (  
    "id" INTEGER PRIMARY KEY REFERENCES "Usuario" ("id") ON DELETE CASCADE,  
    "numero_matricula" VARCHAR(20) UNIQUE NOT NULL,  
    "curso" VARCHAR(100) NOT NULL  
);  
  
CREATE TABLE "Professor" (  
    "id" INTEGER PRIMARY KEY REFERENCES "Usuario" ("id") ON DELETE CASCADE,  
    "siape" VARCHAR(20) UNIQUE NOT NULL,  
    "departamento" VARCHAR(100) NOT NULL  
);  
  
CREATE TABLE "Funcionario" (  
    "id" INTEGER PRIMARY KEY REFERENCES "Usuario" ("id") ON DELETE CASCADE,  
    "siape" VARCHAR(20) UNIQUE NOT NULL,  
    "setor" VARCHAR(100) NOT NULL  
);
```

3. Empréstimos e Multas

```
CREATE TABLE "Emprestimo" (  
    "id" SERIAL PRIMARY KEY,  
    "usuario_id" INTEGER NOT NULL REFERENCES "Usuario" ("id"),  
    "exemplar_codigo" VARCHAR(20) NOT NULL  
        REFERENCES "Exemplar" ("codigo_tombamento"),  
    "funcionario_id" INTEGER NOT NULL REFERENCES "Funcionario" ("id"),  
    "data_emprestimo" TIMESTAMP NOT NULL DEFAULT CURRENT_TIMESTAMP,  
    "data_devolucao_prevista" DATE NOT NULL,  
    "data_devolucao_efetiva" DATE,  
    "status" VARCHAR(20) NOT NULL DEFAULT 'ATIVO',  
    "numero_renovacoes" INTEGER NOT NULL DEFAULT 0,  
    "observacoes" TEXT  
);  
  
CREATE TABLE "Multa" (  
    "id" SERIAL PRIMARY KEY,  
    "emprestimo_id" INTEGER NOT NULL REFERENCES "Emprestimo" ("id"),  
    "valor" DECIMAL(10,2) NOT NULL,  
    "status" VARCHAR(20) NOT NULL DEFAULT 'PENDENTE',  
    "data_geracao" DATE NOT NULL DEFAULT CURRENT_DATE,  
    "data_pagamento" DATE  
);
```

4. Atributos Multivalorados e Compostos

```
CREATE TABLE "Telefone_Usuario" (  
    "usuario_id" INTEGER REFERENCES "Usuario" ("id"),  
    "telefone" VARCHAR(20),  
    PRIMARY KEY ("usuario_id", "telefone")  
);  
  
CREATE TABLE "Autor_Livro" (  
    "isbn" VARCHAR(13) REFERENCES "Livro" ("isbn"),  
    "autor" VARCHAR(100),  
    PRIMARY KEY ("isbn", "autor")  
);  
  
CREATE TABLE "PalavraChave_Livro" (  
    "isbn" VARCHAR(13) REFERENCES "Livro" ("isbn"),  
    "palavra_chave" VARCHAR(50),  
    PRIMARY KEY ("isbn", "palavra_chave")  
);  
  
CREATE TABLE "Endereco_Usuario" (  
    "usuario_id" INTEGER PRIMARY KEY REFERENCES "Usuario" ("id"),  
    "logradouro" VARCHAR(100) NOT NULL,  
    "numero" VARCHAR(10) NOT NULL,  
    "complemento" VARCHAR(50),  
    "bairro" VARCHAR(50) NOT NULL,  
    "cidade" VARCHAR(50) NOT NULL,  
    "estado" CHAR(2) NOT NULL,  
    "cep" VARCHAR(8) NOT NULL  
);
```

8.5 Justificativas do Mapeamento

8.5.1 Entidades Principais

- **LIVRO**: Mantida como entidade independente com ISBN como chave primária natural.
- **EXEMPLAR**: Mapeada como entidade dependente do LIVRO, com código de tombamento como identificador único.
- **USUARIO**: Implementada como superclasse com identificador próprio para facilitar as especializações.

8.5.2 Especializações

A especialização de USUARIO foi implementada usando a abordagem de chave estrangeira para cada subclasse (Aluno, Professor, Funcionario), permitindo:

- Manutenção da integridade referencial
- Acesso eficiente aos dados específicos de cada tipo de usuário

- Flexibilidade para consultas e atualizações

8.6 Restrições de Integridade

Principais Restrições

RI1. Integridade de Entidade

- Chaves primárias NOT NULL e UNIQUE
- CPF e Email com restrição UNIQUE
- Tipos enumerados para status e categorias

RI2. Integridade Referencial

- ON DELETE RESTRICT para preservar histórico
- ON DELETE CASCADE para especializações
- Constraints nomeadas para melhor manutenção

RI3. Restrições de Domínio

- CHECK constraints para valores válidos
- Valores DEFAULT para campos comuns
- Tipos de dados adequados para cada campo

9 Operações e Consultas

9.1 Visão Geral

Esta seção apresenta as principais operações e consultas que podem ser realizadas no banco de dados, organizadas por área funcional. Para cada operação, são fornecidos os detalhes de implementação e exemplos de consultas SQL correspondentes.

9.2 Gestão do Acervo

Operações de Cadastro e Atualização

OA1. Cadastro de Novo Livro

- Inserção de informações bibliográficas básicas.
- Registro de autores.
- Cadastro de palavras-chave.

```
{\small
INSERT INTO "Livro" (isbn, titulo, editora, edicao,
    ano_publicacao, area_conhecimento)
VALUES ('9788533302273', 'Dom Casmurro', 'Editora XYZ',
    '1ª Edição', '2023-01-01', 'Literatura');

INSERT INTO "Autor_Livro" (isbn, autor)
VALUES ('9788533302273', 'Machado de Assis');
}
```

OA2. Registro de Exemplares

- Vinculação ao livro existente.
- Definição da localização física.
- Atribuição de código de tombamento.

```
{\small
INSERT INTO "Exemplar" (codigo_tombamento, isbn,
    codigo_barras, status_conservacao, data_aquisicao,
    secao, prateleira, posicao)
VALUES ('TOMB001', '9788533302273', 'BAR001', 'NOVO',
    CURRENT_DATE, 'LIT', 'A', '01');
}
```

Consultas do Acervo

CA1. Busca de Livros

- Por título, autor ou área.
- Verificação de disponibilidade.

```
{\small
SELECT l.*, COUNT(e.codigo_tombamento) as exemplares_disponiveis
FROM "Livro" l
LEFT JOIN "Exemplar" e ON l.isbn = e.isbn
LEFT JOIN "Emprestimo" emp ON e.codigo_tombamento =
    emp.exemplar_codigo AND emp.data_devolucao_efetiva IS NULL
WHERE l.titulo ILIKE '%pesquisa%'
GROUP BY l.isbn;
}
```

CA2. Relatórios de Acervo

- Estatísticas por área.
- Status de conservação.

```
{\small
SELECT l.area_conhecimento,
    COUNT(DISTINCT l.isbn) as total_titulos,
    COUNT(e.codigo_tombamento) as total_exemplares
FROM "Livro" l
LEFT JOIN "Exemplar" e ON l.isbn = e.isbn
GROUP BY l.area_conhecimento;
}
```

9.3 Gestão de Usuários

Operações de Usuários

OU1. Cadastro de Usuários

- Registro de dados pessoais.
- Criação de credenciais.
- Definição de perfil.

```
{\small
INSERT INTO "Usuario" (nome, cpf, tipo_usuario, email,
    login, senha, status_conta)
VALUES ('João Silva', '12345678901', 'ALUNO',
    'joao@email.com', 'joao.silva',
    'senha_hash', 'ATIVO');

INSERT INTO "Aluno" (id, numero_matricula, curso)
VALUES (CURRVAL('Usuario_id_seq'), '20241001',
    'Ciência da Computação');
}
```

OU2. Gerenciamento de Status

- Ativação/desativação de contas.
- Atualização de dados.

```
{\small
UPDATE "Usuario"
SET status_conta = 'INATIVO'
WHERE id = 1001 AND EXISTS (
    SELECT 1 FROM "Multa" m
    JOIN "Emprestimo" e ON m.emprestimo_id = e.id
    WHERE e.usuario_id = 1001
    AND m.status = 'PENDENTE'
);
}
```

9.4 Empréstimos e Devoluções

Operações de Circulação

OC1. Realização de Empréstimo

- Verificação de elegibilidade.
- Registro do empréstimo.
- Cálculo de data de devolução.

```
{\small
-- Verificar elegibilidade
SELECT COUNT(*) as emprestimos_ativos
FROM "Emprestimo"
WHERE usuario_id = 1001
AND data_devolucao_efetiva IS NULL;

-- Registrar empréstimo
INSERT INTO "Emprestimo" (usuario_id, exemplar_codigo,
    funcionario_id, data_devolucao_prevista)
VALUES (1001, 'TOMB001', 2001,
    CURRENT_DATE + INTERVAL '15 days');
}
```

OC2. Processamento de Devolução

- Registro de devolução.
- Cálculo de multas.
- Atualização de status.

```
{\small
UPDATE "Emprestimo"
SET data_devolucao_efetiva = CURRENT_DATE,
    status = 'CONCLUIDO'
WHERE id = 3001;

-- Gerar multa se necessário
INSERT INTO "Multa" (emprestimo_id, valor, status)
SELECT id,
    (CURRENT_DATE - data_devolucao_prevista) * 1.00,
    'PENDENTE'
FROM "Emprestimo"
WHERE id = 3001
AND CURRENT_DATE > data_devolucao_prevista;
}
```

9.5 Relatórios e Consultas Analíticas

Relatórios Gerenciais

RG1. Estatísticas de Uso

- Frequência de empréstimos.
- Áreas mais procuradas.

```
{\small
SELECT l.area_conhecimento,
       COUNT(e.id) as total_emprestimos,
       COUNT(DISTINCT e.usuario_id) as usuarios_unicos
FROM "Livro" l
JOIN "Exemplar" ex ON l.isbn = ex.isbn
JOIN "Emprestimo" e ON ex.codigo_tombamento = e.exemplar_codigo
WHERE e.data_emprestimo >= CURRENT_DATE - INTERVAL '30 days'
GROUP BY l.area_conhecimento
ORDER BY total_emprestimos DESC;
}
```

RG2. Relatório de Multas

- Total de multas pendentes.
- Histórico de pagamentos.

```
{\small
SELECT u.nome, COUNT(m.id) as total_multas,
       SUM(m.valor) as valor_total,
       MAX(m.data_geracao) as multa_mais_recente
FROM "Usuario" u
JOIN "Emprestimo" e ON u.id = e.usuario_id
JOIN "Multa" m ON e.id = m.emprestimo_id
WHERE m.status = 'PENDENTE'
GROUP BY u.id, u.nome
ORDER BY valor_total DESC;
}
```

9.6 Consultas de Monitoramento

Monitoramento do Sistema

MS1. Atrasos e Notificações

- Identificação de atrasos.
- Geração de alertas.

```
{\small
SELECT u.nome, u.email, l.titulo,
       e.data_devolucao_prevista,
       CURRENT_DATE - e.data_devolucao_prevista as dias_atraso
FROM "Emprestimo" e
JOIN "Usuario" u ON e.usuario_id = u.id
JOIN "Exemplar" ex ON e.exemplar_codigo = ex.codigo_tombamento
JOIN "Livro" l ON ex.isbn = l.isbn
WHERE e.data_devolucao_efetiva IS NULL
AND e.data_devolucao_prevista < CURRENT_DATE;
}
```

MS2. Status de Reservas

- Fila de reservas.
- Disponibilidade de exemplares.

```
{\small
SELECT l.titulo, r.data_reserva,
       u.nome as usuario,
       ROW_NUMBER() OVER (
         PARTITION BY r.exemplar_codigo
         ORDER BY r.data_reserva
       ) as posicao_fila
FROM "Reserva" r
JOIN "Usuario" u ON r.usuario_id = u.id
JOIN "Exemplar" ex ON r.exemplar_codigo = ex.codigo_tombamento
JOIN "Livro" l ON ex.isbn = l.isbn
WHERE r.status = 'ATIVA'
ORDER BY r.exemplar_codigo, r.data_reserva;
}
```

9.7 Indicadores de Desempenho

Métricas do Sistema

- Taxa de Circulação

```
{\small
SELECT l.isbn, l.titulo,
       COUNT(e.id) as total_emprestimos,
       COUNT(e.id)::float /
         EXTRACT(MONTH FROM AGE(CURRENT_DATE, MIN(e.data_emprestimo)))
       as media_mensal
FROM "Livro" l
JOIN "Exemplar" ex ON l.isbn = ex.isbn
LEFT JOIN "Emprestimo" e ON ex.codigo_tombamento = e.exemplar_codigo
GROUP BY l.isbn, l.titulo
HAVING MIN(e.data_emprestimo) IS NOT NULL;
}
```

- Eficiência do Acervo

```
{\small
SELECT l.area_conhecimento,
       COUNT(DISTINCT l.isbn) as total_titulos,
       COUNT(DISTINCT e.id) as total_emprestimos,
       COUNT(DISTINCT e.id)::float / COUNT(DISTINCT l.isbn)
       as emprestimos_por_titulo
FROM "Livro" l
LEFT JOIN "Exemplar" ex ON l.isbn = ex.isbn
LEFT JOIN "Emprestimo" e ON ex.codigo_tombamento = e.exemplar_codigo
WHERE e.data_emprestimo >= CURRENT_DATE - INTERVAL '1 year'
GROUP BY l.area_conhecimento;
}
```

10 Conclusão

O sistema proposto visa atender todas as necessidades operacionais de uma biblioteca universitária moderna, proporcionando eficiência na gestão do acervo e melhor experiência para os usuários. A implementação seguirá as melhores práticas de desenvolvimento de banco de dados, garantindo integridade, segurança e desempenho adequado. O esquema conceitual apresentado reflete de forma precisa as entidades, relacionamentos e regras de negócio identificadas, fornecendo uma base sólida para as próximas etapas do projeto.