

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

VINICIUS SOUSA LUCENA CHINAQUI

Sistema de Gestão de Academia

**CAMPOS DO JORDÃO
2024**

RESUMO

Este projeto visa desenvolver um sistema de gestão de academias utilizando um banco de dados relacional. O sistema armazenará informações sobre alunos, instrutores, treinos e pagamentos. Utilizando o modelo Entidade-Relacionamento (MER) e implementando o sistema em MySQL, espera-se melhorar a eficiência de armazenamento e recuperação de dados.

Palavras-Chave: Banco de Dados Relacional; Sistema de Gestão; Academia; MySQL; MER.

ABSTRACT

This project aims to develop a gym management system using a relational database. The system will store information about students, instructors, training and payments. Using the Entity-Relationship model (MER) and implementing the system in MySQL, it is expected to improve the efficiency of data storage and retrieval.

Keywords: Relational Databases; Management System; Academy; MySQL; Mer

SUMÁRIO

1	INTRODUÇÃO	5
1.1	Objetivos	5
1.2	Justificativa	5
1.3	Aspectos Metodológicos	6
1.4	Aporte Teórico	6
2	METODOLOGIA	7
2.1	FERRAMENTAS UTILIZADAS	7
2.2	DESCRIÇÃO DO PROJETO DE DADOS	8
2.3	COLETA DAS REGRAS DE NEGÓCIOS	9
3	RESULTADOS OBTIDOS	10
3.1	DESCRIÇÃO DO PROJETO DE DADOS	11
4	CONCLUSÃO	13
5	REFERÊNCIAS BIBLIOGRÁFICAS	14

1 INTRODUÇÃO

A administração eficaz de uma academia é um desafio que abrange o manejo de várias informações e dados que devem ser guardados, processados e consultados de maneira ordenada. Historicamente, academias utilizam métodos manuais ou planilhas para a gestão de dados, o que pode resultar em erros, retrabalho e complicações na obtenção de informações. Com o crescimento da quantidade de estudantes, planos de treino e pagamentos, fica cada vez mais complicado garantir a precisão e a agilidade desses procedimentos com sistemas convencionais.

1.1 Objetivos

Este trabalho tem por objetivo;

- Desenvolver um banco de dados relacional para o sistema de gestão de academias;
- Identificar requisitos funcionais e não funcionais;
- Modelar o banco de dados utilizando o MER.
- Implementar o banco de dados em MySQL.
- Testar a integridade e normalização das tabelas.

1.2 Justificativa

A administração eficiente de uma academia exige dados organizados, acessíveis e seguros. Ao serem mal administrados, esses dados podem gerar impactos adversos na empresa, como enganos na cobrança, problemas na gestão de programas de treinamento e obstáculos no suporte aos alunos. O uso de um banco de dados relacional possibilita a organização dessas informações de forma segura e escalável, além de reduzir o risco de falhas humanas. Com a automatização dos procedimentos administrativos, o sistema sugerido oferecerá uma operação mais rápida, eficaz e isenta de erros, favorecendo diretamente o desenvolvimento e a profissionalização da academia.

1.3 Aspectos Metodológicos

O projeto foi conduzido com base em uma pesquisa bibliográfica a respeito de bancos de dados relacionais, modelagem de dados e normalização, juntamente com a implementação prática do banco de dados utilizando MySQL e Pé de Galinha. O projeto passou pelas fases de coleta de requisitos, modelagem do banco de dados através do MER e criação das tabelas no SGBD. Além disso, foi executado o teste de integridade e normalização das tabelas, utilizando dados simulados para confirmar o funcionamento do sistema.

1.4 Aporte Teórico

A base teórica do projeto apoia-se nos princípios essenciais de bancos de dados relacionais, modelagem de dados e normalização, utilizando o modelo MER como fundamento para a organização do banco de dados. As principais referências incluem autores reconhecidos no campo, como C. J. Date e Abraham Silberschatz, cujas publicações oferecem a fundamentação teórica essencial para a criação do sistema de administração de academias.

2 METODOLOGIA

A execução deste projeto adotou uma metodologia organizada que incluiu estudo teórico e implementação prática. A abordagem foi segmentada em diversas fases fundamentais para assegurar a entrega de um Sistema de Gestão de Academia operacional, eficaz e seguro. Cada uma dessas fases foi meticulosamente planejada e implementada, empregando recursos e métodos adequados para assegurar o êxito do projeto.

2.1 Ferramentas Utilizadas

Notação MER (Modelo Entidade-Relacionamento): A notação MER foi empregada para ilustrar as entidades e suas interações no sistema. O MER possibilitou a visualização de maneira clara das tabelas, atributos e das relações entre as entidades como “Aluno”, “Treino”, “Instrutor” e “Pagamento”. O modelo contribuiu para organizar o banco de dados de maneira lógica e intuitiva, tornando mais fácil a compreensão e a implementação das tabelas no SGBD.

Pé de Galinha (Diagrama de Fluxo de Dados): A ferramenta Pé de Galinha foi empregada para delinear o fluxo de dados entre as entidades e os processos do sistema. Este recurso foi fundamental para reconhecer as interações entre os usuários (como colaboradores e estudantes) e o sistema. O diagrama facilitou a compreensão das operações do sistema e a determinação de como as informações seriam tratadas, assegurando que o fluxo de dados fosse coerente e eficaz. O Pé de Galinha foi empregado para ilustrar graficamente o percurso dos dados no sistema, desde o registro de um aluno até a criação de relatórios de pagamentos.

2.2 Descrição do Projeto de Dados

O Projeto de Dados tem como objetivo organizar de forma lógica e eficaz as informações essenciais para o Sistema de Gestão de Academia. Para isso, foi empregado o Modelo Entidade-Relacionamento (MER), que possibilitou reconhecer as entidades principais e suas interações, assegurando a estrutura e a integridade das informações. O banco de dados foi elaborado para guardar as informações referentes aos alunos, professores, exercícios e pagamentos.

As principais **entidades** e suas **atribuições** são as seguintes:

Aluno

- **Atributos:**
 - ID_Aluno: Identificador único do aluno (chave primária).
 - Nome: Nome completo do aluno.
 - Data_Nascimento: Data de nascimento do aluno.
 - Telefone: Número de telefone do aluno.
 - Email: Endereço de e-mail do aluno.

Instrutor

- **Atributos:**
 - ID_Instrutor: Identificador único do instrutor (chave primária).
 - Nome: Nome completo do instrutor.
 - Especialidade: Especialidade ou área de atuação do instrutor (exemplo: musculação, pilates, etc.).
 - Telefone: Número de telefone de contato do instrutor.

Treino

- **Atributos:**
 - ID_Treino: Identificador único do treino (chave primária).
 - Descricao: Descrição do treino ou exercício.
 - Duracao: Duração do treino (em minutos).
 - Tipo: Tipo de treino (exemplo: aeróbico, musculação, etc.).
 - Instrutor_ID: Referência ao instrutor que ministra o treino (chave estrangeira).
 - Aluno_ID: Referência ao aluno que participa do treino (chave estrangeira).

Atributos:

- ID_Pagamento: Identificador único do pagamento (chave primária).
- Data: Data do pagamento.
- Valor: Valor pago.
- Forma_Pagamento: Forma de pagamento (exemplo: cartão de crédito, boleto, etc.).
- Aluno_ID: Referência ao aluno que efetuou o pagamento (chave estrangeira).
- Status: Se a mensalidade expirou ou não

2.3 Coleta das Regras De Negócios

A coleta das regras de negócios foi uma etapa fundamental para a criação do banco de dados, uma vez que essas regras definem como os dados devem ser geridos e tratados no sistema. As diretrizes de negócio foram reunidas com base nas exigências operacionais da academia e nas características que o sistema deveria apresentar.

Cadastro de Alunos

- Cada aluno deve ter um ID único, e ao se matricular, o sistema deve registrar o plano de treino atribuído ao aluno, além de permitir a atualização de informações como telefone, email e histórico de treinos.
- O sistema deve permitir que um aluno altere seus dados de contato (telefone, email) e outros dados pessoais, mas deve garantir que o ID do aluno seja imutável.

Cadastro de Instrutores

- Cada instrutor deve ter um ID único e um conjunto de especialidades definidas, o que determina os tipos de treino que ele pode ministrar.
- O sistema deve garantir que o instrutor tenha horários de disponibilidade que não se sobreponham a outros instrutores.

Planejamento de Treinos

- Cada treino deve ser associado a um instrutor e a um ou mais alunos.
- O treino deve ter informações sobre sua duração, tipo e a descrição dos exercícios. Um treino deve ser alocado em horários específicos e não pode se sobrepor a outro treino agendado para o mesmo instrutor.

Pagamentos

- Cada aluno pode realizar múltiplos pagamentos, mas cada pagamento deve estar vinculado a um único aluno.
- Os pagamentos devem ser registrados com a data, valor e forma de pagamento (exemplo: cartão de crédito, débito, transferência bancária, etc.).
- O sistema deve garantir que, ao realizar um pagamento, o aluno tenha sua matrícula renovada ou atualizada, dependendo do plano contratado.

Relatórios e Acessibilidade de Dados

- O sistema deve permitir que os funcionários da academia consultem relatórios de pagamentos, treinos realizados e alunos matriculados, garantindo que as informações sejam acessíveis de maneira eficiente e segura.
- Os relatórios devem ser gerados com base em filtros como datas, valores e tipos de treino, e podem ser exportados para formatos como PDF ou Excel.

3 RESULTADOS OBTIDOS

O modelo conceitual do banco de dados para o gerenciamento de academia foi desenvolvido utilizando o Draw.io. Ele inclui as seguintes entidades e relacionamentos principais:

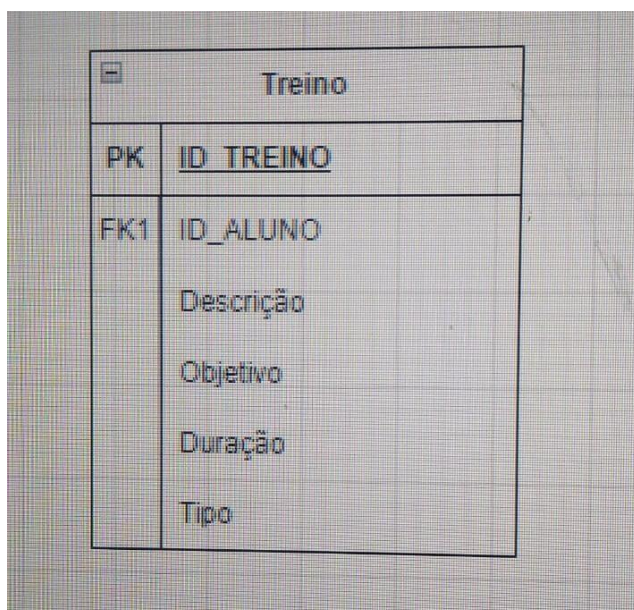
Aluno está relacionado com **Treino** e **Pagamento**. Cada aluno pode ter múltiplos treinos e pagamentos, mas um pagamento está vinculado a um único aluno.

Aluno	
PK	<u>ID_ALUNO</u>
	Nome
	CPF
	Data_Nascimento
	Telefone
	Email

Instrutor é associado a **Treino**. Cada treino é ministrado por um único instrutor, mas um instrutor pode ministrar vários treinos.

Instrutor	
PK	<u>ID INSTRUTOR</u>
	Nome
	Especialização
	Telefone

Treino está relacionado com **Aluno**. Cada aluno pode participar de múltiplos treinos, e cada treino pode ter vários alunos participando.



Treino	
PK	<u>ID TREINO</u>
FK1	ID_ALUNO
	Descrição
	Objetivo
	Duração
	Tipo

3.1 Descrição do Projeto de Dados

A descrição do projeto de dados trata do layout e da organização do banco de dados, especificando as principais entidades, seus atributos e as relações entre elas. O sistema de banco de dados foi elaborado com o Modelo Entidade-Relacionamento (MER), visando estabelecer uma estrutura eficiente, escalável e de fácil manutenção.

1. Aluno

A tabela de Aluno armazena as informações pessoais dos alunos. Seus principais atributos são:

- ID_Aluno (chave primária): Identificador único para cada aluno.
- Nome: Nome completo do aluno.
- Data_Nascimento: Data de nascimento do aluno.
- Telefone: Número de telefone para contato.
- Email: Endereço de e-mail do aluno.
- Plano: O plano de treino associado ao aluno.

2. Instrutor

A tabela de Instrutor armazena informações sobre os instrutores da academia. Seus principais atributos são:

- ID_Instrutor (chave primária): Identificador único para cada instrutor.
- Nome: Nome completo do instrutor.
- Especialidade: A especialidade do instrutor (ex: musculação, pilates).
- Telefone: Número de telefone de contato.

3. Treino

A tabela de Treino contém informações sobre os treinos disponíveis na academia. Seus principais atributos são:

- ID_Treino (chave primária): Identificador único para cada treino.
- Descricao: Descrição do treino ou exercício.
- Duracao: Duração do treino em minutos.
- Tipo: Tipo de treino (aeróbico, musculação, etc.).
- Instrutor_ID (chave estrangeira): Relaciona o treino com o instrutor responsável.
- Aluno_ID (chave estrangeira): Relaciona o treino com os alunos que o realizam.

4. Pagamento

A tabela de Pagamento armazena os registros de pagamentos efetuados pelos alunos. Seus principais atributos são:

- ID_Pagamento (chave primária): Identificador único para cada pagamento.
- Data: Data do pagamento.
- Valor: Valor pago pelo aluno.

- **Forma_Pagamento:** Forma utilizada para o pagamento (cartão de crédito, transferência bancária, etc.).
- **Aluno_ID (chave estrangeira):** Relaciona o pagamento ao aluno que efetuou o pagamento.

Relacionamentos:

- **Aluno e Treino:** Cada aluno pode participar de vários treinos, mas um treino pode ter vários alunos. Este é um relacionamento de muitos para muitos.
- **Instrutor e Treino:** Cada instrutor pode ministrar vários treinos, mas cada treino é ministrado por um único instrutor. Este é um relacionamento de um para muitos.
- **Aluno e Pagamento:** Cada aluno pode fazer múltiplos pagamentos, mas um pagamento é associado a um único aluno. Este é um relacionamento de um para muitos.

4 CONCLUSÃO

A criação do Sistema de Gestão de Academia fundamentado em banco de dados relacional atingiu suas metas principais, oferecendo uma solução eficaz e estruturada para a administração de dados essenciais, como alunos, instrutores, treinos e pagamentos. No decorrer do processo de desenvolvimento, diversas fases foram executadas, começando pela definição dos requisitos e seguindo até a implementação do banco de dados em MySQL e Pé de Galinha, incluindo a modelagem de dados através do Modelo Entidade-Relacionamento (MER) e no Draw.io. O sistema foi desenvolvido para assegurar a integridade, normalização e escalabilidade dos dados, satisfazendo requisitos funcionais e não funcionais.

O esquema conceitual do banco de dados foi bem organizado, com definição clara das entidades e de seus vínculos, o que torna a manutenção e a ampliação futura do sistema mais simples. A normalização das tabelas garante a remoção de redundâncias e a otimização na eficácia das consultas e operações no banco de dados. Adicionalmente, o uso de ferramentas como o Pé de Galinha contribuiu para mapear e visualizar o fluxo de dados, oferecendo uma compreensão nítida das interações entre as diversas entidades do sistema.

Nos testes, o sistema apresentou um bom desempenho nas operações de registro, atualização, pesquisa e pagamento, além de elaborar relatórios detalhados com informações precisas e relevantes. A validação do sistema utilizando dados simulados possibilitou reconhecer setores que podem ser melhorados, como a criação de uma interface gráfica para facilitar a utilização pelos colaboradores da academia.

Como propostas para aperfeiçoamentos futuros, ressalta-se a adição de recursos extras, como a conexão do sistema com aparelhos de supervisão de treinos e a automação de notificações para renovação de matrícula e quitação de mensalidades. Seria igualmente interessante ampliar a plataforma para celulares, oferecendo mais acessibilidade aos usuários.

5 REFERÊNCIAS BIBLIOGRÁFICAS

Documentação Draw.io. User Guide and Notation Standards. Disponível em:

<https://www.diagrams.net>

GUEDES, Gerson L. F. Modelagem de Dados: Fundamentos e Técnicas. 1ª ed. São Paulo: Erica, 2015.

ISO/IEC 9075. Database Language SQL - SQL/Foundation (SQL:2016). International Organization for Standardization, 2016

DATE, C. J. An Introduction to Database Systems. 8. ed. Boston: Addison-Wesley, 2019.