

# Especificação do Trabalho

**Primeira entrega: 07/08/2024**

## Sistema de Delivery de Comida

Uma empresa que gerencia a entrega de comida, necessita de um banco de dados bem estruturado para gerenciar de forma eficiente todos os aspectos do seu funcionamento, desde os usuários até os pedidos. Para essa empresa há dois tipos de usuários principais: os compradores de comida, que chamaremos de clientes, e os restaurantes que vendem a comida.

1. Para cada cliente o banco de dados deve armazenar um ID único, nome completo, endereço de e-mail, telefone, uma lista de possíveis endereços para entrega, dados de pagamento (cpf) e histórico de pedidos.

2. Para os restaurantes o banco deve armazenar um ID único, o nome, o endereço, telefone, tipo de culinária oferecida pelo restaurante, a lista de pratos oferecidos, o horário de funcionamento e as avaliações feitas pelos clientes sobre o restaurante (nota de 0 a 5). A avaliação deve incluir um feedback textual, e a data e a hora em que foi realizada. Para uma gestão mais detalhada e flexível dos horários de funcionamento, crie a tabela `HorariosFuncionamentoRestaurantes`. Esta tabela permitirá que os restaurantes informem seus horários de abertura e fechamento para cada dia da semana, acomodando variações. Ela terá um `horario_funcionamento_id` como chave primária. Um `restaurante_id` (chave estrangeira para `Restaurantes`) indicará a qual restaurante o horário pertence. Os campos `dia_semana` (ex: 'Segunda', 'Sábado'), `horario_abertura` e `horario_fechamento` definirão os períodos. Uma restrição única será aplicada à combinação de `restaurante_id` e `dia_semana` para evitar horários duplicados para o mesmo dia. Com essa tabela, os campos de horário fixo da tabela `Restaurantes` podem ser removidos, tornando o sistema mais dinâmico.

3. Cada prato oferecido por um restaurante deve ter um nome, uma descrição detalhada, preço e um status de disponibilidade do prato. Para organizar o menu e facilitar a busca dos clientes, crie a tabela `CategoriasPratos`. Ela conterá `categoria_id` como chave primária e um `nome_categoria` (não nulo e único) para classificar os pratos (ex: "Entradas", "Massas", "Sobremesas", "Bebidas"). Lembrando que uma categoria pode conter vários pratos. Uma melhoria importante é a inclusão de um campo `estoque` (inteiro, maior ou igual a zero), que permite aos restaurantes gerenciar a quantidade disponível de itens, crucial para pratos com produção limitada.

4. Os pedidos são feitos diretamente pelos clientes. Portanto eles precisam visualizar os pratos oferecidos por cada restaurante. Um pedido deve ter um identificador único, bem como deve estar associado ao cliente que o fez e ao restaurante que o atenderá. Também é necessário armazenar a data e hora (timestamp) que o pedido foi realizado, o status atual (e.g., preparação, trânsito, entregue), a forma de pagamento utilizada (será simulado) e o valor total do pedido.

5. Um pedido é composto por uma lista de itens. Cada item está associado a um prato oferecido por algum restaurante, e deve conter a quantidade, preço por item e observações dos clientes relacionadas aos pratos.

6. As entregas são de inteira responsabilidade dos restaurantes, portanto não serão gerenciados entregadores. A empresa cobra 3% dos restaurantes para cada venda consumada. O registro é feito mediante pagamento do cliente.

É importante que seu modelo dê suporte aos requisitos funcionais da aplicação. O servidor (backend), onde MySQL será executado, atenderá duas aplicações diferentes. Uma acessada por clientes compradores de comida e outra acessada pelos restaurantes parceiros. Os clientes devem se autocadastrar, devem ter acesso aos pratos e aos preços para decidirem sobre suas compras. Devem também ter acesso a algum método de pagamento. Os donos de restaurantes devem cadastrar seus restaurantes no sistema, modificar pratos, preços e disponibilidade de entrega dos pratos. Use sua criatividade para sofisticar, sem deixar de lado os requisitos mínimos especificados aqui. Trabalho em equipes de 4 alunos, sendo 2 responsáveis pelo cliente e 2 pelo servidor/restaurante. A conexão deverá ser feita remotamente, ou seja, cada dupla executará em seu computador.