

Análise de Modelagem Conceitual

Projeto Oficina Mecânica

1. Introdução

Este documento relata o desenvolvimento de um projeto de modelagem conceitual de banco de dados para um sistema de oficina mecânica, realizado como parte de um desafio proposto no bootcamp de Análise de Dados. O objetivo principal foi transformar uma narrativa textual em um modelo conceitual (diagrama ER) completo e funcional, aplicando boas práticas de modelagem.

2. Objetivos do Projeto

- Compreender os requisitos do sistema a partir de uma narrativa
 - Identificar entidades, atributos e relacionamentos
 - Normalizar e estruturar o modelo conforme boas práticas
 - Controle de ordens de serviço
 - Rastreamento de veículos e seus serviços
 - Registro de peças utilizadas
 - Acompanhamento de pagamentos
 - Representar o esquema relacional no MySQL Workbench
 - Documentar o raciocínio por trás das decisões de modelagem
-

3. Entidades e Relacionamentos

- **Cliente:** Unifica pessoa física e jurídica, evitando duplicidade de entidades. Armazena dados de pessoas físicas ou jurídicas (nome/razão social, CPF/CNPJ, telefone, endereço).
- **Veículo:** Define os veículos atendidos pela oficina, com informações sobre revisões, trocas de peças e consertos.
- **Cliente/Veículo:** Permite relacionamentos N:M entre clientes e veículos. Relaciona clientes e veículos com um tipo de vínculo (ex: proprietário, empresa, copropriedade).
- **Ordem de Serviço:** Registra o número da ordem, status da execução, serviços prestados, peças utilizadas e referência ao veículo, equipe e pagamento. `status_ordem ENUM('pendente', 'em_execucao', 'concluida', 'cancelada')`.

- **Serviço:** Informa o tipo de serviço, valor da mão de obra, quantidade, data de execução e autorização do cliente.
 - **Peça:** Contém detalhes sobre as peças utilizadas, tipo, valor, quantidade, datas e autorização do cliente.
 - **Equipe Mecânica:** Registra equipes da oficina com identificação, nome, endereço e especialidade.
 - **Pagamento:** Armazena as informações de pagamento vinculadas a uma ordem de serviço, status_pagamento ENUM('aguardando', 'confirmado', 'recusado', 'estornado').
-

4. Decisões de Modelagem

- Foram evitados nomes com espaços ou acentos para garantir compatibilidade com SQL.
- O uso de ENUM foi escolhido para atributos com valores fixos (ex: status_ordem, status_pagamento).
- Chaves primárias e estrangeiras seguem convenções claras, com padronização de nomes.
- Campos booleanos, como autorizacao_cliente, foram definidos como TINYINT (0/1) conforme prática do MySQL.

A modelagem foi baseada na narrativa do sistema de ordens de serviço de uma oficina, priorizando clareza, normalização e representação fiel do contexto real. Essa estrutura facilita consultas, relatórios e futura implementação do sistema em um banco relacional.

5. Desafios e Aprendizados

Aqui você pode escrever algo pessoal:

- Apreendi a aplicar as regras de cardinalidade corretamente
 - Desenvolvi mais confiança no uso de ferramentas como MySQL Workbench
 - Percebi a importância de nomear atributos de forma consistente e clara
 - Entendi melhor como modelar um sistema próximo do mundo real
-

6. Considerações Finais

Este projeto foi essencial para consolidar meus conhecimentos em modelagem conceitual. Com base na narrativa, foi possível criar um modelo relacional realista e bem estruturado, pronto para futuras implementações em ambientes reais.