



UNIVERSIDADE PITÁGORAS UNOPAR

Engenharia de Software

Modelagem de Dados

Banco de Dados

Cidade de Serrana, SP

2025

AUTOR: ALLAS MAYCON DO VALLE

RA:3446927007

Orientador:

Tutor à Distância: Frederico Aparecido Faedo Pinto.

Prof. Murilo Caminotto Barbosa

**Modelagem de Banco de Dados: Sistema de Empréstimos da
Biblioteca Universitária**

Faculdade Anhanguera, Ribeirão Preto, SP.

Cidade de Serrana, SP

2025

SUMÁRIO

1 INTRODUÇÃO.....	4
2 MODELAGEM E DESENVOLVIMENTO.....	5
2.1 DEFINIÇÃO DAS ENTIDADES E ATRIBUTOS.....	5
2.2 ESTABELECIMENTO DE RELACIONAMENTOS.....	5
2.3 USO DO MYSQL WORKBENCH E ENGENHARIA DIRETA.....	6
3 DIAGRAMA ENTIDADE-RELACIONAMENTO (DER).....	7
4 SCRIPT SQL DE CRIAÇÃO.....	8
5 CONCLUSÃO.....	11
REFERÊNCIAS.....	13

1. INTRODUÇÃO

O presente trabalho visa apresentar o desenvolvimento do Modelo Entidade-Relacionamento (MER) para um sistema de controle de empréstimos de uma biblioteca universitária, conforme os requisitos estabelecidos na proposta da atividade. A modelagem de dados é uma etapa crucial no ciclo de vida do desenvolvimento de sistemas, pois define a estrutura lógica e as regras de integridade do banco de dados que dará suporte à aplicação.

O objetivo primário deste projeto é criar um Diagrama Entidade-Relacionamento (DER) robusto, que mapeie de forma clara as entidades envolvidas no processo de empréstimo e as relações entre elas. Para tal, foi utilizada a ferramenta MySQL Workbench, que permite a conversão do modelo lógico para o modelo relacional de maneira eficiente.

A análise da situação proposta identificou quatro entidades centrais: **Aluno**, **Livro**, **Colaborador** e **Empréstimo**. A correta estruturação dessas entidades exige a definição precisa de atributos, tipos de dados, chaves primárias (PK) e chaves estrangeiras (FK), bem como a aplicação de restrições como Not Null (NN) e Unique (UQ).

A tabela **Empréstimo** atua como a entidade de ligação, conectando as demais através de três relacionamentos de cardinalidade Um para Muitos (1:N). Essa ligação garante que toda transação registrada seja rastreável, indicando o aluno que pegou a obra, o livro emprestado e o colaborador responsável pela ação.

O DER final, elaborado e validado no MySQL Workbench, cumpre o requisito de ser um modelo lógico íntegro. O modelo está apto a ser convertido em um script SQL para a criação imediata das tabelas e seus respectivos relacionamentos em um SGBD MySQL. Este documento demonstra a conclusão bem-sucedida do projeto de modelagem de dados para o sistema da biblioteca.

2. MODELAGEM E DESENVOLVIMENTO

O desenvolvimento do Diagrama Entidade-Relacionamento (DER) para o sistema de biblioteca seguiu o processo de modelagem lógica, utilizando a ferramenta **MySQL Workbench**. O objetivo desta fase foi traduzir as entidades e regras de negócio propostas em uma estrutura relacional implementável.

2.1. Definição das Entidades e Atributos

Foram identificadas e criadas as quatro entidades centrais do sistema, definindo suas chaves primárias e aplicando as regras de integridade de dados:

- **Aluno:** Utilizou-se o atributo ra (VARCHAR) como Chave Primária (PK). Os campos nome e email foram definidos como obrigatórios (NOT NULL), e email recebeu a restrição de unicidade (UNIQUE).
- **Livro:** O isbn (VARCHAR) foi definido como PK, sendo um identificador único de obra.
- **Colaborador:** O cpf (VARCHAR) foi definido como PK. Similar ao Aluno, o campo email foi configurado como obrigatório e único.
- **Empréstimo:** Esta tabela recebeu um id (INT) como PK e Auto Incremento (AI) para servir como identificador da transação.

2.2. Estabelecimento de Relacionamentos

A tabela **Empréstimo** atua como a entidade de ligação (associativa), registrando a transação de empréstimo. Utilizando a ferramenta *1:N Non-Identifying Relationship* do Workbench, foram estabelecidos três relacionamentos Um para Muitos (1:N):

- Aluno (1) \rightarrow Empréstimo (N)
- Livro (1) \rightarrow Empréstimo (N)
- Colaborador (1) \rightarrow Empréstimo (N)
-

Este processo gerou automaticamente as Chaves Estrangeiras (FKs) na tabela Empréstimo (Aluno_ra, Livro_isbn, Colaborador_cpf), garantindo que um empréstimo não possa ser registrado sem um Aluno, Livro e Colaborador válidos.

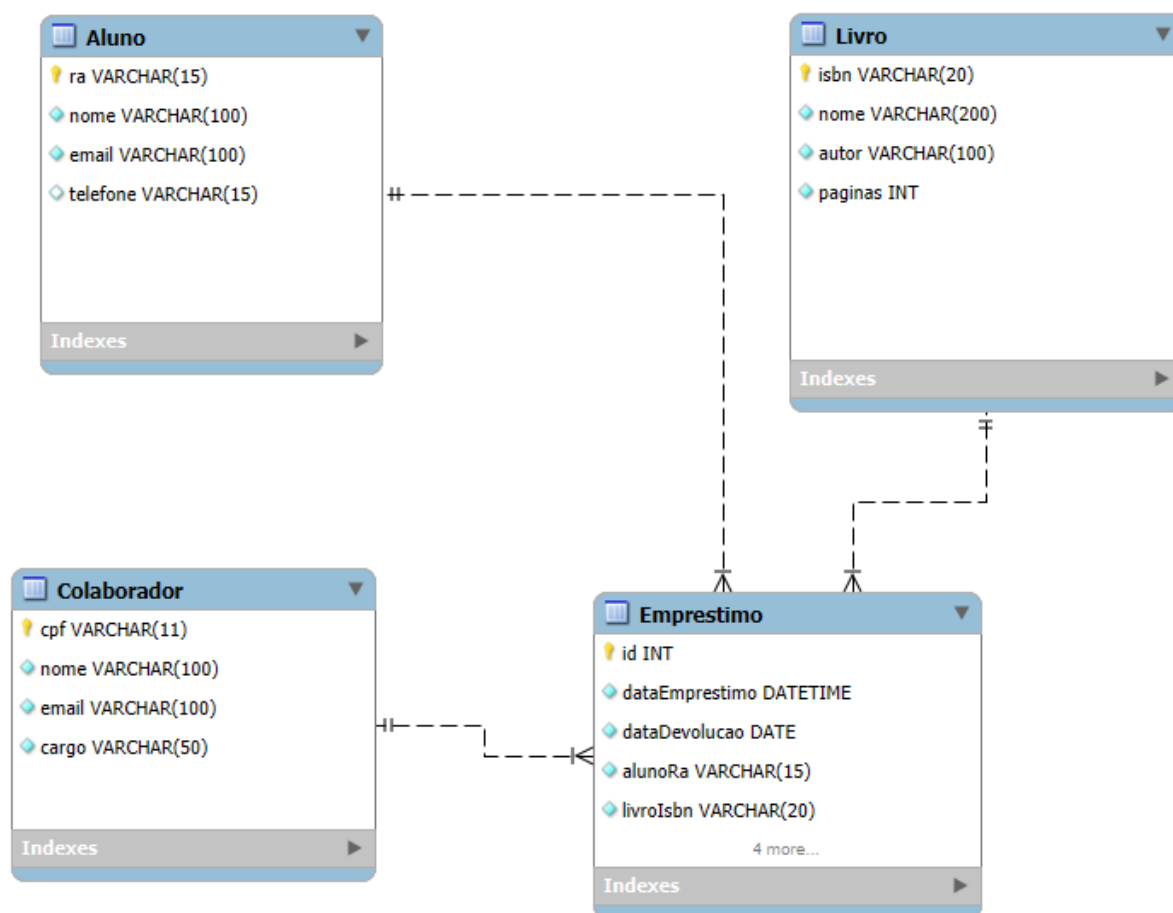
2.3. Uso do MySQL Workbench e Engenharia Direta

O Workbench facilitou a visualização do diagrama no modo **EER Diagram**. Após a validação visual do modelo, a funcionalidade **Forward Engineer** (Engenharia Direta) foi utilizada para exportar o modelo para o código SQL de criação, comprovando que o DER está apto para a implementação física em um SGBD MySQL.

3. DIAGRAMA ENTIDADE-RELACIONAMENTO (DER)

O propósito desta seção é apresentar o resultado gráfico da modelagem de dados realizada no MySQL Workbench, o qual representa a estrutura lógica do banco de dados da biblioteca.

A imagem a seguir exibe o Diagrama Entidade-Relacionamento (DER) final, contendo as quatro entidades (tabelas) e as relações de cardinalidade Um-para-Muitos (1:N) que conectam as transações de empréstimo aos registros de Aluno, Livro e Colaborador, validando a integridade referencial do sistema.



Fonte: Elaboração Própria, Fonte Autor: Autoria Própria.

4. SCRIPT SQL DE CRIAÇÃO

O **Script SQL de Criação** (Data Definition Language - DDL) é o código gerado pelo recurso *Forward Engineer* do MySQL Workbench. Ele representa a tradução fiel do Diagrama Entidade-Relacionamento (DER) para comandos que podem ser executados em um Sistema Gerenciador de Banco de Dados (SGBD) MySQL.

Este script é a comprovação técnica de que a modelagem realizada é implementável, contendo os comandos para:

Criar o esquema (CREATE SCHEMA).

Criar as tabelas (CREATE TABLE) com a definição correta dos atributos e tipos de dados.

Estabelecer as Chaves Primárias (PRIMARY KEY) e as restrições de unicidade (UNIQUE).

Configurar as Chaves Estrangeiras (FOREIGN KEY e CONSTRAINT) para garantir a integridade referencial entre as entidades, conforme o modelo 1:N desenhado.

O código gerado é apresentado a seguir:

SQL

```
-- MySQL Workbench Forward Engineering
SET @OLD_UNIQUE_CHECKS=@@UNIQUE_CHECKS, UNIQUE_CHECKS=0;
SET @OLD_FOREIGN_KEY_CHECKS=@@FOREIGN_KEY_CHECKS,
FOREIGN_KEY_CHECKS=0;
SET @OLD_SQL_MODE=@@SQL_MODE,
SQL_MODE='ONLY_FULL_GROUP_BY,STRICT_TRANS_TABLES,NO_ZERO_IN_DATE,NO_ZERO
_DATE,ERROR_FOR_DIVISION_BY_ZERO,NO_ENGINE_SUBSTITUTION';

-----
-- Schema mydb
-----
CREATE SCHEMA IF NOT EXISTS `mydb` DEFAULT CHARACTER SET utf8 ;
USE `mydb` ;

-----
-- Table `mydb`.`Aluno`
-----
DROP TABLE IF EXISTS `mydb`.`Aluno` ;

CREATE TABLE IF NOT EXISTS `mydb`.`Aluno` (
  `ra` VARCHAR(15) NOT NULL,
  `nome` VARCHAR(100) NOT NULL,
  `email` VARCHAR(100) NOT NULL,
  `telefone` VARCHAR(15) NULL,
  PRIMARY KEY (`ra`),
  UNIQUE INDEX `idx_aluno_email_unique` (`email` ASC) VISIBLE)
ENGINE = InnoDB;
```



```

-----
-- Table `mydb`.`Livro`
-----
DROP TABLE IF EXISTS `mydb`.`Livro` ;

CREATE TABLE IF NOT EXISTS `mydb`.`Livro` (
  `isbn` VARCHAR(20) NOT NULL,
  `nome` VARCHAR(200) NOT NULL,
  `autor` VARCHAR(100) NOT NULL,
  `paginas` INT NOT NULL,
  PRIMARY KEY (`isbn`))
ENGINE = InnoDB;

-----
-- Table `mydb`.`Colaborador`
-----
DROP TABLE IF EXISTS `mydb`.`Colaborador` ;

CREATE TABLE IF NOT EXISTS `mydb`.`Colaborador` (
  `cpf` VARCHAR(11) NOT NULL,
  `nome` VARCHAR(100) NOT NULL,
  `email` VARCHAR(100) NOT NULL,
  `cargo` VARCHAR(50) NOT NULL,
  PRIMARY KEY (`cpf`),
  UNIQUE INDEX `email_UNIQUE` (`email` ASC) VISIBLE)
ENGINE = InnoDB;

-----
-- Table `mydb`.`Emprestimo`
-----
DROP TABLE IF EXISTS `mydb`.`Emprestimo` ;

CREATE TABLE IF NOT EXISTS `mydb`.`Emprestimo` (
  `id` INT NOT NULL AUTO_INCREMENT,
  `dataEmprestimo` DATETIME NOT NULL,
  `dataDevolucao` DATE NOT NULL,
  `alunoRa` VARCHAR(15) NOT NULL,
  `livroIsbn` VARCHAR(20) NOT NULL,
  `colaboradorCof` VARCHAR(11) NOT NULL,
  `Aluno_ra` VARCHAR(15) NOT NULL,
  `Colaborador_cpf` VARCHAR(11) NOT NULL,
  `Livro_isbn` VARCHAR(20) NOT NULL,
  PRIMARY KEY (`id`),
  INDEX `fk_Emprestimo_Aluno_idx` (`Aluno_ra` ASC) VISIBLE,
  INDEX `fk_Emprestimo_Colaborador1_idx` (`Colaborador_cpf` ASC) VISIBLE,
  INDEX `fk_Emprestimo_Livro1_idx` (`Livro_isbn` ASC) VISIBLE,
  CONSTRAINT `fk_Emprestimo_Aluno`
    FOREIGN KEY (`Aluno_ra`)
    REFERENCES `mydb`.`Aluno` (`ra`)
    ON DELETE NO ACTION
    ON UPDATE NO ACTION,
  CONSTRAINT `fk_Emprestimo_Colaborador1`
    FOREIGN KEY (`Colaborador_cpf`)
    REFERENCES `mydb`.`Colaborador` (`cpf`)
    ON DELETE NO ACTION
    ON UPDATE NO ACTION,
  CONSTRAINT `fk_Emprestimo_Livro1`

```

```
FOREIGN KEY (`Livro_isbn`)  
REFERENCES `mydb`.`Livro` (`isbn`)  
ON DELETE NO ACTION  
ON UPDATE NO ACTION)  
ENGINE = InnoDB;
```

```
SET SQL_MODE=@OLD_SQL_MODE;  
SET FOREIGN_KEY_CHECKS=@OLD_FOREIGN_KEY_CHECKS;  
SET UNIQUE_CHECKS=@OLD_UNIQUE_CHECKS;
```

5 CONCLUSÃO

O presente projeto de modelagem atingiu o objetivo de desenvolver um Diagrama Entidade-Relacionamento (DER) funcional para o sistema de empréstimos da biblioteca universitária. Utilizando o MySQL Workbench, foi possível:

- **Representar de forma clara** as quatro entidades essenciais (Aluno, Livro, Colaborador e Empréstimo).
- **Definir corretamente** as Chaves Primárias (PK) e aplicar as restrições de unicidade (UNIQUE) aos atributos.
- **Estabelecer os relacionamentos** de cardinalidade Um-para-Muitos (1:N) por meio das Chaves Estrangeiras (FKs) na tabela **Empréstimo**, garantindo a integridade referencial.

A geração do Script SQL através da funcionalidade *Forward Engineer* confirmou a **viabilidade e implementabilidade** do modelo lógico. O resultado é uma estrutura de banco de dados robusta e coerente com as regras de negócio propostas, pronta para ser implementada em um ambiente de produção.

REFERÊNCIAS

ELMASRI, Ramez; NAVATHE, Shamkant B. *Sistemas de Banco de Dados*. 7. ed. São Paulo: Pearson Education do Brasil, 2018. (Cobre Modelagem E-R e SQL, sendo um clássico da área).

DATE, C. J. *Introdução a Sistemas de Bancos de Dados*. 8. ed. Rio de Janeiro: Elsevier (Campus), 2004. (Obra fundamental sobre teoria relacional e integridade de dados).

HEUSER, Carlos Alberto. *Projeto de Banco de Dados: Modelagem E Implementação*. 7. ed. Porto Alegre: Bookman, 2014. (Focado diretamente no processo de projeto e implementação, incluindo DER e SQL).

ATIVIDADE DE PORTFÓLIO/INTERDISCIPLINAR

As atividades possuem material exclusivo do portfólio que se encontra no GitHub para download.

https://github.com/allas-amk/modelagem_de_banco_de_dados_sql_portfolio_faculdade