

UNIVERSIDADE FEDERAL DE ALFENAS - UNIFAL-MG
CURSO DE CIÊNCIA DA COMPUTAÇÃO

JOÃO PEDRO P. DE FREITAS
CARLOS AUGUSTO R. LORIM
JOÃO VICTOR R. G. NUNES
THALLES HENRIQUE R. G. PEREIRA

**MOTOMANAGER PRO: SISTEMA DE GESTÃO DE
REVENDA E OFICINA**

Trabalho Prático de Banco de Dados

ALFENAS - MG

2025

JOÃO PEDRO P. DE FREITAS
CARLOS AUGUSTO R. LORIM
JOÃO VICTOR R. G. NUNES
THALLES HENRIQUE R. G. PEREIRA

MOTOMANAGER PRO: SISTEMA DE GESTÃO DE REVENDA E OFICINA

Relatório apresentado como requisito avaliativo para
a disciplina de Banco de Dados do curso de Ciência
da Computação.

ALFENAS - MG

2025

Sumário

1	Definição do Domínio	2
1.1	Contexto do Sistema	2
1.2	Objetivo	2
2	Modelagem de Dados	2
2.1	Modelo Lógico	2
2.2	Esquema Relacional	2
3	Consultas Complexas (SQL)	3
3.1	1. Junção de Múltiplas Tabelas (JOIN)	3
3.2	2. Subconsulta (Subquery)	3
3.3	3. Agregação e Agrupamento (GROUP BY)	3
3.4	4. Multiconjunto (IN)	3
3.5	5. Busca Textual (LIKE)	4

1 Definição do Domínio

1.1 Contexto do Sistema

O **MotoManager Pro** é um software desenvolvido para modernizar a gestão de pequenas e médias lojas de motocicletas que operam no modelo híbrido (venda e oficina).

No cenário atual, muitos gestores utilizam cadernos ou planilhas desconexas para controlar o estoque e os custos de manutenção. Isso gera o problema do "*Lucro Invisível*": o lojista sabe por quanto vendeu a moto, mas desconhece quanto gastou exatamente com peças e reparos nela antes da venda.

1.2 Objetivo

O objetivo principal é fornecer uma plataforma unificada onde o gestor possa cadastrar veículos, lançar despesas atreladas a esses veículos e visualizar, em tempo real, o lucro líquido projetado e o valor total investido no estoque.

2 Modelagem de Dados

2.1 Modelo Lógico

O sistema foi modelado seguindo as regras da normalização (3FN). A estrutura baseia-se em três entidades principais interligadas:

- **Categorias:** Armazena os tipos de motos (Street, Sport, Custom, etc.).
- **Motos:** Entidade central. Armazena dados técnicos, legais e financeiros.
- **Gastos:** Registra despesas individuais vinculadas a uma moto.

2.2 Esquema Relacional

- **categorias** (id, nome)
- **motos** (id, nome, ano, placa, status, km, cor, valor_compra, valor_venda, #categoria_id)
- **gastos** (id, descricao, valor, data, #moto_id)

3 Consultas Complexas (SQL)

Implementação dos requisitos de consultas avançadas:

3.1 1. Junção de Múltiplas Tabelas (JOIN)

```
1 SELECT g.data, c.nome, m.nome, g.descricao, g.valor
2 FROM gastos g
3 JOIN motos m ON g.moto_id = m.id
4 JOIN categorias c ON m.categoria_id = c.id
5 ORDER BY g.data DESC;
```

3.2 2. Subconsulta (Subquery)

```
1 SELECT m.nome, m.valor_venda, c.nome
2 FROM motos m
3 JOIN categorias c ON m.categoria_id = c.id
4 WHERE m.valor_venda > (SELECT AVG(valor_venda) FROM motos)
5 ORDER BY m.valor_venda DESC;
```

3.3 3. Agregação e Agrupamento (GROUP BY)

```
1 SELECT c.nome, COUNT(m.id), AVG(m.valor_venda - m.valor_compra)
2 FROM categorias c
3 JOIN motos m ON m.categoria_id = c.id
4 GROUP BY c.nome
5 HAVING COUNT(m.id) >= 1;
```

3.4 4. Multiconjunto (IN)

```
1 SELECT m.nome, c.nome, m.valor_venda
2 FROM motos m
3 JOIN categorias c ON m.categoria_id = c.id
4 WHERE c.nome IN ('Sport', 'Touring', 'Custom');
```

3.5 5. Busca Textual (LIKE)

```
1 SELECT id, nome, placa, valor_venda
2 FROM motos
3 WHERE nome LIKE '%termo%' OR placa LIKE '%termo%';
```