



UNIVERSIDADE FEDERAL DE SANTA MARIA
CENTRO DE TECNOLOGIA
DEPARTAMENTO DE ELETRÔNICA E COMPUTAÇÃO
CIÊNCIA DE DADOS I

Bruno Henrique Spies
Êmilly Farias Brunning
Mathias Cirolini Michelotti
Nathan Cezar
Vinícius Gabriel Schultz

Trabalho para a disciplina de ciência de dados I
Modelagem de um banco de dados relacional



UNIVERSIDADE FEDERAL DE SANTA MARIA
CENTRO DE TECNOLOGIA
DEPARTAMENTO DE ELETRÔNICA E COMPUTAÇÃO
CIÊNCIA DE DADOS I

Nomes do Alunos(as):

Bruno Henrique Spies
Êmilly Farias Brunning
Mathias Cirolini Michelotti
Nathan Cezar
Vinícius Gabriel Schultz

Matrículas:

202220187
emilly
202220937
202121115
202220819

E-mail:

email

Curso:

Engenharia de Computação

Nome do Profesor(a): Gabriel Machado Lunardi

Sumário

1	RESUMO	1
2	INTRODUÇÃO.....	2
3	METODOLOGIA.....	4
4	DESENVOLVIMENTO	5
	4.1 Modelo Conceitual	5
	4.2 Modelo Lógico.....	10
	4.3 Modelo Físico	12
	4.4 Interface Gráfica e Integração com o Banco de Dados	17
5	CONCLUSÃO.....	18
	REFERÊNCIAS BIBLIOGRÁFICAS	19

1 Resumo

O objetivo proposto pelo professor para o este trabalho é aplicar as técnicas de modelagem conceitual de bancos de dados relacionais e converter, de acordo com as regras especificadas, para um modelo lógico.

O presente relatório descreve o processo inicial da criação de um banco de dados para o gerenciamento de bibliotecas universitárias. Foram desenvolvidos nesta etapa do trabalho os Modelos Conceitual e Lógico. O projeto enfatiza a importância de um sistema computacional para organizar, armazenar e assegurar a integridade das informações, essenciais para a eficiência operacional de qualquer instituição.

Além de consolidar os conhecimentos técnicos adquiridos em aula, o projeto também inclui uma apresentação oral, na qual os estudantes demonstrarão suas habilidades de comunicação e a capacidade de trabalhar em equipe. O exercício, além de fortalecer a compreensão dos conceitos de banco de dados, também preparou os alunos para desafios futuros no mercado de trabalho, onde a integração de conhecimentos técnicos e habilidades interpessoais é fundamental.

2 Introdução

Gerir uma fábrica, empresa, banco ou até mesmo pequenas lojas, em qualquer cenário, é um desafio que exige bastante atenção e organização. O expressivo número de dados que necessitam ser armazenados e gerenciados precisam ser tratados com bastante delicadeza para garantir a confiabilidade, integridade e segurança, além de mantê-los organizados afim de facilitar sua consulta e leitura.

Uma forma prática, atual e tecnológica para fazer isso é utilizando um sistema computacional, pois automatiza todos os processos e métodos, garante a integridade dos dados, os armazena e mantém o histórico de alterações e agiliza a gerência das informações e a produtividade das tarefas. A grande maioria dos estabelecimentos adotam essa abordagem para gerir todo o negócio, de modo que se tornou o padrão atualmente.

Uma peça muito importante para esse sistema é a implementação de um banco de dados para armazenar e gerenciar as informações, sendo o gerenciamento de dados o assunto principal da disciplina em questão. Para a consolidação e prática do aprendizado, foi proposto pelo docente a implementação de um banco de dados, abrangendo o modelo conceitual, o modelo lógico, o modelo físico e uma interface gráfico, de um caso real ou fictício, com o intuito de que ao final do processo todos os alunos tenham compreendido perfeitamente os assuntos discutidos em aula e estejam aptos a realizar essa tarefa em suas futuras carreiras quando lhes forem solicitada.

O modelo conceitual de um banco de dados descreve o seu comportamento de forma abstrata e independente do sistema de gerenciamento de banco de dados (SGDB). Comumente é expresso em forma de diagrama utilizando a abordagem entidade-relacionamento (ER) que é a mais difundida na comunidade de programadores.

O modelo lógico baixa um pouco o nível e é dependente do SGBD escolhido. Ele pode ser expresso por diagramas assim como o modelo conceitual por meio do chamado modelo "pé de galinha" ou então com descrições textuais e tabelas.

O modelo físico é a implementação em código do banco de dados usualmente feita em SQL. Aqui é informado os tipos dos atributos e suas regras de negócio, além de definir campos opcionais ou não, auto-incremento das chaves e entre outras informações relevantes para a aplicação final.

A interface gráfica tem o intuito de capacitar os discentes a integrar o banco de dados criado até o modelo físico a um software real, de modo que necessitem realizar inserções, edições, remoções e buscas no banco e assim aplicando o restante do conteúdo a cerca de SQL aprendido.

Para o trabalho foi escolhido realizar a informatização de um banco de dados para gerencia das bibliotecas universitárias. A turma foi separada em grupos de quatro a cinco integrantes para a realização da tarefa. Além de aplicação dos conhecimentos, o docente avaliará também aspectos qualitativos como trabalho em equipe e oratória em uma apresentação oral do trabalho para a turma.

O enunciado formulado para a realização do trabalho foi o seguinte:

"Cada biblioteca tem um nome, endereço e seu código e pode ser pertencer a nenhum ou mais centros da faculdade e possuir nenhuma ou várias publicações.

Os centros têm um nome, sigla e seu código para ser identificado e podem ou não ter bibliotecas e alunos associados.

Alunos são cadastrados com nome, endereço e matrícula para os identificar e são locados a um centro e podem requisitar nenhum ou vários empréstimos nas bibliotecas.

Cada empréstimo possui data, hora e um código para identificá-lo, pode ser requisitado por somente um aluno, emprestar somente uma publicação e é efetuado por um funcionário

Cada funcionário tem um nome, seu salário e um código para ser identificado e pode efetuar nenhum ou vários empréstimos

Cada publicação tem ano, tipo e um código, deve estar de posse de somente uma biblioteca, pode ter associada nenhum ou vários empréstimos, nem sempre precisam ser acadêmicas, de autoajuda ou de literatura e pode ser escrita por um ou vários autores. As publicações acadêmicas têm área e edição; as de autoajuda, assunto; e as de literatura, gênero textual.

Cada autor tem um código, nome e país e pode escrever nenhuma ou várias publicações".

3 Metodologia

Inicialmente, o grupo se reuniu a fim de decidir o tema do trabalho, após isso, foram divididas tarefas para cada membro do grupo para garantir a uniformidade do trabalho entre todos os integrantes, para ao final reunir todas as contribuições e formar o trabalho completo.

Para esboçar o Diagrama Entidade-Relacionamento (DER) e o Modelo Lógico foi utilizado o software brModelo na versão *desktop*. O relatório foi redigido seguindo as normas da ABNT em linguagem Latex utilizando o editor Overleaf. A apresentação de slides foi desenvolvida pelo Canva. As imagens utilizadas foram editadas pelo GIMP junto do pacote PhotoGIMP que deixa sua interface semelhante à do Adobe Photoshop.

O modelo físico foi desenvolvido utilizando o SGDB MariaDB, porém para aplicações simples foi utilizado o SQLLiteOnline. A interface gráfica foi desenvolvida em linguagem Java Swing utilizando o NetBeans. Para alguns diagramas foi utilizado o software online LucidChart.

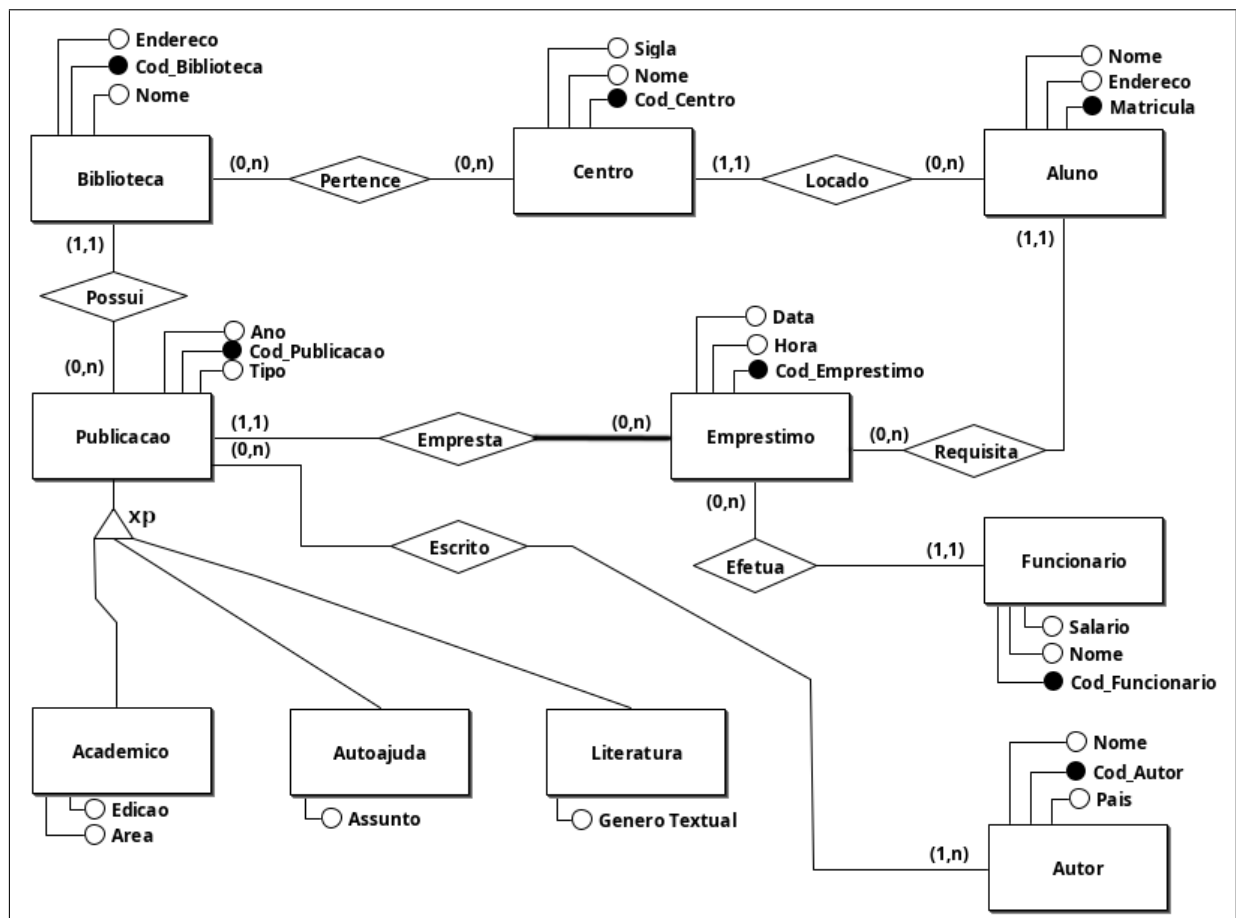
Todos os arquivos foram compartilhados entre o grupo por meio de um repositório no GitHub chamado "grupo7-CDI-20242-tg2".

4 Desenvolvimento

4.1 Modelo Conceitual

A Figura 1 abaixo se trata do Diagrama Entidade-Relacionamento desenvolvido pelo grupo, o qual representa a organização de um banco de dados para gerenciar as bibliotecas de uma universidade.

Figura 1 – Modelo Conceitual

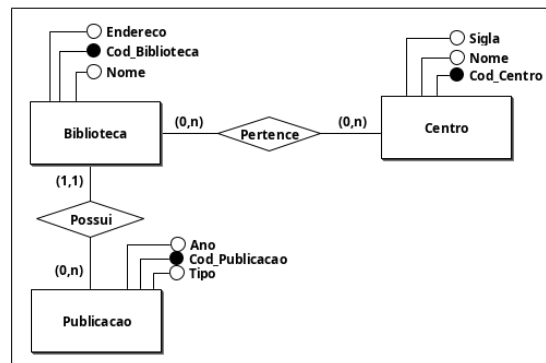


Fonte: Arquivo pessoal, 2024.

O Diagrama projetado possui as seguintes entidades: Biblioteca, Centro, além de Aluno, Publicação, Empréstimo, Funcionário e Autor, sendo que a entidade Publicação se especializa de forma parcial e exclusiva nas entidades Acadêmico, Autoajuda e Literatura.

A entidade Biblioteca possui os atributos: Cod Biblioteca (atributo identificador), Nome e Endereço e se relaciona com as entidades Centro e Publicação. Cada biblioteca pertence a zero ou mais centros e possui zero ou mais publicações. A entidade está ilustrada na Figura 2 abaixo.

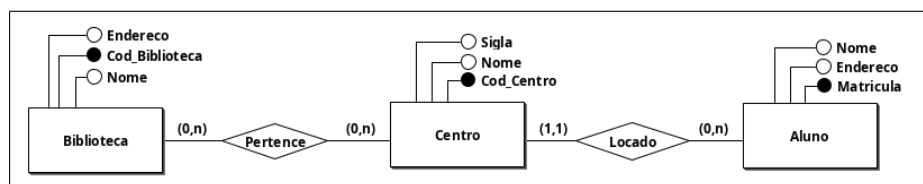
Figura 2 – Entidade Biblioteca



Fonte: Arquivo pessoal, 2024.

A entidade Centro ilustrada na Figura 3, possui os atributos: Cod_Centro (atributo identificador), Sigla e Nome e se relaciona com as entidades Biblioteca e Aluno. Cada Centro possui zero ou mais bibliotecas e tem locado zero ou mais alunos.

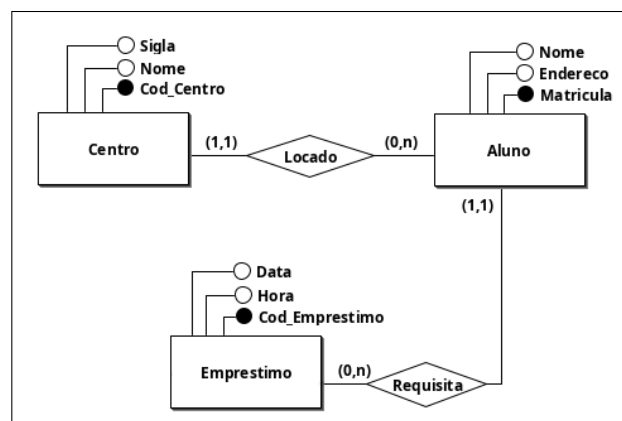
Figura 3 – Entidade Centro



Fonte: Arquivo pessoal, 2024.

A entidade Aluno ilustrada na Figura 4, possui os atributos: Matrícula (atributo identificador), Nome e Endereço e se relaciona com as entidades Centro e Empréstimo. Cada Aluno está locado a somente um centro e pode requisitar zero ou mais empréstimos de Publicações.

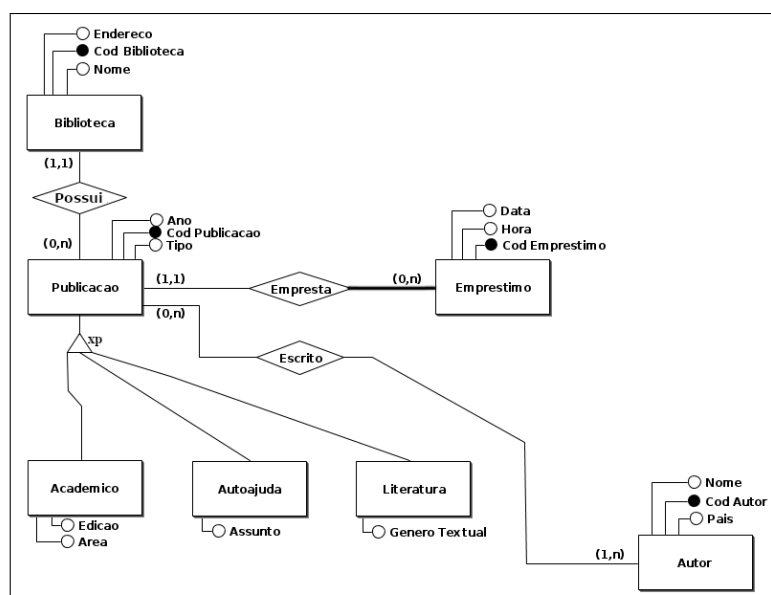
Figura 4 – Entidade Aluno



Fonte: Arquivo pessoal, 2024.

A entidade Publicação ilustrada na Figura 5, possui os atributos: Cod Publicação (atributo identificador), Ano e Tipo. O atributo Tipo foi adicionado pelo fato de que a entidade Publicação é genérica, e possui generalização parcial e exclusiva, portanto a presença do atributo Tipo é obrigatória para incluir generalizações não descritas no diagrama. A entidade se generaliza nas seguintes entidades especializadas: Acadêmico destinada para livros didáticos e semelhantes, Autoajuda para livros de aperfeiçoamento pessoal e Literatura dedicada à livros narrativos. Se relaciona com as entidades Biblioteca, Autor e Empréstimo. Cada publicação pertence a somente uma Biblioteca, é escrita por um ou mais Autores e pode ser emprestada zero ou muitas vezes.

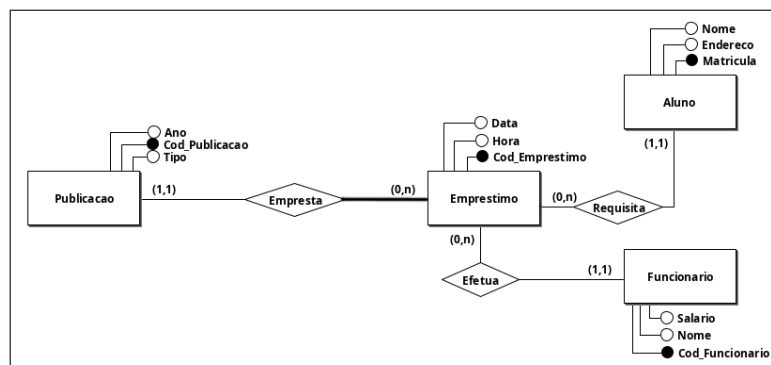
Figura 5 – Entidade Publicação



Fonte: Arquivo pessoal, 2024.

A entidade Empréstimo ilustrada na Figura 6 abaixo, é uma entidade fraca, pois depende da existência da entidade Publicação para existir e ser identificada, por isso o traçado do relacionamento desta entidade com Publicação é mais espesso. A entidade possui os seguintes atributos: Cod Empréstimo (atributo identificador), Data e Hora e se relaciona com as entidades Publicação, Funcionário e Aluno. Cada empréstimo empresta somente uma publicação, é efetuado por somente um funcionário e é requisitado por somente um aluno.

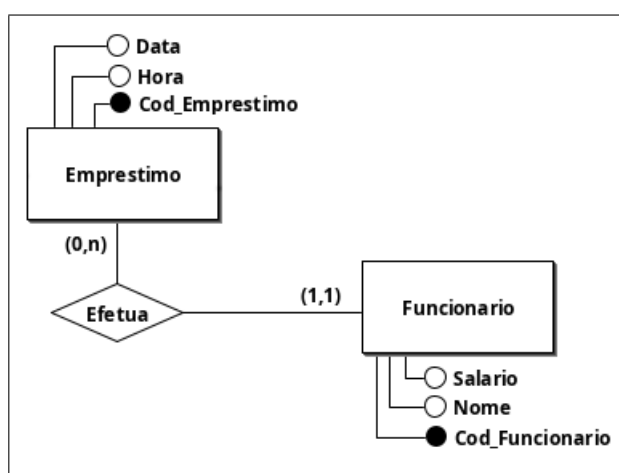
Figura 6 – Entidade Empréstimo



Fonte: Arquivo pessoal, 2024.

A entidade Funcionário possui os atributos: Cod Funcionario (atributo identificador), Salario e Nome e se relaciona com a entidade Empréstimo. Cada funcionário efetua zero ou vários empréstimos. A entidade está ilustrada na Figura 7 abaixo.

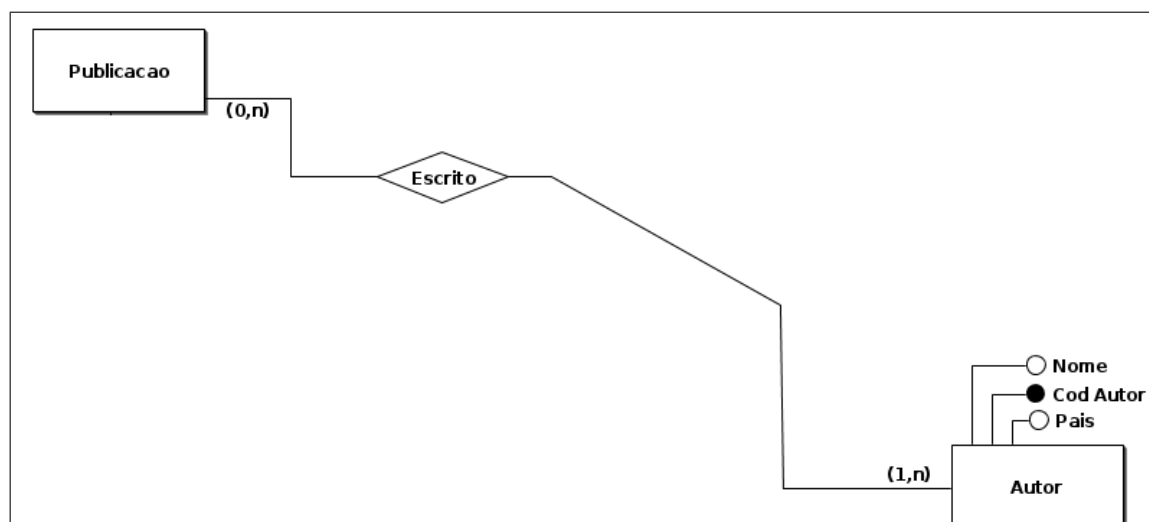
Figura 7 – Entidade Funcionário



Fonte: Arquivo pessoal, 2024.

A entidade Autor ilustrada na Figura 8 abaixo, possui os atributos: Cod Autor (atributo identificador), Nome e Pais (país) e se relaciona com a entidade Publicacao. Cada Autor escreveu zero ou mais publicações.

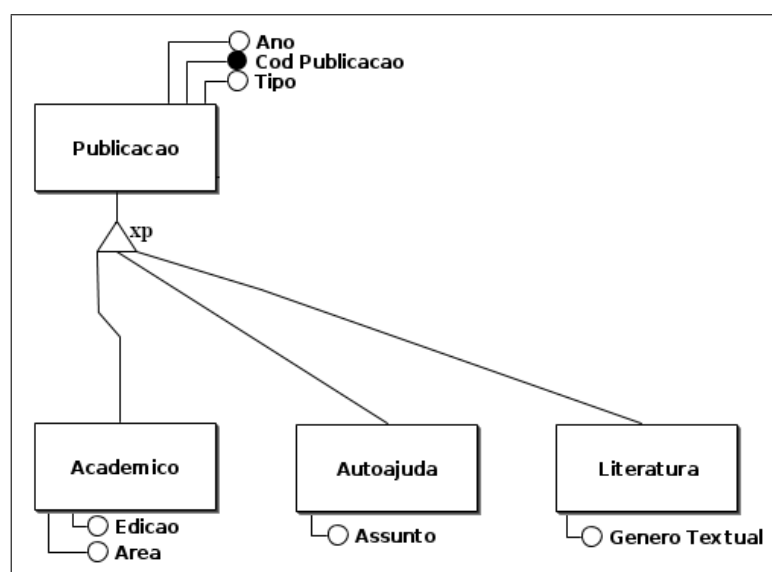
Figura 8 – Entidade Autor



Fonte: Arquivo pessoal, 2024.

A entidade Acadêmico possui os atributos específicos: Edicao e Area. A entidade Autoajuda possui o atributo específico: Assunto. A entidade Literatura possui o atributo específico: Genero Textual. Todas são generalizações da entidade Publicação e estão ilustradas na Figura 9 abaixo. As especializações são do tipo *xp* que indica que são exclusivas, pode ser generalizado em somente uma categoria, e parcial, pode ser generalizado em outro tipo que não esteja indicado e será armazenado no atributo Tipo da entidade Publicacao.

Figura 9 – Entidades das especializações de Publicação

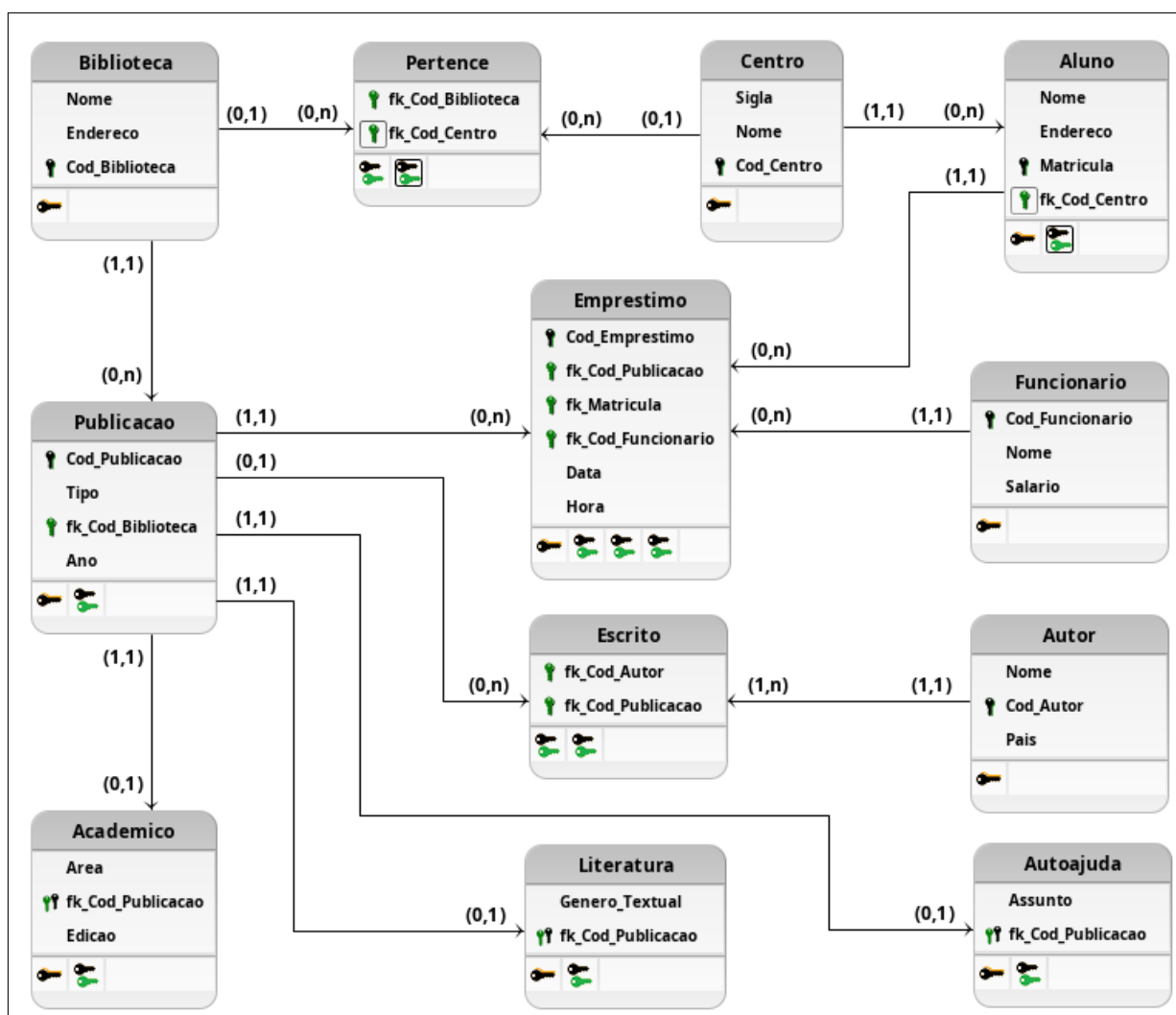


Fonte: Arquivo pessoal, 2024.

4.2 Modelo Lógico

Para a realização do modelo lógico, foram seguidas as regras de transformações entre modelos, sendo três regras principais de três diferentes relacionamentos que devem ser seguidas. No caso deste trabalho, apenas as regras de transformação 1:N e N:N foram necessárias. Conforme a regra N:N, *Pertence* (entre *Biblioteca* e *Centro*) virou uma tabela própria, possuindo duas chaves estrangeiras, sendo elas *fk_Cod_Biblioteca* e *fk_Cod_Centro*, usadas para demonstrar de maneira mais lógica quais atributos com identificadores únicos interagem entre si para realizar suas devidas funções. O mesmo ocorre com a criação de uma nova tabela própria chamada *Escrito*, onde pode-se perceber o surgimento de duas chaves estrangeiras, conforme pode-se analisar na Figura 10.

Figura 10 – Modelo Lógico



Fonte: Arquivo pessoal, 2024.

Além disso, foi implementando adição de colunas, entre relações (x,n) para $(x,1)$ onde x segue a mesma lógica, podendo ser n , 0 ou 1, não tem diferença pois o que interessa no modelo lógico é apenas o último valor, com exceção nos relacionamentos 1:1, onde a cardinalidade mínima importa. Neste caso, mostra-se de extrema importância seguir as regras de transformação para relacionamentos 1:N, no qual não se cria nenhuma tabela nova, apenas adiciona-se a chave estrangeira no lado N, vale ressaltar que a chave estrangeira é sempre a chave primária de outra tabela. Essa adição de colunas funciona de maneira diferente da anterior, neste caso, a relação não se torna uma nova entidade, apenas a chave primária sendo copiada para dentro da entidade que possui (x,n) como relação. Nesse caso, *Empréstimo* recebeu a chave estrangeira de *Funcionário*, *Aluno* e *Publicação*, conforme as regras.

Como “*Publicação*” tem uma generalização, não é realizada a fusão, mesmo o relacionamento entre estes sendo $(x,1)$ para $(x,1)$. Isso se deve ao fato de que, como cada tipo de publicação tem atributos diferentes que não são identificadores únicos, a fusão resultaria em uma confusão onde não teria modo de distinguir qual dos atributos são de um determinado tipo.

No caso deste trabalho, como dito anteriormente, possui-se apenas relações N:N e 1:N. As duas relações N:N foram explicadas anteriormente, entretanto, para as relações 1:N, torna-se dispensável a explicação individual de cada transformação, pois todas seguem a mesma regra. Portanto, a fim de evitar redundâncias, foi explicado a criação da tabela *Empréstimo*, cuja ideia segue-se a mesma para todos os outros relacionamentos 1:N.

4.3 Modelo Físico

Após a concepção do modelo lógico passou-se para o modelo físico e implementação das tabelas escritas em SQL. Foi utilizado o SGDB MariaDB pois ser o substituto do MySQL que foi adquirido pela Oracle.

Abaixo, na Figura 11, o código de criação da tabela Biblioteca. Foi criado com uma chave primária chamada Cod_biblioteca como inteiro e os atributos Endereco e Nome como array de caracteres, denotado por "VARCHAR", com 100 e 50 caracteres respectivamente.

Figura 11 – Criação da tabela "Biblioteca" em SQL.

```
CREATE TABLE Biblioteca (
  Cod_Biblioteca INT PRIMARY KEY,
  Endereco VARCHAR(100),
  Nome VARCHAR(50)
);
```

Fonte: Arquivo pessoal, 2024.

A tabela Centro foi criada como indicado na Figura 12 abaixo. Foi criada a chave primária Cod_Centro como INT e os atributos Sigla como um VARCHAR de 10 caracteres e Nome como um VARCHAR de 50 caracteres.

Figura 12 – Criação da tabela "Centro" em SQL.

```
CREATE TABLE Centro (
  Cod_Centro INT PRIMARY KEY,
  Sigla VARCHAR(10),
  Nome VARCHAR(50)
);
```

Fonte: Arquivo pessoal, 2024.

Como o relacionamento entre as entidades Biblioteca e Centro é N:N isso infere na criação de uma nova tabela chamada Pertence. A criação dessa tabela está informada na Figura 13 abaixo. Ela contém somente a chave primária formada pela composição das chaves estrangeiras fk_Cod_Biblioteca, que referencia Biblioteca, e fk_Cod_Centro, que referencia Centro.

Figura 13 – Criação da tabela "Pertence" em SQL.

```
CREATE TABLE Pertence (
  fk_Cod_Biblioteca INT,
  fk_Cod_Centro INT,
  FOREIGN KEY (fk_Cod_Biblioteca) REFERENCES Biblioteca(Cod_Biblioteca),
  FOREIGN KEY (fk_Cod_Centro) REFERENCES Centro(Cod_Centro)
);
```

Fonte: Arquivo pessoal, 2024.

Na Figura 14 abaixo está informada a criação da tabela Aluno. Tem uma chave primária chamada Matricula, criada como INT, e os atributos Endereco e Nome como VARCHAR de 100 e 50 caracteres respectivamente, além de uma chave estrangeira chamada fk_Cod_Centro criada como INT, que referencia Centro para implementar o relacionamento 1:N que ocorre por adição de coluna de Centro em Aluno.

Figura 14 – Criação da tabela "Aluno" em SQL.

```
CREATE TABLE Aluno (  
    Matricula INT PRIMARY KEY,  
    Endereco VARCHAR(100),  
    Nome VARCHAR(50),  
    fk_Cod_Centro INT,  
    FOREIGN KEY (fk_Cod_Centro) REFERENCES Centro(Cod_centro)  
);
```

Fonte: Arquivo pessoal, 2024.

A criação da tabela Funcionario está indicada na Figura 15 abaixo. Conta com uma chave primária chamada Cod_Funcionario como INT, e com os atributos Salario como DECIMAL de 10 dígitos, sendo 2 para as casas decimais, e Nome como VARCHAR de 50 caracteres.

Figura 15 – Criação da tabela "Funcionario" em SQL.

```
CREATE TABLE Funcionario (  
    Cod_Funcionario INT PRIMARY KEY,  
    Salario DECIMAL(10,2),  
    Nome VARCHAR(50)  
);
```

Fonte: Arquivo pessoal, 2024.

A Figura 16 abaixo indica a criação da tabela Publicacao. Conta com a chave primária Cod_Publicacao como INT, o atributo opcional Tipo como VARCHAR de 50 caracteres e o atributo Ano como DATE, além da chave estrangeira fk_Cod_Biblioteca, criada como INT que referencia Biblioteca para implementar o relacionamento 1:N entre Publicacao e Biblioteca por adição de coluna.

Figura 16 – Criação da tabela "Publicacao" em SQL.

```
CREATE TABLE Publicacao (  
    Cod_Publicacao INT PRIMARY KEY,  
    Tipo VARCHAR(50) NULL,  
    Ano DATE,  
    fk_Cod_Biblioteca INT,  
    FOREIGN KEY (fk_Cod_Biblioteca) REFERENCES Biblioteca(Cod_Biblioteca)  
);
```

Fonte: Arquivo pessoal, 2024.

O atributo Tipo é opcional utilizando a palavra reservada NULL pois a generalização/especialização é parcial, portanto esse atributo é utilizado caso não se especialize nos tipos pré-definidos. Para as especializações foram implementadas tabelas próprias para cada uma. Abaixo na Figura 17 está indicada a criação da tabela Academico, que conta com os atributos Edicao como INT e Area com VARCHAR de 50 caracteres, além da chave estrangeira fk_Cod_Publicacao que referencia Publicacao.

Figura 17 – Criação da tabela "Academico"em SQL.

```
CREATE TABLE Academico (  
    Edicao INT,  
    Area VARCHAR(50),  
    fk_Cod_Publicacao INT,  
    FOREIGN KEY (fk_Cod_Publicacao) REFERENCES Publicacao(Cod_Publicacao)  
);
```

Fonte: Arquivo pessoal, 2024.

Além da especialização Academico também há a Literatura que sua criação está indicada na Figura 18 abaixo. Ela conta com o atributo Genero_Textual que é um VARCHAR de 50 caracteres além da chave estrangeira fk_Cod_Publicacao que referencia Publicacao.

Figura 18 – Criação da tabela "Literatura"em SQL.

```
CREATE TABLE Literatura (  
    Genero_Textual VARCHAR(50),  
    fk_Cod_Publicacao INT,  
    FOREIGN KEY (fk_Cod_Publicacao) REFERENCES Publicacao(Cod_Publicacao)  
);
```

Fonte: Arquivo pessoal, 2024.

Ainda há a especialização Autoajuda que tem sua criação indicada na Figura 19 abaixo. Ela conta com o atributo Assunto como VARCHAR de 50 caracteres e a chave estrangeira fk_Cod_Publicacao que referencia Publicacao.

Figura 19 – Criação da tabela "Autoajuda"em SQL.

```
CREATE TABLE Autoajuda (  
    Assunto VARCHAR(50),  
    fk_Cod_Publicacao INT,  
    FOREIGN KEY (fk_Cod_Publicacao) REFERENCES Publicacao(Cod_Publicacao)  
);
```

Fonte: Arquivo pessoal, 2024.

Abaixo, na Figura 20 está indicada a criação da tabela Autor. Tem uma chave primária nomeada Cod_Autor como INT e os atributos Nome e Pais, ambos como VARCHAR de 50 caracteres.

Figura 20 – Criação da tabela "Autor" em SQL.

```
CREATE TABLE Autor (  
    Cod_Autor INT PRIMARY KEY,  
    Nome VARCHAR(50),  
    Pais VARCHAR(50)  
);
```

Fonte: Arquivo pessoal, 2024.

O relacionamento entre Publicacao e Autor também é N:N o que obriga a criação de uma tabela própria para sua implementação. A criação da tabela Escrito está indicada abaixo na Figura 21. Ela tem somente as chaves estrangeiras das duas entidades; fk_Cod_Publicacao que referencia Publicacao e fk_Cod_Autor que referencia Autor, ambas INT.

Figura 21 – Criação da tabela "Escrito" em SQL.

```
CREATE TABLE Escrito (  
    fk_Cod_Publicacao INT,  
    fk_Cod_Autor INT,  
    FOREIGN KEY (fk_Cod_Publicacao) REFERENCES Publicacao(Cod_Publicacao),  
    FOREIGN KEY (fk_Cod_Autor) REFERENCES Autor(Cod_Autor)  
);
```

Fonte: Arquivo pessoal, 2024.

Por fim, na Figura 22 abaixo o código de criação da tabela Emprestimo. Ela conta com a chave primária Cod_Emprestimo, criada como INT, os atributos Data_, criado como DATE e com o "_" para diferenciá-lo da palavra reservada DATA, Hora criada como TIME e também as chaves estrangeiras fk_Cod_Publicacao que referencia Publicacao, fk_Matricula que referencia Aluno e fk_Cod_Funcionario que referencia Funcionario, todos criados como INT com o intuito de implementar por adição de coluna os relacionamentos 1:N com as entidades Publicacao, Aluno e Funcionario.

Figura 22 – Criação da tabela "Emprestimo" em SQL.

```
CREATE TABLE Emprestimo (  
  Cod_Emprestimo INT PRIMARY KEY,  
  Data_ DATE,  
  Hora TIME,  
  fk_Cod_Publicacao INT,  
  fk_Matricula int,  
  fk_Cod_Funcionario INT,  
  FOREIGN KEY (fk_Cod_Publicacao) REFERENCES Publicacao(Cod_Publicacao),  
  FOREIGN KEY (fk_Matricula) REFERENCES Aluno(Matricula),  
  FOREIGN KEY (fk_Cod_Funcionario) REFERENCES Funcionario(Cod_Funcionario)  
);
```

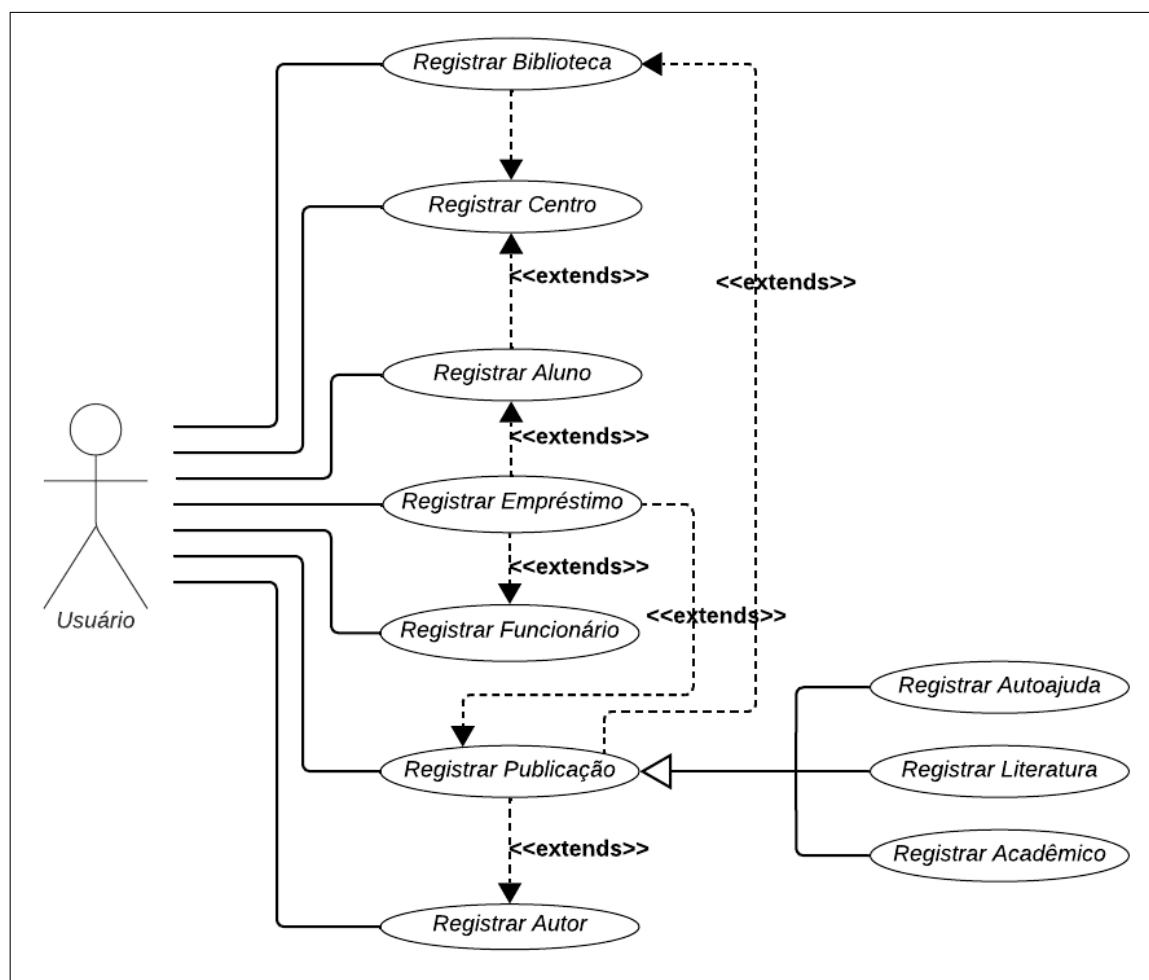
Fonte: Arquivo pessoal, 2024.

Com isso, todas as tabelas do modelo lógico indicado na Figura 10 foram implementadas em SQL. Outras aplicações em SQL, como inserções, remoções e buscar serão abordadas na Seção 4.4 referente à criação da interface gráfica que foi onde surgiram as necessidades de implementar essas aplicações.

4.4 Interface Gráfica e Integração com o Banco de Dados

Para auxiliar na criação da interface gráfica para integrar com o banco de dados foi desenvolvido o diagrama de casos de uso, indicado na Figura 23 abaixo, baseado no diagrama lógico do banco de dados.

Figura 23 – Diagrama de casos de uso da interface gráfica.



Fonte: Arquivo pessoal, 2024.

5 Conclusão

A utilização de um banco de dados para o gerenciamento de grandes volumes de dados é fundamental, pois permite uma maior organização, estruturação, segurança e escalabilidade informacional. O tema do trabalho (produção de um banco de dados para o gerenciamento de bibliotecas universitárias) é um exemplo extremamente válido da importância do uso de um banco de dados, pois uma biblioteca possui uma enorme variedade de livros que podem ser emprestados diversas vezes por vários alunos. A partir do desenvolvimento dos Modelos Lógico e Conceitual, os alunos foram capazes de perceber essa importância e consolidar o aprendizado no desenvolvimento de banco de dados adquirido ao longo da disciplina.

Ademais, o trabalho ajudou a aprimorar habilidades técnicas, como o uso de ferramentas de modelagem de diagramas, a escrita de um relatório seguindo as normas da ABNT e o entendimento da estruturação de banco de dados. O projeto também incentivou o desenvolvimento de habilidades interpessoais, pois dependeu do trabalho em equipe, comunicação de ideias entre os membros do grupo e por fim integração do material desenvolvido.

Dessa forma, o objetivo principal do trabalho (aplicar as técnicas de modelagem conceitual de bancos de dados relacionais e converter, de acordo com as regras especificadas, para um modelo lógico) foi totalmente atingido preparando os alunos para futuros desafios relacionados à estruturação de banco de dados que podem surgir nas suas devidas carreiras de trabalho.

Referências Bibliográficas

HEUSER, Carlos Alberto. **Projeto de banco de dados**. 6. ed. Porto Alegre: Bookman, 2009.

SCHULTZ, Vinícius et al. **grupo7-CDI-20242-tg2**. 2024. Disponível em: <<https://github.com/VenixBR/grupo7-CDI-20242-tg2/tree/main>>.