
****INSTITUTO FEDERAL DE EDUCAÇÃO, CIÊNCIA E TECNOLOGIA DE SÃO PAULO****

****PEDRO LUIZ DA SILVA TRINDADE MAGALHÃES****

****SISTEMA DE GERENCIAMENTO DE CINEMA****

****CAMPOS DO JORDÃO****

****2025****

****TERMO DE APROVAÇÃO****

****PROJETO DE SISTEMA DE GERENCIAMENTO DE CINEMA****

****PEDRO LUIZ DA SILVA TRINDADE MAGALHÃES****

Projeto apresentado à disciplina Banco de Dados I do curso de Análise e Desenvolvimento de Sistemas do IFSP Campus Campos do Jordão, como requisito parcial para avaliação.

Aprovado em: ____/____/____

****PROFESSOR PAULO****

****RESUMO****

Este trabalho apresenta o projeto e o modelo de dados para um Sistema de Gerenciamento de Cinema, com o propósito de automatizar as principais operações de bilheteria e gestão. O objetivo principal consistiu em modelar e documentar o sistema, detalhando requisitos, a metodologia de desenvolvimento, o projeto de dados (modelo conceitual e dicionário de dados), casos de uso, arquitetura proposta e sugestões de implementação. Para o desenvolvimento do modelo de dados, utilizou-se a metodologia que combinou levantamento de requisitos, a modelagem conceitual (Diagrama Entidade-Relacionamento com notação Chen) e a geração de scripts SQL iniciais exemplificativos. Os resultados obtidos incluem um Modelo Conceitual completo, detalhamento das entidades, chaves e um Dicionário de Dados inicial. Conclui-se que o modelo atende às necessidades operacionais básicas de bilheteria, controle de sessões, gerenciamento de salas e reservas, garantindo a integridade dos dados e estabelecendo uma base sólida para a implementação futura do sistema.

****Palavras-chave:**** Cinema; Gerenciamento; Banco de Dados; Modelo Conceitual; Reservas.

****ABSTRACT****

This work presents the project and data model for a Cinema Management System, aimed at automating the main box office and management operations. The main objective was to model and document the system, detailing requirements, development methodology, data design (conceptual model and data dictionary), use cases, proposed architecture, and implementation suggestions. For the development of the data model, a methodology was used that combined requirements gathering, conceptual modeling (Entity-Relationship Diagram with Chen notation), and the generation of initial exemplary SQL scripts. The results obtained include a complete Conceptual Model, detailing of entities, keys, and an initial Data Dictionary. It is concluded that the model meets the basic operational needs of box office, session

control, room management, and reservations, ensuring data integrity and establishing a solid foundation for the future implementation of the system.

****Keywords:**** Cinema; Management; Database; Conceptual Model; Reservations.

****SUMÁRIO****

1. INTRODUÇÃO 1

1.1 Objetivos 1

1.2 Justificativa 2

1.3 Aspectos Metodológicos 2

1.4 Aporte Teórico 3

2. FUNDAMENTAÇÃO TEÓRICA 4

2.1 Modelo Entidade-Relacionamento (MER) 4

2.2 Normalização de Dados 4

2.3 Trabalhos Relacionados 5

3. PROJETO PROPOSTO (METODOLOGIA) 6

3.1 Considerações Iniciais 6

3.2 Requisitos 6

3.3 Projeto de Dados 7

3.4 Arquitetura 8

3.5 Interfaces 8

4. AVALIAÇÃO 9

4.1 Resultados Obtidos 9

4.2 Discussão 13

5. CONCLUSÃO 14

5.1 Sugestões de Melhorias (Trabalhos Futuros) 14

REFERÊNCIAS 15

APÊNDICES 16

****1. INTRODUÇÃO****

Neste capítulo serão introduzidos todos os assuntos abordados por este documento. Pretende-se apresentar a motivação, os objetivos e a organização do texto. O presente trabalho descreve o projeto de um Sistema de Gerenciamento de Cinema destinado a automatizar a venda de ingressos, o controle de salas e sessões, as reservas e a geração de relatórios gerenciais. O cenário abordado é o gerenciamento das operações de um cinema padrão.

****1.1 Objetivos****

Este trabalho tem por objetivo principal modelar e documentar o Sistema de Gerenciamento de Cinema a partir da perspectiva de Banco de Dados I.

Para a consecução deste objetivo foram estabelecidos os objetivos específicos:

- Levantar requisitos funcionais e não funcionais inerentes à operação de um cinema.
- Propor um Modelo Conceitual (MER) que represente as entidades e relacionamentos principais (Filme, Sala, Sessão, Ingresso, etc.).

- Produzir o projeto lógico (tabelas, chaves, constraints) adequado para implementação em SGBD relacional.
- Apresentar casos de uso principais e fluxos de operação do sistema.
- Indicar tecnologias sugeridas para implementação e plano de implantação inicial.

****1.2 Justificativa****

A relevância do trabalho reside na necessidade de automatizar a gestão de cinemas para garantir a consistência das informações, agilizar processos como a venda de ingressos e o controle de lotação. A informatização reduz erros manuais, otimiza a ocupação das salas, facilita o controle financeiro e a geração de relatórios, sendo crucial para a saúde operacional e financeira da empresa. A elaboração deste sistema visa oferecer uma solução robusta e escalável para o gerenciamento completo de um cinema.

****1.3 Aspectos Metodológicos****

O presente estudo fez uso das pesquisas de natureza bibliográfica, para o que remete à parte escrita, e de campo (simulação de entrevistas e levantamento de regras de negócio), para o que se refere à parte prática. A metodologia de desenvolvimento principal incluiu o levantamento de requisitos, a Modelagem Conceitual (utilizando a notação Chen), a geração do Projeto Lógico, a elaboração do Dicionário de Dados e a proposta de arquitetura do sistema.

****1.4 Aporte Teórico****

As bases teóricas que fundamentaram esta pesquisa concentram-se em conceitos de Sistemas Gerenciadores de Banco de Dados (SGBD), Modelagem de Dados (Modelagem Entidade-Relacionamento) e Normalização de Dados para garantir a integridade e evitar redundâncias no modelo. A notação adotada para o modelo conceitual foi o Diagrama Entidade-Relacionamento (Notação Chen).

****2. FUNDAMENTAÇÃO TEÓRICA****

Nesta seção será apresentada uma revisão de textos, artigos, livros e periódicos, pertinente à revisão da literatura que será utilizada no desenvolvimento do trabalho. A seguir, são apresentados conceitos fundamentais para a modelagem do sistema.

****2.1 Modelo Entidade-Relacionamento (MER)****

O Modelo Entidade-Relacionamento (MER) é uma ferramenta conceitual utilizada para representar graficamente a estrutura de dados de um sistema. Ele descreve as entidades (objetos do mundo real), seus atributos e os relacionamentos entre elas. Esta modelagem é essencial na fase de projeto de banco de dados, pois permite uma compreensão clara dos requisitos de dados antes da implementação física.

****2.2 Normalização de Dados****

A normalização de dados é um processo de organização de atributos em tabelas para reduzir a redundância e a dependência de dados. Aplicando as formas normais (1FN, 2FN, 3FN), garante-se a integridade do banco de dados do cinema, evitando anomalias de inserção, atualização e exclusão. Este processo é crucial para um sistema de gerenciamento que requer alta confiabilidade nas transações.

****2.3 Trabalhos Relacionados****

Existem diversos sistemas comerciais de gerenciamento de cinema (como o Veezi, MovieTickets, etc.) que oferecem funcionalidades semelhantes. O diferencial deste projeto está no foco acadêmico e na construção detalhada do modelo de dados como objeto de estudo da disciplina Banco de Dados I, priorizando a correta modelagem conceitual e lógica sobre a implementação de todas as funcionalidades comerciais.

****3. PROJETO PROPOSTO (METODOLOGIA)****

Nesta seção serão apresentadas detalhadamente a metodologia utilizada neste trabalho, e os documentos referentes ao sistema proposto, como eles foram elaborados e demais artefatos referentes a este projeto.

****3.1 Considerações Iniciais****

O escopo do sistema abrange o gerenciamento de filmes, sessões, salas, assentos, vendas e reservas, com foco na precisão do controle de ocupação e transações financeiras. Considera-se um cinema de porte pequeno a médio, com múltiplas salas e sessões diárias.

****3.2 Requisitos****

****3.2.1 Requisitos Funcionais (RF)****

- RF01: Cadastrar, consultar, atualizar e inativar filmes.
- RF02: Cadastrar salas e seus respectivos assentos.
- RF03: Agendar sessões, vinculando filme, sala, data e horário.
- RF04: Realizar venda de ingressos, com seleção específica de assento.
- RF05: Permitir reserva online de assentos com tempo limitado para confirmação.
- RF06: Cancelar/estornar ingressos vendidos.
- RF07: Gerar relatórios de vendas por período, ocupação por sessão e receita.

****3.2.2 Requisitos Não Funcionais (NFR)****

- NFR01: Segurança – Autenticação obrigatória para funcionários com diferentes níveis de acesso.
- NFR02: Desempenho – Tempo de resposta inferior a 2 segundos para operações de venda.
- NFR03: Confiabilidade – Sistema disponível 99% do horário de funcionamento do cinema.
- NFR04: Manutenibilidade – Backup diário completo dos dados.

****3.3 Projeto de Dados****

O projeto de dados é o foco desta disciplina e apresenta o detalhamento das estruturas de armazenamento das informações. A modelagem garante que as regras de negócio sejam implementadas corretamente na estrutura do banco.

****3.3.1 Modelo Conceitual (Descrição)****

O modelo é composto pelas seguintes entidades principais e seus relacionamentos:

- ****FILME**** – (1,N) – ****SESSAO****: Um filme pode ser exibido em várias sessões.
- ****SALA**** – (1,N) – ****ASSENTO****: Uma sala contém vários assentos.
- ****SALA**** – (1,N) – ****SESSAO****: Uma sala pode hospedar várias sessões (em horários distintos).
- ****SESSAO**** – (1,N) – ****INGRESSO****: Uma sessão pode ter vários ingressos vendidos.
- ****CLIENTE**** – (0,N) – ****INGRESSO/RESERVA****: Um cliente pode comprar vários ingressos ou fazer várias reservas.
- ****INGRESSO**** – (1,1) – ****ASSENTO**** (por sessão): Cada ingresso corresponde a um assento específico em uma sessão. (Relacionamento tratado via chave única composta).

****3.3.2 Regras de Negócio Principais****

- RN01: Um assento não pode ser vendido duas vezes para a mesma sessão.
- RN02: Sessões na mesma sala não podem ter sobreposição de horários.
- RN03: Reservas online expiram após 15 minutos sem confirmação de pagamento, liberando o assento.
- RN04: A venda de ingresso para um filme com classificação indicativa deve validar a idade do cliente.

****3.4 Arquitetura****

O sistema é proposto em uma arquitetura em três camadas:

1. ****Camada de Apresentação****: Interface Web para reservas online e interface Desktop (POS) para a bilheteria física.
2. ****Camada de Negócio (API/Backend)****: Responsável por processar as regras de negócio, autenticação e gerenciar a comunicação com o banco de dados.
3. ****Camada de Dados****: Banco de Dados Relacional (SGBD), onde reside o modelo físico implementado a partir do projeto lógico.

- ****SGBD Recomendado****: PostgreSQL ou MySQL.

- ****Tecnologias Sugeridas****: Backend em Node.js/Express ou Java Spring Boot; Frontend em React; Autenticação via JWT.

****3.5 Interfaces****

As interfaces principais propostas incluem:

- ****Tela de Agendamento de Sessões****: Formulário para seleção de filme, sala, data, horário e definição de preço.
- ****Mapa de Assentos****: Representação visual em grade da sala, com assentos coloridos conforme status (livre, reservado, vendido).
- ****Módulo de Venda (POS)****: Interface simplificada para seleção rápida de sessão, assento e registro de pagamento.
- ****Painel do Gerente****: Dashboard com gráficos e relatórios de vendas, ocupação e receita.

****4. AVALIAÇÃO****

Nesta seção serão apresentados os resultados deste trabalho e uma discussão sobre eles.

****4.1 Resultados Obtidos****

Nessa seção, os resultados são descritos de maneira objetiva, obedecendo uma sequência lógica usando texto e tabelas. O principal artefato do projeto é o Modelo Lógico derivado do MER, pronto para implementação.

****4.1.1 Modelo Lógico (Tabelas Principais)****

****Tabela 1: Estrutura da Tabela FILME****

Atributo	Tipo de Dado	Descrição	Restrição
id_filme (PK)	SERIAL / INTEGER	Identificador único do filme.	NOT NULL
titulo	VARCHAR(255)	Título original do filme.	NOT NULL
duracao_min	INTEGER	Duração em minutos.	NOT NULL
classificacao_indicativa	VARCHAR(10)	Classificação etária (L, 10, 12, 14, 16, 18).	
idioma	VARCHAR(50)	Idioma original.	
sinopse	TEXT	Sinopse do filme.	
diretor	VARCHAR(150)	Nome do diretor.	
genero_id (FK)	INTEGER	Referência ao gênero do filme.	
data_lancamento	DATE	Data de lançamento.	
ativo	BOOLEAN	Indica se o filme está ativo para programação.	DEFAULT TRUE

****Tabela 2: Estrutura da Tabela SALA****

Atributo	Tipo de Dado	Descrição	Restrição

id_sala (PK)	SERIAL / INTEGER	Identificador único da sala.	NOT NULL
nome_sala	VARCHAR(50)	Nome comercial (ex: "Sala 1").	NOT NULL
capacidade	INTEGER	Número total de assentos.	NOT NULL, >0
tipo	VARCHAR(20)	Tipo de exibição (CONVENCIONAL, 3D, IMAX).	
ativo	BOOLEAN	Status de uso da sala.	DEFAULT TRUE

****Tabela 3: Estrutura da Tabela ASSENTO****

Atributo	Tipo de Dado	Descrição	Restrição
-----	-----	-----	-----

id_assento (PK)	SERIAL / INTEGER	Identificador único do assento.	NOT NULL
id_sala (FK)	INTEGER	Referência à sala que contém o assento.	NOT NULL, REF: SALA(id_sala)
linha	VARCHAR(5)	Letra da linha (ex: 'A', 'B').	
numero	INTEGER	Número do assento na linha.	
codigo_assento	VARCHAR(10)	Código único por sala (ex: 'A12').	NOT NULL, UNIQUE (por sala)
tipo_assento	VARCHAR(20)	Tipo (normal, premium, deficiente).	

****Tabela 4: Estrutura da Tabela SESSAO****

Atributo	Tipo de Dado	Descrição	Restrição
-----	-----	-----	-----

id_sessao (PK)	SERIAL / INTEGER	Identificador único da sessão.	NOT NULL
id_filme (FK)	INTEGER	Referência ao filme da sessão.	NOT NULL, REF: FILME(id_filme)
id_sala (FK)	INTEGER	Referência à sala da sessão.	NOT NULL, REF: SALA(id_sala)
data_sessao	DATE	Data da exibição.	NOT NULL
hora_inicio	TIME	Horário de início.	NOT NULL
preco_base	NUMERIC(8,2)	Preço base do ingresso para esta sessão.	NOT NULL
formato	VARCHAR(20)	Formato de exibição (2D, 3D).	
ativo	BOOLEAN	Status da sessão.	DEFAULT TRUE

****Tabela 5: Estrutura da Tabela INGRESSO****

Atributo	Tipo de Dado	Descrição	Restrição
----- ----- ----- -----			

id_ingresso (PK)	SERIAL / INTEGER	Identificador único do ingresso.	NOT NULL
id_sessao (FK)	INTEGER	Referência à sessão.	NOT NULL, REF: SESSAO(id_sessao)
id_assento (FK)	INTEGER	Referência ao assento vendido.	NOT NULL, REF: ASSENTO(id_assento)
id_cliente (FK)	INTEGER	Referência ao cliente comprador (opcional).	REF: CLIENTE(id_cliente)
codigo_barra	VARCHAR(50)	Código único para validação na entrada.	UNIQUE

status	VARCHAR(20)	Status (VENDIDO, CANCELADO, UTILIZADO).	
DEFAULT 'VENDIDO'			
data_compra	TIMESTAMP	Data e hora da compra.	DEFAULT
NOW()			
valor_pago	NUMERIC(8,2)	Valor efetivamente pago.	NOT
NULL			
id_funcionario (FK)	INTEGER	Funcionário que realizou a venda (NULL para	
online).	REF: FUNCIONARIO(id_funcionario)		
CONSTRAINT UNIQUE (id_sessao, id_assento) – Garante que um assento não			
seja vendido duas vezes para a mesma sessão.			

****4.1.2 Script SQL Inicial (Exemplo - PostgreSQL)****

```
```sql
```

```
-- Criação das Tabelas Principais
```

```
CREATE TABLE filme (
 id_filme SERIAL PRIMARY KEY,
 titulo VARCHAR(255) NOT NULL,
 duracao_min INTEGER NOT NULL,
 classificacao_indicativa VARCHAR(10),
 idioma VARCHAR(50),
 sinopse TEXT,
 diretor VARCHAR(150),
 data_lancamento DATE,
 ativo BOOLEAN DEFAULT TRUE
);
```

```
CREATE TABLE sala (
 id_sala SERIAL PRIMARY KEY,
 nome_sala VARCHAR(50) NOT NULL,
```

```
 capacidade INTEGER NOT NULL CHECK (capacidade > 0),
 tipo VARCHAR(20),
 ativo BOOLEAN DEFAULT TRUE
);
```

```
CREATE TABLE assento (
 id_assento SERIAL PRIMARY KEY,
 id_sala INTEGER NOT NULL REFERENCES sala(id_sala),
 codigo_assento VARCHAR(10) NOT NULL,
 linha VARCHAR(5),
 numero INTEGER,
 tipo_assento VARCHAR(20),
 UNIQUE (id_sala, codigo_assento)
);
```

```
CREATE TABLE sessao (
 id_sessao SERIAL PRIMARY KEY,
 id_filme INTEGER NOT NULL REFERENCES filme(id_filme),
 id_sala INTEGER NOT NULL REFERENCES sala(id_sala),
 data_sessao DATE NOT NULL,
 hora_inicio TIME NOT NULL,
 preco_base NUMERIC(8,2) NOT NULL CHECK (preco_base >= 0),
 formato VARCHAR(20),
 ativo BOOLEAN DEFAULT TRUE
);
```

```
CREATE TABLE ingresso (
 id_ingresso SERIAL PRIMARY KEY,
```

```
id_sessao INTEGER NOT NULL REFERENCES sessao(id_sessao),
id_assento INTEGER NOT NULL REFERENCES assento(id_assento),
codigo_barra VARCHAR(50) UNIQUE,
status VARCHAR(20) DEFAULT 'VENDIDO',
data_compra TIMESTAMP DEFAULT NOW(),
valor_pago NUMERIC(8,2) NOT NULL CHECK (valor_pago >= 0),
UNIQUE (id_sessao, id_assento)
);
...
```

#### **\*\*4.2 Discussão\*\***

Os resultados mostram que o modelo de dados é robusto o suficiente para garantir a integridade das operações principais de um cinema. A restrição `UNIQUE (id\_sessao, id\_assento)` na tabela `INGRESSO` é fundamental para evitar a venda duplicada de um mesmo assento, implementando diretamente a RN01 no nível do banco de dados. A estrutura proposta separa claramente as entidades de negócio (Filme, Sala, Sessão) das transacionais (Ingresso), seguindo boas práticas de modelagem. A ausência de sobreposição de horários (RN02) pode ser garantida por uma trigger ou validação na camada de aplicação, uma vez que verificar este tipo de restrição complexa puramente com constraints SQL é mais desafiador. O modelo apresenta uma base sólida para a implementação do sistema completo.

#### **\*\*5. CONCLUSÃO\*\***

A “Conclusão” de um trabalho, seguindo as normas da ABNT, é onde será feita a conclusão final de tudo que foi comentado ao longo do trabalho e também dos objetivos.

O projeto do Sistema de Gerenciamento de Cinema resultou em um Modelo Entidade-Relacionamento (MER) e seu correspondente Modelo Lógico de alta fidelidade para os

requisitos operacionais de um cinema. O objetivo principal de propor um modelo de dados funcional e bem estruturado foi plenamente alcançado.

A síntese dos resultados do trabalho confirma que o modelo é capaz de suportar as operações críticas de bilheteria, reservas, controle de sessões e gerenciamento de salas, garantindo a integridade referencial e a consistência dos dados. As atividades de modelagem conceitual e lógica, que eram o foco da disciplina Banco de Dados I, foram integralmente cumpridas. O sistema está, portanto, pronto para a próxima fase de implementação da camada física em um SGBD específico.

#### **\*\*5.1 Sugestões de Melhorias (Trabalhos Futuros)\*\***

Como sugestões para a evolução do projeto e trabalhos futuros, podem-se considerar:

1. **\*\*Desenvolvimento do Modelo Físico e Implementação\*\***: Criar o banco de dados em um SGBD real (PostgreSQL/MySQL) e popular com dados de teste.
2. **\*\*Implementação da Camada de Aplicação\*\***: Desenvolver a API Backend e as interfaces de usuário (bilheteria e web) propostas.
3. **\*\*Módulo de Promoções e Fidelidade\*\***: Criar um sistema de pontos, cupons de desconto e preços diferenciados para sessões específicas.
4. **\*\*Controle de Bombonière\*\***: Integrar um módulo para gerenciar o estoque e vendas da bombonière do cinema.
5. **\*\*Dashboards Analíticos Avançados\*\***: Implementar relatórios preditivos e análise de dados para auxiliar na decisão de quais filmes exibir e em qual horário.

#### **\*\*REFERÊNCIAS\*\***

ELMASRI, R.; NAVATHE, S. B. **\*\*Sistemas de Banco de Dados\*\***. 7. ed. São Paulo: Pearson Education do Brasil, 2019.

HEUSER, C. A. **\*\*Projeto de Banco de Dados\*\***. 6. ed. Porto Alegre: Bookman, 2009.



POSTGRESQL GLOBAL DEVELOPMENT GROUP. **\*\*PostgreSQL 15 Documentation\*\***. 2023. Disponível em: <https://www.postgresql.org/docs/15/>. Acesso em: 05 dez. 2025.

SILBERSCHATZ, A.; KORTH, H. F.; SUDARSHAN, S. **\*\*Sistema de Banco de Dados\*\***. 6. ed. Rio de Janeiro: Elsevier, 2012.