CENTRO DE EDUCAÇÃO TECNOLÓGICA DO AMAZONAS

INSTITUTO BENJAMIN CONSTANT - IBC

CURSO TÉCNICO DE INFORMÁTICA

SISTEMA DE GERENCIAMENTO INTEGRADO DE GESTÃO PARA PIZZARIAS

Sophia Roque Martins

Suely Nascimento Costa

Joquebede Barboza Santos

Suelen Tavares dos Santos

MANAUS - AM

2026

Avaliação Final do Projeto Técnico

Ambiente de Aprendizagem onde foi realizada a Atividade Prática III: Instituto Benjamin Constant - Laboratório de Informática 2.

Endereço em que se realizou a Prática Profissional: Rua Ramos Ferreira, 991 - Centro, Manaus - AM, 69010 - 120

Ambiente de Aprendizagem: Instituto Benjamin Constant - Laboratório de Informática 2.
Turno: Vespertino
Período de realização: 10/04/2026 a 12/05/2026.

Relatório Técnico apresentado como requisito parcial para aprovação no componente curricular no curso Técnico em Informática do CETAM.

Orientadora: Profª. Priscila L. S. Gonçalves

Nota Final das Atividades Práticas I da Etapa 01:_____.

Considerações do Instrutor da Atividade Prática:

_____
_____
_____
_____
_____
_____

Instrutora: Priscila L. S. Gonçalves

CENTRO DE EDUCAÇÃO TECNOLÓGICA DO AMAZONAS - CETAM

CURSO TÉCNICO DE NÍVEL MÉDIO EM INFORMÁTICA

Avaliação Final do Projeto Técnico

A Avaliação do projeto intitulado **SISTEMA DE GERENCIAMENTO INTEGRADO DE GESTÃO PARA PIZZARIAS**, desenvolvido pelos estudantes: **Joquebede Barboza Santos, Sophia Roque Martins, Suelen Tavares Dos Santos e Suely Nascimento Costa**, está composta de duas partes, sendo:

Parte 1 - Avaliação do acompanhamento sistemático realizado no decorrer do desenvolvimento do projeto, obtendo a nota: _____.

Parte 2 - Entrega do trabalho impresso e socialização da temática com a turma: _____.

Nota final nesta Etapa do Componente (Projeto): _____.

Considerações do Professor/Instrutor:

_____

_____

_____

_____.

.....

Professora Priscila Gonçalves
Instrutora do Projeto

LISTA DE FIGURAS – CONSULTAS SQL SIMPLES

Imagem 01 - Caso de Uso..... 22
Imagem 02 - Diagrama Conceitual.....23
Imagem 03 - Diagrama Lógico..... 24
Figura 01 - Consulta Simples..... 28
Figura 02 - Consulta Simples..... 29
Figura 03 - Consulta Simples..... 30
Figura 04 - Consulta Simples..... 31
Figura 05 - Consulta Simples..... 32
Figura 06- Consulta Simples..... 33
Figura 07 - Consulta Simples..... 34
Figura 08 - Consulta Simples..... 35
Figura 09 - Consulta Simples..... 36
Figura 10 - Consulta Simples..... 37
Figura 11 - Consulta Simples..... 38

LISTA DE FIGURAS – CONSULTAS SQL AVANÇADAS

Figura 01 - Consulta Avançada..... 48
Figura 02 - Consulta Avançada..... 48
Figura 03 - Consulta Avançada..... 48
Figura 04 - Consulta Avançada..... 49
Figura 05 - Consulta Avançada..... 49
Figura 06 - Consulta Avançada..... 49
Figura 07 - Consulta Avançada..... 49
Figura 09 - Consulta Avançada..... 50
Figura 10 - Consulta Avançada..... 50
Figura 11 - Consulta Avançada..... 50
Figura 12 - Consulta Avançada..... 50
Figura 13 - Consulta Avançada..... 50
Figura 14 - Consulta Avançada..... 51
Figura 15 - Consulta Avançada..... 51
Figura 16 - Consulta Avançada..... 51
Figura 18 - Consulta Avançada..... 51
Figura 19 - Consulta Avançada..... 52
Figura 20 - Consulta Avançada..... 52

Figura 21 - Consulta Avançada..... 52
Figura 22 - Consulta Avançada..... 52
Figura 23 - Consulta Avançada..... 75
Figura 24 - Consulta Avançada..... 76
Figura 25 - Consulta Avançada..... 77
Figura 26 - Consulta Avançada..... 78
Figura 27 - Consulta Avançada..... 79
Figura 28 - Consulta Avançada..... 80
Figura 29 - Consulta Avançada..... 81

LISTAS DE TABELA

Tabela 01 - ADM / Gerente.....25
Tabela 02 - Cliente..... 26
Tabela 03 - Atendente / Garçom..... 26
Tabela 04 - Entregador.....26
Tabela 05 - Requisitos Não Funcionais.....27
Tabela 06: Cargos..... 39
Tabela 07: Funcionários..... 40
Tabela 08: Clientes.....40
Tabela 09: Pedidos.....41
Tabela 10: Produtos..... 42
Tabela 11: Insumos..... 43
Tabela 12: Pagamentos.....43
Tabela 13: Estoque.....44
Tabela 14: Mesas..... 45
Tabela 15: Entregas..... 45
Tabela 16: Reservas.....46
Tabela 17: Dispositivos.....47
Tabela 18: Itens_Pedido.....48
Tabela 19: Quadro Financeiro..... 54
Tabela 20: Cronograma.....55

SUMÁRIO

1. INTRODUÇÃO..... 9
Modelagem de Dados..... 9
Gerenciamento de Requisitos..... 9
Implementação Técnica..... 9
2. JUSTIFICATIVA..... 10
Confiabilidade..... 10
Eficiência Logística..... 10
Segurança e Escalabilidade..... 11
Apoio à Tomada de Decisão..... 11
3. OBJETIVOS..... 11
3.1 Objetivo Geral..... 11
3.2 Objetivos Específicos..... 11
4.1 FERRAMENTAS..... 12
4.1.1 MySQL & phpMyAdmin..... 12
4.1.2 XAMPP..... 13
4.1.3 BrModelo..... 14
4.1.4 Draw.io..... 16
4.1.5 Visual Studio Code (VS Code)..... 17
4.1.6 Plataforma Google Drive, Google Docs e Google Sheets..... 18
4.1.7 Plataforma Chat GPT..... 19
4.1.8 Plataforma Gemini..... 19
4.1.9 Apostilas CETAM..... 20
4.2 MODELAGEM (CASO DE USO / MER / DIAGRAMA)..... 20
4.2.1 Diagrama de Caso de Uso e DER/MER..... 21
4.2.2. BANCO DE DADOS - Análise de Requisitos (Sigep - Pizzaria)..... 25
4.2.3 Requisitos Funcionais..... 25
Tabela 01 - ADM / Gerente..... 25
Tabela 02 - Cliente..... 26
Tabela 03 - Atendente / Garçom..... 26
Tabela 04 - Entregador..... 26
4.2.4 Requisitos Não Funcionais..... 27
Tabela 05 - Requisitos Não Funcionais..... 27
4.2.6 DICIONÁRIO DE DADOS..... 39
Tabela 06: Cargos..... 39
Tabela 07: Funcionários..... 40
Tabela 08: Clientes..... 40
Tabela 09: Pedidos..... 41
Tabela 10: Produtos..... 42

Tabela 11: Insumos....43
Tabela 12: Pagamentos....43
Tabela 13: Estoque....44
Tabela 14: Mesas.... 45
Tabela 15: Entregas....45
Tabela 16: Reservas....46
Tabela 17: Dispositivos....47
Tabela 18: Itens_Pedido.... 48
4.2.7 CONSULTAS AVANÇADAS (DML, DDL, DTL, DQL, DCL).... 49
4.2.8 FASE DE EXECUÇÃO....54
Tabela 19: Quadro Financeiro.... 54
Tabela 20: Cronograma.... 55
5. CONCLUSÃO.... 56
6. REFERÊNCIAS....57
7. APÊNDICE A.... 59
8. ANEXO.... 60

1. INTRODUÇÃO

Este relatório apresenta o desenvolvimento do SIGEP – Sistema Integrado de Gestão para Pizzarias, um projeto voltado à arquitetura e implementação de um banco de dados robusto capaz de centralizar e automatizar as operações de uma pizzaria. O sistema foi desenvolvido com o objetivo de otimizar processos administrativos e operacionais, proporcionando maior controle sobre pedidos, estoque, pagamentos, mesas e delivery.

A relevância deste trabalho fundamenta-se na necessidade de substituir processos manuais que frequentemente ocasionam inconsistências no estoque, atrasos na produção, falhas no atendimento e dificuldades no gerenciamento das informações. Nesse contexto, o SIGEP surge como uma solução tecnológica capaz de integrar os diferentes setores da pizzaria em um único sistema informatizado.

O projeto foi estruturado com base em três pilares principais:

Modelagem de Dados

Foram utilizadas ferramentas como brModelo e Draw.io para a elaboração dos diagramas e modelos de dados, permitindo representar de forma clara e organizada as regras de negócio do sistema, bem como os relacionamentos entre clientes, pedidos, produtos, estoque, funcionários e entregas.

Gerenciamento de Requisitos

Realizou-se o levantamento e definição dos Requisitos Funcionais (RF) e Requisitos Não Funcionais (RNF), assegurando que o sistema atendesse às necessidades operacionais da pizzaria, além de garantir segurança, desempenho, confiabilidade e escalabilidade.

Implementação Técnica

A implementação do banco de dados foi realizada utilizando a linguagem SQL em ambiente MySQL/phpMyAdmin, contemplando a criação das tabelas,

relacionamentos, restrições e mecanismos necessários para garantir a integridade referencial e o correto funcionamento do sistema.

Além disso, foram desenvolvidos o modelo entidade-relacionamento, o modelo lógico e o diagrama de classes, visando representar fielmente a estrutura e o comportamento do SIGEP. Dessa forma, o sistema foi projetado para oferecer suporte às operações diárias da pizzaria, promovendo maior organização, agilidade no atendimento e confiabilidade das informações armazenadas.

2. JUSTIFICATIVA

A sobrevivência e o crescimento de empresas no setor alimentício dependem diretamente da capacidade de gerenciar informações com precisão e eficiência. No caso das pizzarias, a existência de múltiplos canais de venda — como atendimento presencial, telefone, aplicativos de delivery e pedidos online — gera um grande volume de dados que, quando administrados manualmente, podem resultar em falhas operacionais, inconsistências no estoque e atrasos na produção e entrega dos pedidos.

Diante desse cenário, a implementação do SIGEP justifica-se pela necessidade de aplicar soluções tecnológicas capazes de automatizar e integrar os processos administrativos e operacionais da pizzaria. O sistema busca resolver gargalos comuns de gestão por meio da centralização das informações em um banco de dados estruturado e confiável.

Entre os principais benefícios proporcionados pelo SIGEP, destacam-se:

Confiabilidade

Redução de inconsistências entre os setores de cozinha, estoque, atendimento e financeiro, garantindo maior precisão no armazenamento e manipulação das informações.

Eficiência Logística

Integração entre o salão, a cozinha e a equipe de entregas, permitindo maior sincronização das operações e redução do tempo de espera dos clientes.

Segurança e Escalabilidade

Utilização de restrições e relacionamentos no banco de dados SQL para assegurar a integridade referencial, proporcionando um sistema seguro, organizado e preparado para futuras expansões.

Apoio à Tomada de Decisão

Centralização dos dados operacionais e financeiros para geração de consultas e relatórios precisos, auxiliando no controle de vendas, estoque e desempenho da pizzaria.

Dessa forma, este projeto propõe uma solução técnica viável e necessária para a transformação digital de empreendimentos do ramo gastronômico, contribuindo para a modernização da gestão comercial e para a melhoria da qualidade dos serviços prestados.

O objetivo deste documento é detalhar as etapas de construção dessa infraestrutura, demonstrando a viabilidade técnica do SIGEP como solução para a automação e gerenciamento eficiente de pizzarias.

3. OBJETIVOS

3.1 Objetivo Geral

Desenvolver e implementar a arquitetura de um banco de dados relacional para o sistema SIGEP, visando a automação dos processos operacionais, a integridade das informações e a modernização do gerenciamento de pedidos presenciais, mesas, estoque e delivery.

3.2 Objetivos Específicos

● Levantar e documentar os requisitos funcionais e não funcionais necessários para o pleno funcionamento de uma pizzaria informatizada;

● Elaborar a modelagem conceitual e lógica do banco de dados utilizando Diagramas Entidade-Relacionamento (DER), representando as conexões entre clientes, pedidos, produtos, estoque, funcionários, pagamentos e entregas;

● Implementar o esquema físico do banco de dados no SGBD MySQL por meio da criação de tabelas, definição de chaves primárias e estrangeiras e aplicação de restrições de integridade;

● Validar a persistência dos dados através da inserção de registros de teste e execução de consultas SQL para análise e auditoria do sistema;

● Garantir a integridade referencial e o cumprimento das regras de negócio, reduzindo erros humanos, redundância e duplicidade de informações;

● Desenvolver mecanismos que auxiliem no controle operacional da pizzaria, incluindo gerenciamento de pedidos, estoque, mesas e delivery;

● Criar consultas e relatórios que forneçam suporte à tomada de decisão, permitindo acompanhar vendas, movimentações de estoque e desempenho operacional do estabelecimento.

4. DESENVOLVIMENTO

4.1 FERRAMENTAS

4.1.1 MySQL & phpMyAdmin

Para a execução deste projeto, foram utilizadas ferramentas consagradas no desenvolvimento de software: MySQL & phpMyAdmin: A etapa de implementação física e gerenciamento dos dados foi realizada utilizando o MySQL, um Sistema Gerenciador de Banco de Dados (SGBD) relacional de código aberto, amplamente reconhecido por sua performance, confiabilidade e robustez. No contexto deste trabalho, o MySQL atuou como o motor principal para a persistência das informações, sendo responsável por interpretar e executar os scripts SQL gerados previamente na fase de modelagem conceitual e lógica. A escolha por este SGBD justifica-se pela sua total compatibilidade com os padrões da linguagem SQL

(Structured Query Language), o que garante a integridade referencial e a organização eficiente dos dados em tabelas relacionadas.

Para operacionalizar a interação com o servidor MySQL de forma intuitiva, utilizou-se o phpMyAdmin como interface gráfica de administração baseada em web. Esta ferramenta foi fundamental para simplificar tarefas complexas, permitindo a criação do banco de dados, a importação dos scripts.sql e a visualização imediata da estrutura das tabelas sem a necessidade exclusiva de linhas de comando. Através do phpMyAdmin, foi possível realizar testes de inserção de registros, validar as chaves primárias e estrangeiras e garantir que a arquitetura projetada fosse fielmente replicada no ambiente de produção local. Juntas, essas ferramentas proporcionaram um ambiente de desenvolvimento ágil, seguro e alinhado com as práticas atuais do mercado de tecnologia da informação.

Recursos:

- **Suporte a Transações (ACID):** Garante que todas as operações em uma venda, por exemplo, ocorram com sucesso ou sejam canceladas, evitando dados incompletos.
- **Integridade Referencial:** Através do uso de Chaves Estrangeiras (Foreign Keys), ele impede que você apague um dado que esteja ligado a outro (ex: apagar um cliente que tem pedidos).
- **Alta Performance:** Capacidade de lidar com milhões de registros e consultas simultâneas de forma rápida.
- **Segurança de Acesso:** Sistema de privilégios que permite definir exatamente quem pode ler, escrever ou apagar dados no sistema.
- **Multiplataforma:** Pode ser instalado em Windows, Linux ou macOS, facilitando o trabalho em diferentes ambientes.

4.1.2 XAMPP

Para viabilizar a execução e os testes do projeto em um ambiente controlado, foi utilizada a distribuição XAMPP, uma ferramenta essencial que atua como um servidor local multiplataforma. O XAMPP funciona como uma pilha de soluções (software stack) que integra, em um único pacote, o servidor HTTP Apache, o

sistema gerenciador de banco de dados MariaDB/MySQL e interpretadores para as linguagens de programação PHP e Perl.

A utilização desta ferramenta foi estratégica para o trabalho escolar, pois permitiu transformar a estação de trabalho em um ambiente de desenvolvimento completo (localhost), eliminando a necessidade de conexão externa com servidores de hospedagem remota durante as fases de implementação e validação. Através do painel de controle do XAMPP, foi possível gerenciar de forma centralizada ativação e monitoramento dos serviços necessários para o funcionamento do banco de dados.

O módulo Apache foi responsável por hospedar a interface de gerenciamento, enquanto o serviço MySQL permitiu o processamento das requisições de dados em tempo real. Além disso, a facilidade de configuração do XAMPP garantiu que as portas de comunicação e os privilégios de acesso fossem estabelecidos de maneira ágil, proporcionando um fluxo de trabalho contínuo entre a fase de modelagem e a aplicação prática. Assim, o XAMPP consolidou-se como a base infraestrutural do projeto, garantindo que todas as ferramentas integradas pudessem operar de forma harmoniosa e eficiente.

Recursos:

- **Apache (Servidor Web):** O recurso que processa requisições e "entrega" as páginas para o seu navegador.
- **MariaDB/MySQL:** O motor de banco de dados para armazenamento persistente de informações.
- **PHP:** O interpretador de linguagem que permite criar sites dinâmicos que se comunicam com o banco de dados.
- **Perl:** Uma linguagem de programação voltada para manipulação de textos e administração de sistemas.

4.1.3 BrModelo

A fase inicial de abstração e estruturação de dados foi realizada através do brModelo, uma ferramenta de software dedicada à modelagem de dados relacionais baseada na metodologia de Carlos Alberto Heuser. A utilização desta ferramenta foi fundamental para a construção do Modelo Entidade-Relacionamento (MER),

permitindo identificar com clareza as entidades, seus respectivos atributos e os relacionamentos existentes no sistema.

O brModelo destaca-se no ambiente acadêmico brasileiro por sua fidelidade aos conceitos teóricos da engenharia de software, oferecendo uma interface didática que prioriza a organização lógica das regras de negócio antes da implementação técnica no banco de dados. Um dos recursos mais significativos utilizados no projeto foi a capacidade do brModelo de realizar a engenharia direta. Após a validação do diagrama conceitual, a ferramenta permitiu a conversão automática para o modelo lógico, estruturando as tabelas, definindo chaves primárias e estabelecendo a integridade referencial por meio de chaves estrangeiras. Além disso, a geração automática do script SQL possibilitou uma transição fluida e livre de erros para o ambiente de execução no MySQL.

Dessa forma, o brModelo não atuou apenas como um editor gráfico, mas como uma ferramenta de suporte à decisão, garantindo que a arquitetura do banco de dados fosse planejada de forma eficiente, normalizada e documentada conforme as melhores práticas de desenvolvimento.

Recursos:

- **Criação do Diagrama Entidade-Relacionamento (DER):** Paleta de Componentes Padrão que oferece ícones específicos para entidades (retângulos), relacionamentos (losangos), atributos (elipses) e especializações (triângulos).
- **Gestão de Cardinalidades:** Permite definir facilmente os limites de participação (0,1; 1,1; 0,n; 1,n) através de menus de contexto.
- **Atributos Especializados:** Suporte para atributos compostos, multivalorados, derivados e chaves primárias.
- **Conversão Conceitual para Lógica:** Transforma o desenho de entidades em um modelo de tabelas (modelo relacional) com apenas um clique.
- **Geração de Script SQL:** Cria o código necessário (DDL) para ser executado em SGBDs como MySQL, PostgreSQL ou Oracle.
- **Engenharia Reversa (Básica):** Em algumas versões, auxilia na visualização de um esquema lógico a partir de um script.

- **Verificador de Consistência:** O software analisa o diagrama e aponta erros, como entidades sem atributos, relacionamentos sem cardinalidade ou objetos isolados.
- **Dicionário de Dados:** Permite descrever o significado e o tipo de dado de cada campo, servindo como uma documentação técnica completa para o projeto.
- **Normalização Automatizada:** Ajuda a garantir que as tabelas geradas sigam as primeiras formas normais do modelo relacional.

4.1.4 Draw.io

Uma ferramenta de diagramação baseada em vetores amplamente reconhecida pela sua versatilidade e suporte a padrões abertos. No desenvolvimento deste projeto, o Draw.io desempenhou um papel estratégico ao permitir a criação de representações gráficas que transcendem a modelagem técnica, facilitando a comunicação da arquitetura do sistema para diferentes públicos.

A ferramenta foi utilizada para padronizar a estética dos diagramas conceituais, garantindo que a simbologia de entidades e relacionamentos fosse apresentada de forma clara, profissional e visualmente atraente, o que é essencial para a qualidade de um relatório técnico acadêmico. Além da estética, a escolha pelo Draw.io justificou-se pela sua ampla biblioteca de formas específicas para a Engenharia de Software e Modelagem de Dados (ERD).

Através de seus recursos de edição flexíveis, foi possível organizar os fluxos de informações de maneira lógica, utilizando camadas e conectores inteligentes que mantêm a integridade do desenho mesmo durante ajustes estruturais. A capacidade de exportação em múltiplos formatos de alta resolução, como PNG e PDF, garantiu que as imagens inseridas na documentação final mantivessem a legibilidade e a precisão dos detalhes. Assim, o Draw.io atuou como o elo final entre a engenharia de dados e a documentação, provendo uma visão sistêmica e organizada que enriquece a compreensão global do banco de dados projetado.

Recursos:

- **Bibliotecas ERD Integradas:** Conjunto de formas específicas para Diagramas Entidade-Relacionamento.
- **Ferramentas de Alinhamento:** Recursos para garantir simetria e organização visual automática.
- **Integração com Nuvem:** Salvamento direto no Google Drive ou GitHub, facilitando o trabalho colaborativo.
- **Exportação Multiformato:** Suporte para SVG, PDF, PNG e XML (para edição futura).
- **Extensibilidade:** Possibilidade de usar o software online ou offline como aplicativo desktop.

4.1.5 Visual Studio Code (VS Code)

O Visual Studio Code (VSCode) foi utilizado como o ambiente de desenvolvimento principal (IDE) para a organização dos scripts SQL, edição de códigos e estruturação da documentação técnica do projeto. A escolha por esta ferramenta justificou-se pela sua leveza e versatilidade, oferecendo um ambiente organizado e eficiente que contribuiu diretamente para a produtividade durante o desenvolvimento do sistema SIGEP.

No contexto deste trabalho, o VS Code permitiu a manipulação ágil dos arquivos de script, oferecendo recursos de destaque de sintaxe que facilitam a identificação de comandos e erros estruturais no código SQL. Além disso, a ferramenta serviu como repositório central para a organização dos arquivos de banco de dados e textos complementares, garantindo que o fluxo de trabalho entre a modelagem e a implementação física fosse mantido de forma integrada e acessível.

Recursos:

- **Edição de Scripts SQL:** Ambiente otimizado para a escrita e revisão dos comandos de criação e inserção de dados.
- **Destaque de Sintaxe (Syntax Highlighting):** Recurso visual que diferencia palavras-chave, strings e operadores, reduzindo a margem de erro humano.

- **Organização de Arquivos do Projeto:** Gerenciamento eficiente das pastas e arquivos que compõem a documentação técnica e técnica.
- **Suporte a Extensões:** Possibilidade de integrar plugins específicos para bancos de dados e formatação de texto.
- **Ambiente Multiplataforma:** Compatibilidade total com diferentes sistemas operacionais, mantendo a consistência do ambiente de desenvolvimento.

4.1.6 Plataforma Google Drive, Google Docs e Google Sheets

A plataforma Google Drive foi utilizada como ferramenta principal de armazenamento em nuvem, compartilhamento e organização dos arquivos relacionados ao desenvolvimento do projeto SIGEP. Sua utilização foi essencial para garantir acessibilidade, segurança das informações e integração entre os integrantes da equipe. Por meio do Google Drive, foi possível armazenar relatórios, diagramas, scripts SQL, imagens e demais documentos técnicos utilizados no projeto.

Dentro do ambiente Google Drive, também foram utilizados o Google Docs e o Google Sheets. O Google Docs foi empregado na elaboração colaborativa do relatório técnico, permitindo edição simultânea em tempo real. A ferramenta auxiliou na padronização textual, revisão acadêmica e organização das seções do documento. O Google Sheets foi utilizado para organização de tarefas, controle de cronogramas, estruturação de informações do banco de dados e acompanhamento das etapas de implementação e testes do sistema.

Recursos:

- **Compartilhamento de arquivos em nuvem:** Garantia de que todos os arquivos estivessem disponíveis para a equipe em qualquer local.
- **Armazenamento e backup automático:** Proteção contra a perda de dados durante o desenvolvimento.
- **Edição simultânea (Google Docs):** Colaboração em tempo real na redação do projeto técnico.
- **Estruturação de informações (Google Sheets):** Gestão de cronogramas e organização das tabelas e registros.

4.1.7 Plataforma ChatGPT

A plataforma ChatGPT foi utilizada como ferramenta de apoio acadêmico e técnico durante o desenvolvimento do projeto SIGEP. Sua utilização auxiliou na organização textual do relatório, revisão de conteúdos, esclarecimento de dúvidas técnicas e apoio complementar na elaboração de consultas SQL e descrições conceituais. Além disso, a ferramenta contribuiu para a otimização da pesquisa e organização das informações relacionadas à documentação técnica do projeto.

Recursos:

- **Apoio na elaboração textual:** Auxílio na estruturação de parágrafos e termos técnicos.
- **Revisão de conteúdos acadêmicos:** Verificação da clareza e coesão das informações apresentadas.
- **Auxílio em consultas SQL:** Suporte na construção e correção de códigos DDL e DML.
- **Explicações sobre modelagem:** Esclarecimento de conceitos sobre relacionamentos e normalização.

4.1.8 Plataforma Gemini

A plataforma Gemini foi utilizada como ferramenta complementar de pesquisa e apoio técnico durante o desenvolvimento do SIGEP. Seu uso contribuiu para a obtenção de informações relacionadas à modelagem de banco de dados, desenvolvimento de sistemas e organização de conteúdos acadêmicos. A ferramenta auxiliou na complementação das pesquisas e no entendimento de conceitos aplicados ao projeto de forma ágil.

Recursos:

- **Pesquisa de conteúdos técnicos:** Acesso a informações atualizadas sobre SGBDs e desenvolvimento.

- **Apoio conceitual em banco de dados:** Reforço na compreensão de regras de negócio e integridade referencial.
- **Organização de informações acadêmicas:** Ajuda na estruturação lógica dos dados coletados para o relatório.

4.1.9 Apostilas CETAM

As apostilas disponibilizadas pelo CETAM foram utilizadas como o principal material de apoio teórico durante o desenvolvimento do projeto. O conteúdo estudado forneceu a base necessária para a compreensão dos conceitos de modelagem de dados, bancos de dados relacionais, linguagem SQL e desenvolvimento de sistemas informatizados. Esses materiais contribuíram diretamente para a aplicação correta dos conceitos acadêmicos na construção do SIGEP, orientando as etapas de análise e implementação.

Conteúdos Utilizados:

- **Modelagem de dados e DER:** Guia para a criação de diagramas seguindo as normas técnicas.
- **SQL e comandos DDL/DML:** Base para a criação de tabelas e manipulação de registros.
- **Integridade referencial e Normalização:** Regras para evitar redundância e garantir a consistência dos dados.
- **Estruturação de sistemas:** Metodologia para organizar projetos de informática de forma profissional.

4.2 MODELAGEM (CASO DE USO / MER / DIAGRAMA)

O desenvolvimento deste projeto foi estruturado em fases sequenciais, seguindo as normas da engenharia de software para garantir uma arquitetura de dados eficiente e funcional. O processo dividiu-se nas seguintes etapas fundamentais:

Diagrama de Caso de Uso O processo iniciou-se com o Diagrama de Caso de Uso, onde o foco foi mapear as interações dos usuários com o sistema. Nesta etapa, definiram-se as funcionalidades principais, como o ato de realizar pedidos e

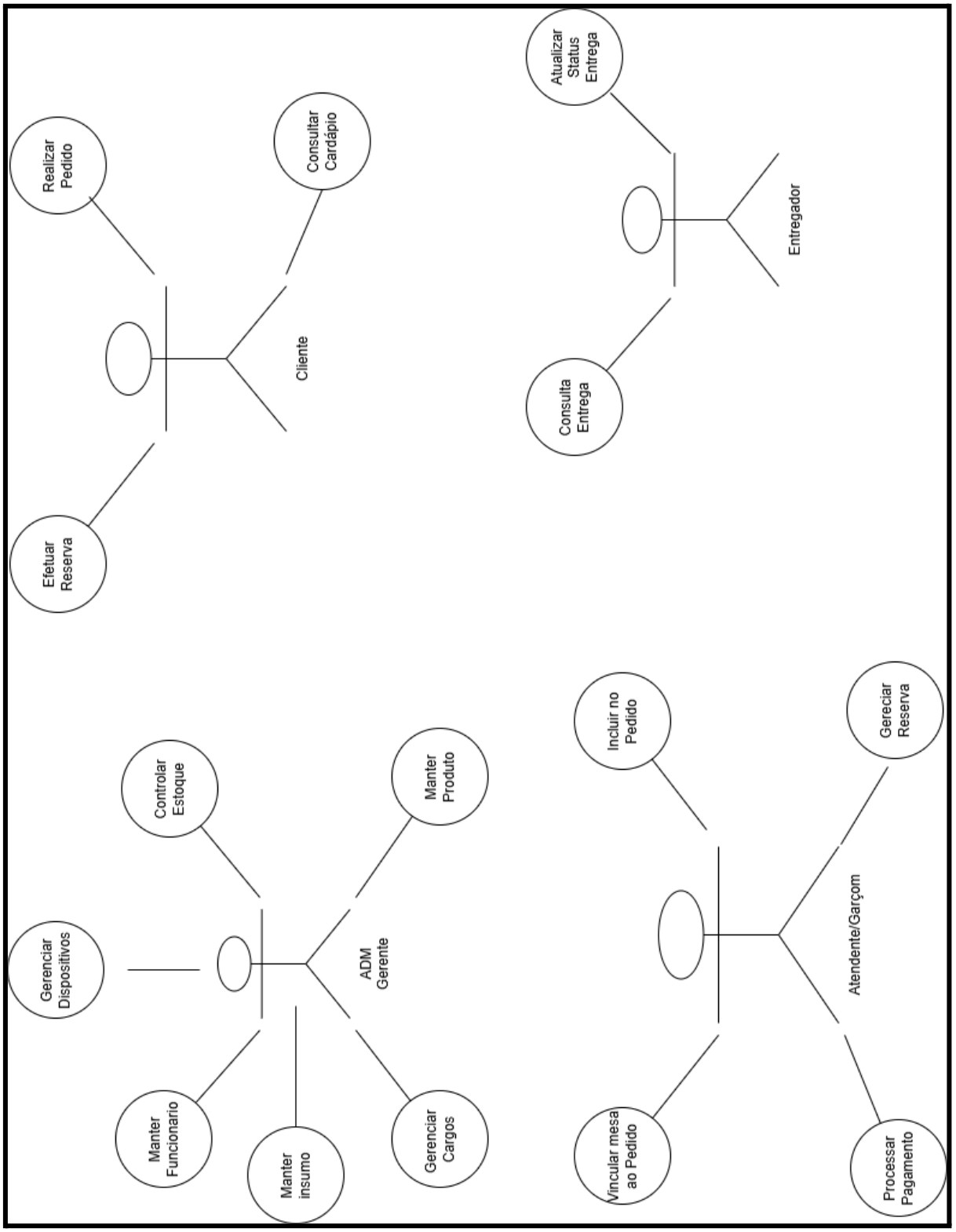
gerenciar o catálogo de produtos, estabelecendo o escopo do que o software deve executar para atender às necessidades do negócio.

Modelagem Conceitual (MER) Na sequência, elaborou-se a Modelagem Conceitual, onde ocorreu o levantamento das entidades principais: **Clientes, Pedidos e Produtos**. Utilizando a ferramenta brModelo, foram estabelecidos os relacionamentos entre essas entidades e as regras de cardinalidade, criando uma visão abstrata que serve de base para o armazenamento das informações. Esta fase foi vital para garantir que a lógica do negócio estivesse correta antes de qualquer implementação técnica, evitando erros estruturais futuros.

Modelagem Lógica (DER) Por fim, realizou-se a Modelagem Lógica, traduzindo o diagrama conceitual para a realidade dos bancos de dados relacionais. Nesta fase, foram definidas as chaves primárias (PK) para identificação única de registros e as chaves estrangeiras (FK) para garantir a integridade referencial entre as tabelas. Além disso, foram especificados os tipos de dados para cada atributo, como `VARCHAR` para nomes e endereços, `INT` para identificadores e quantidades, e `DECIMAL` para valores monetários, assegurando a precisão necessária para o processamento das informações no SGBD MySQL.

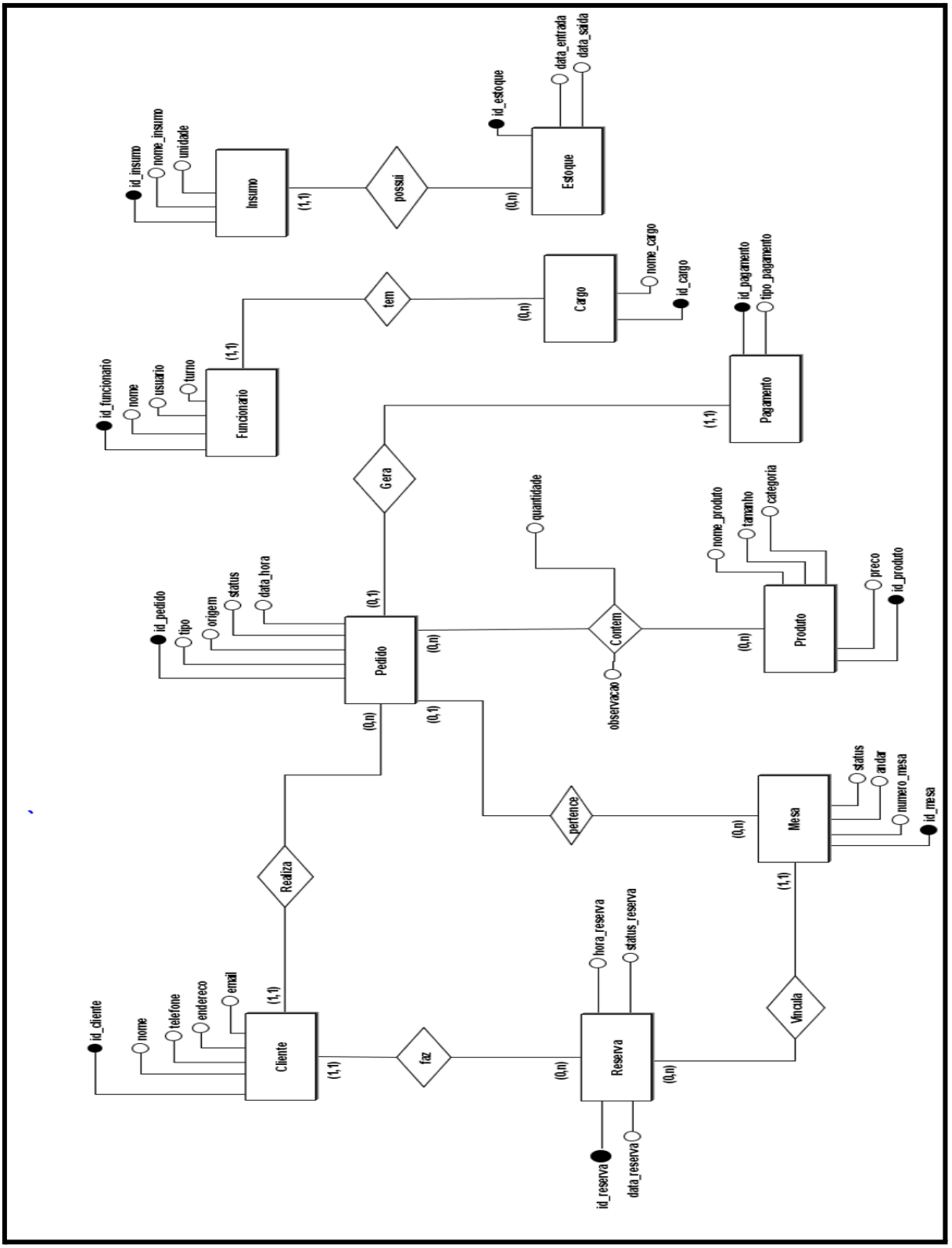
4.2.1 Diagrama de Caso de Uso e DER/MER

Imagem 01 - Caso de Uso



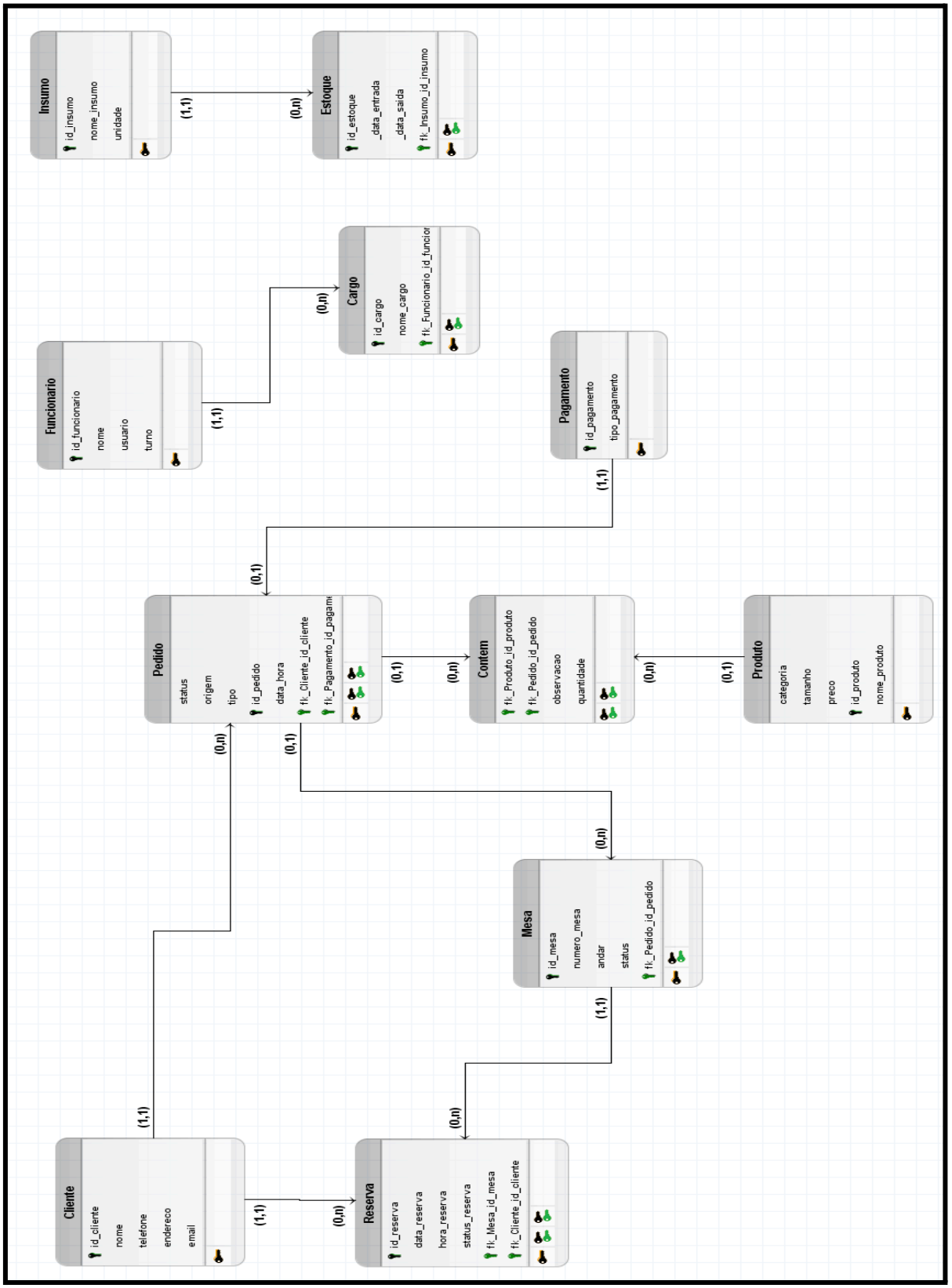
Autora: Suelen Tavares

Imagem 02 - Diagrama Conceitual



Autora: Suelen Tavares

Imagem 03 - Diagrama Lógico



Autora: Suelen Tavares

4.2.2. BANCO DE DADOS - Análise de Requisitos (Sigep - Pizzaria)

Definição :Um requisito representa uma condição ou capacidade essencial que o sistema deve satisfazer para atender às necessidades do cliente. No projeto da pizzaria, estes dividem-se em:

Requisitos Funcionais (RF): O sistema deve:Permitir o cadastro de funcionários; registrar reservas de mesas; emitir e acompanhar o status de entregas.

Requisitos Não Funcionais (RNF): O sistema deve: Garantir a integridade referencial por meio de constraints no banco de dados; realizar backups periódicos, com frequência mínima semestral. RC01: O sistema deve seguir padrões de segurança de dados, alta disponibilidade para pedidos e interface responsiva para operação em tempo real.

4.2.3 Requisitos Funcionais

Tabela 01 - ADM / Gerente

Identificador	Requisito Funcional	Requisito Complementar	Regra de Negócio	Prioridade
RF06	O sistema deve permitir manter o funcionário.	RC01	RGN04	Alta
RF07	O sistema deve permitir gerenciar dispositivos.	RC01	RGN05	Média
RF08	O sistema deve permitir controlar estoque.	RC01	RGN06	Alta
RF09	O sistema deve permitir gerenciar insumos.	RC01	RGN07	Média
RF10	O sistema deve permitir gerenciar cargos.	RC01	-	Baixa
RF11	O sistema deve permitir gerenciar produtos	RC01	RGN08	Alta

Autora: Sophia Roque

Tabela 02 - Cliente

Identificador	Requisito Funcional	Requisito Complementar	Regra de Negócio	Prioridade
RF12	O sistema deve permitir manter o o cliente	RC01	RGN09	Média
RF13	O sistema deve permitir realizar pedido.	RC01	RGN10	Alta
RF14	O sistema deve permitir consultar cardápio.	RC01	-	Alta

Autora: Sophia Roque

Tabela 03 - Atendente / Garçom

Identificador	Requisito Funcional	Requisito Complementar	Regra de Negócio	Prioridade
RF15	O sistema deve permitir vincular a mesa ao pedido.	RC01	RGN11	Alta
RF16	O sistema deve permitir incluir itens no pedido.	RC01	RGN12	Alta
RF17	O sistema deve permitir processar pagamento.	RC01	RGN13	Alta
RF18	O sistema deve permitir gerenciar reserva.	RC01	RGN09	Média

Autora: Sophia Roque

Tabela 04 - Entregador

Identificador	Requisito Funcional	Requisito Complementar	Regra de Negócio	Prioridade
RF19	O sistema deve permitir consultar a entrega.	RC01	-	Média

RF20	O sistema deve permitir atualizar o status da entrega.	RC01	RGN14	Alta

Autora: Sophia Roque

4.2.4 Requisitos Não Funcionais

Tabela 05 - Requisitos Não Funcionais

Identificador	Categoria	Descrição	Prioridade
RNF01	**Segurança**	Controle de acesso por perfil (Login/Senha) para proteger dados de funcionários e estoque.	Alta
RNF02	**Disponibilidade**	O sistema deve estar operacional 99,5% do tempo para garantir a realização de pedidos.	Alta
RNF03	**Desempenho**	O tempo de resposta para consultas (cardápio/entregas) não deve ultrapassar 2 segundos.	Média
RNF04	**Usabilidade**	Interface responsiva e intuitiva para uso em tablets (garçons) e smartphones (clientes).	Alta
RNF05	**Confiabilidade**	Garantia de integridade nos pagamentos; o sistema não deve duplicar ou perder cobranças.	Alta
RNF06	**Privacidade**	Armazenamento de dados pessoais em conformidade com a LGPD.	Alta
RNF07	**Escalabilidade**	Suporte para aumento de acessos simultâneos em horários de pico sem travamentos.	Média

Autora:Sophia Roque

4.2.4 CONSULTA EM SQL SIMPLES

Figura 01 - Consulta Simples

SELECT * FROM Cargos;

Perfil [Editar em linha] [Editar] [Demonstrar SQL] [Criar código PHP] [Atualizar]

Mostrar tudo | Número de linhas: 25 Filtrar linhas: Procurar nesta tabela

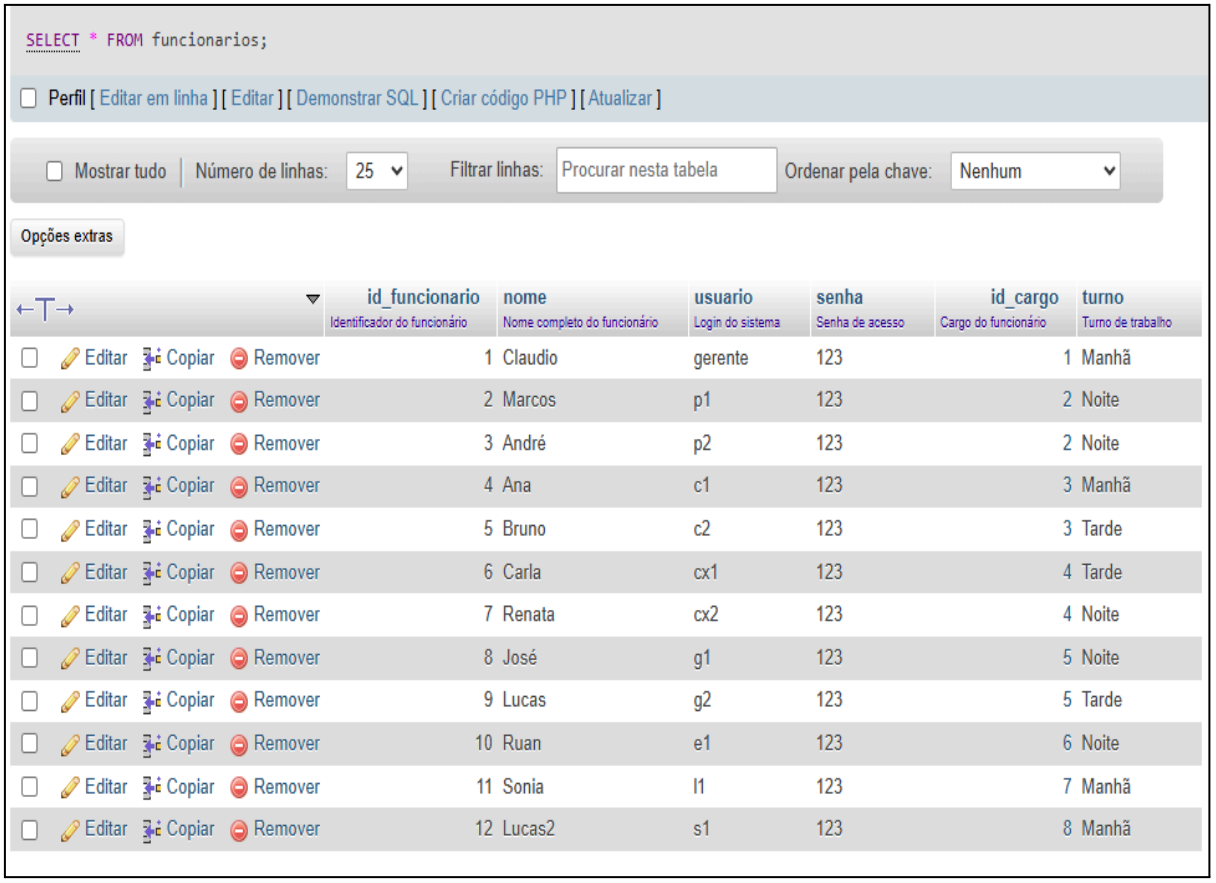
Opções extras

←T→			id_cargo Identificador do cargo	nome_cargo Nome do cargo (Gerente, Garçom, etc.)
Editar	Copiar	Remover	1	Gerente
Editar	Copiar	Remover	2	Pizzaiolo
Editar	Copiar	Remover	3	Auxiliar Cozinha
Editar	Copiar	Remover	4	Caixa
Editar	Copiar	Remover	5	Garçom
Editar	Copiar	Remover	6	Entregador
Editar	Copiar	Remover	7	Limpeza
Editar	Copiar	Remover	8	Supervisor
Editar	Copiar	Remover	9	Atendente
Editar	Copiar	Remover	10	Estoquista

Autora: Joquebede

Figura 01 – Primeiramente foi criado a tabela cargos por meio do comando CREATE TABLE cargos, responsável por armazenar os diferentes cargos existentes no sistema da pizzaria. Após a criação da tabela, foi realizada uma consulta para exibir todos os registros cadastrados. O comando executado retornou todas as colunas e registros da tabela cargos. Como resultado, foram encontrados 10 registros, correspondendo aos 10 cargos cadastrados no banco de dados. Cada registro representa um cargo específico utilizado no sistema sigep_pizzaria.

Figura 02 - Consulta Simples



```
SELECT * FROM funcionarios;
```

Perfil [Editar em linha] [Editar] [Demonstrar SQL] [Criar código PHP] [Atualizar]

Mostrar tudo | Número de linhas: 25 | Filtrar linhas: Procurar nesta tabela | Ordenar pela chave: Nenhum

Opções extras

←T→	id_funcionario Identificador do funcionário	nome Nome completo do funcionário	usuario Login do sistema	senha Senha de acesso	id_cargo Cargo do funcionário	turno Turno de trabalho
Editar Copiar Remover	1	Claudio	gerente	123	1	Manhã
Editar Copiar Remover	2	Marcos	p1	123	2	Noite
Editar Copiar Remover	3	André	p2	123	2	Noite
Editar Copiar Remover	4	Ana	c1	123	3	Manhã
Editar Copiar Remover	5	Bruno	c2	123	3	Tarde
Editar Copiar Remover	6	Carla	cx1	123	4	Tarde
Editar Copiar Remover	7	Renata	cx2	123	4	Noite
Editar Copiar Remover	8	José	g1	123	5	Noite
Editar Copiar Remover	9	Lucas	g2	123	5	Tarde
Editar Copiar Remover	10	Ruan	e1	123	6	Noite
Editar Copiar Remover	11	Sonia	l1	123	7	Manhã
Editar Copiar Remover	12	Lucas2	s1	123	8	Manhã

Autora: Joquebede

Figura 02 - O comando SELECT * FROM funcionarios; foi executado no ambiente phpMyAdmin para realizar a projeção de todos os atributos e tuplas da respectiva entidade. A consulta retornou um total de 12 registros, permitindo a verificação da integridade referencial entre as chaves primárias e as chaves estrangeiras (id_cargo). A estrutura visual confirmou a persistência correta dos campos de autenticação e a distribuição da carga horária nos domínios de turno predefinidos.

Figura 03 - Consulta Simples

SELECT * FROM clientes;

Perfil [Editar em linha] [Editar] [Demonstrar SQL] [Criar código PHP] [Atualizar]

Mostrar tudo | Número de linhas: 25 Filtrar linhas: Procurar nesta tabela Ordenar pela chave: Nenhum

Opções extras

←T→	id_cliente Identificador do cliente	nome Nome do cliente	telefone Telefone para contato	email Email do cliente	endereco Endereço para entrega
Editar Copiar Remover	1	Cliente 1	8191	c1@email.com	Rua A
Editar Copiar Remover	2	Cliente 2	8192	c2@email.com	Rua B
Editar Copiar Remover	3	Cliente 3	8193	c3@email.com	Rua C
Editar Copiar Remover	4	Cliente 4	8194	c4@email.com	Rua D
Editar Copiar Remover	5	Cliente 5	8195	c5@email.com	Rua E
Editar Copiar Remover	6	Cliente 6	8196	c6@email.com	Rua F
Editar Copiar Remover	7	Cliente 7	8197	c7@email.com	Rua G
Editar Copiar Remover	8	Cliente 8	8198	c8@email.com	Rua H
Editar Copiar Remover	9	Cliente 9	8199	c9@email.com	Rua I
Editar Copiar Remover	10	Cliente 10	8200	c10@email.com	Rua J
Editar Copiar Remover	11	Cliente 11	8201	c11@email.com	Rua K
Editar Copiar Remover	12	Cliente 12	8202	c12@email.com	Rua L
Editar Copiar Remover	13	Cliente 13	8203	c13@email.com	Rua M
Editar Copiar Remover	14	Cliente 14	8204	c14@email.com	Rua N
Editar Copiar Remover	15	Cliente 15	8205	c15@email.com	Rua O
Editar Copiar Remover	16	Cliente 16	8206	c16@email.com	Rua P
Editar Copiar Remover	17	Cliente 17	8207	c17@email.com	Rua Q
Editar Copiar Remover	18	Cliente 18	8208	c18@email.com	Rua R
Editar Copiar Remover	19	Cliente 19	8209	c19@email.com	Rua S
Editar Copiar Remover	20	Cliente 20	8210	c20@email.com	Rua T

Autora: Joquebede

Figura 03 - A execução do comando SELECT * FROM clientes; realizou a varredura completa da tabela, retornando todas as colunas e as **20 duplas** popularizadas para fins de teste. A consulta validou a persistência de dados sensíveis como telefone, e-mail e endereço, assegurando a integridade das informações de contato vinculadas a cada cliente. O resultado confirma que o esquema da base de dados suporta o armazenamento estruturado necessário para o gerenciamento de entregas e comunicações do sistema.

Figura 04 - Consulta Simples

```
SELECT * FROM pedidos;
```

Perfil [Editar em linha] [Editar] [Demonstrar SQL] [Criar código PHP] [Atualizar]

Mostrar tudo | Número de linhas: 25 Filtrar linhas: Procurar nesta tabela Ordenar pela chave: Nenhum

Opções extras

←T→	id_pedido Identificador do pedido	id_cliente Cliente que realizou o pedido	id_mesa Mesa (se presencial)	tipo Tipo do pedido	origem Origem do pedido	status Status do pedido	data_hora Data e hora do pedido
Editar Copiar Remover	1	1	1	Presencial	Totem	Finalizado	2026-05-05 15:34:46
Editar Copiar Remover	2	2	2	Presencial	Balcão	Em preparo	2026-05-05 15:34:46
Editar Copiar Remover	3	3	3	Presencial	Site	Servindo mesa	2026-05-05 15:34:46
Editar Copiar Remover	4	4	4	Delivery	iFood	Saiu para entrega	2026-05-05 15:34:46
Editar Copiar Remover	5	5	5	Delivery	Uber Eats	Finalizado	2026-05-05 15:34:46
Editar Copiar Remover	6	6	6	Delivery	Rappi	Em preparo	2026-05-05 15:34:46
Editar Copiar Remover	7	7	7	Presencial	Totem	Finalizado	2026-05-05 15:34:46
Editar Copiar Remover	8	8	8	Presencial	Balcão	Servindo mesa	2026-05-05 15:34:46
Editar Copiar Remover	9	9	9	Presencial	Site	Em preparo	2026-05-05 15:34:46
Editar Copiar Remover	10	10	10	Delivery	iFood	Saiu para entrega	2026-05-05 15:34:46
Editar Copiar Remover	11	11	1	Presencial	Totem	Finalizado	2026-05-05 15:34:46
Editar Copiar Remover	12	12	2	Presencial	Balcão	Em preparo	2026-05-05 15:34:46
Editar Copiar Remover	13	13	3	Delivery	iFood	Saiu para entrega	2026-05-05 15:34:46
Editar Copiar Remover	14	14	4	Delivery	Rappi	Em preparo	2026-05-05 15:34:46
Editar Copiar Remover	15	15	5	Presencial	Site	Finalizado	2026-05-05 15:34:46
Editar Copiar Remover	16	16	6	Delivery	Uber Eats	Saiu para entrega	2026-05-05 15:34:46
Editar Copiar Remover	17	17	7	Presencial	Totem	Servindo mesa	2026-05-05 15:34:46
Editar Copiar Remover	18	18	8	Presencial	Balcão	Finalizado	2026-05-05 15:34:46
Editar Copiar Remover	19	19	9	Delivery	iFood	Em preparo	2026-05-05 15:34:46
Editar Copiar Remover	20	20	10	Delivery	Rappi	Finalizado	2026-05-05 15:34:46

Autora: Joquebede

Figura 04 - A instrução SQL SELECT * FROM pedidos; foi aplicada para auditar o fluxo operacional, resultando na exibição de **20 registros** que detalham a volumetria de vendas do sistema. A consulta permite monitorar o estado das transações em tempo real através dos campos status e origem, validando a integração entre os pedidos presenciais e as plataformas de delivery. A presença de chaves estrangeiras (id_cliente e id_mesa) e o registro de data_hora confirmam a rastreabilidade completa de cada operação realizada na base de dados.

Figura 05 - Consulta Simples

```
SELECT * FROM produtos;
```

Perfil [Editar em linha] [Editar] [Demonstrar SQL] [Criar código PHP] [Atualizar]

Mostrar tudo | Número de linhas: 25 Filtrar linhas: Procurar nesta tabela Ordenar pela chave: Nenhum

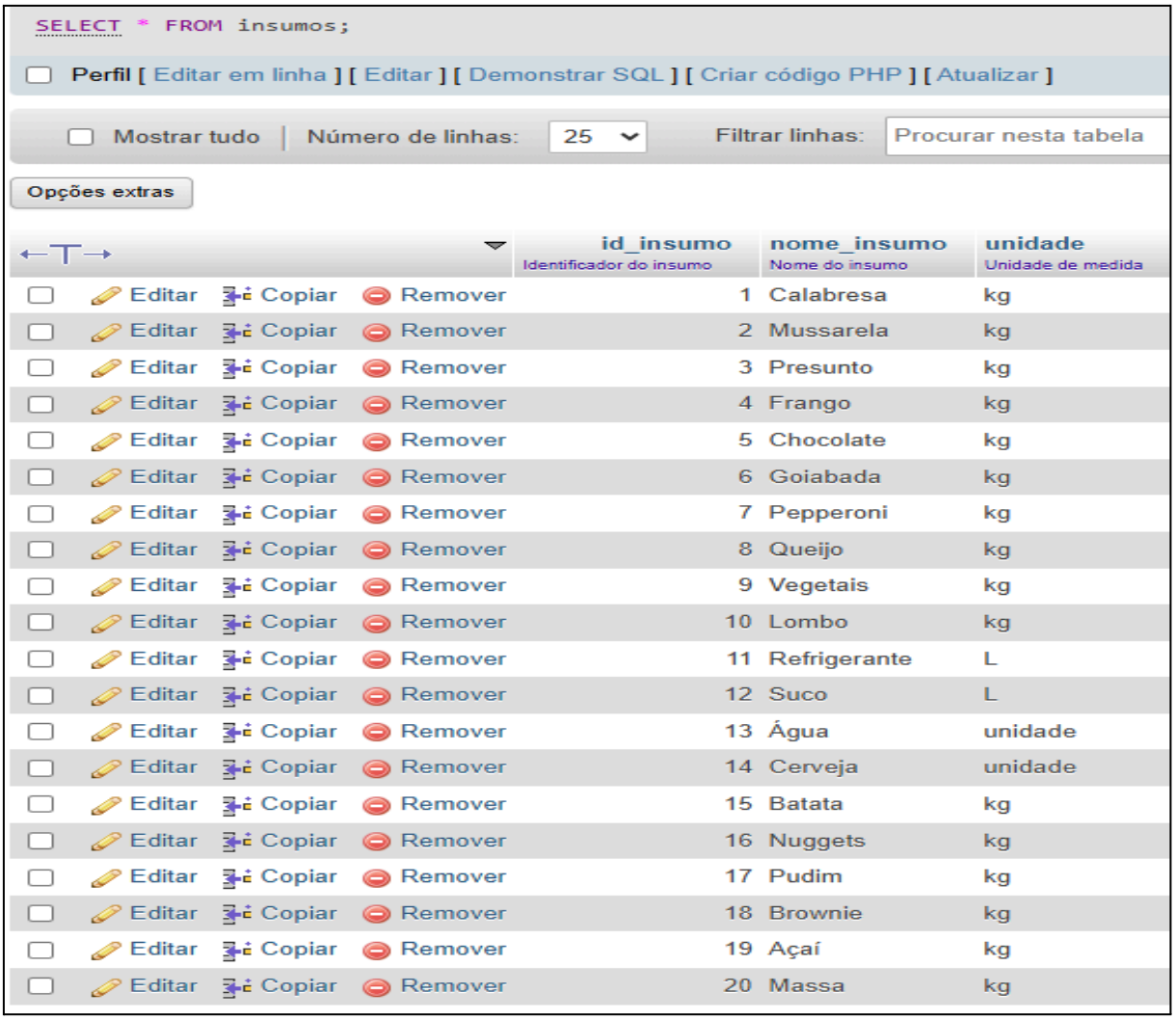
Opções extras

←T→	id_produto Identificador do produto	nome_produto Nome do produto	categoria Categoria (Pizza, Bebida, etc.)	tamanho Tamanho do produto	preco Preço do produto
Editar Copiar Remover	1	Calabresa	Pizza	Grande	45.00
Editar Copiar Remover	2	Mussarela	Pizza	Grande	40.00
Editar Copiar Remover	3	Portuguesa	Pizza	Grande	50.00
Editar Copiar Remover	4	Frango Catupiry	Pizza	Grande	48.00
Editar Copiar Remover	5	Chocolate	Pizza	Broto	30.00
Editar Copiar Remover	6	Romeu e Julieta	Pizza	Broto	28.00
Editar Copiar Remover	7	Pepperoni	Pizza	Grande	55.00
Editar Copiar Remover	8	4 Queijos	Pizza	Grande	52.00
Editar Copiar Remover	9	Vegetariana	Pizza	Grande	49.00
Editar Copiar Remover	10	Lombo	Pizza	Grande	53.00
Editar Copiar Remover	11	Refrigerante	Bebida	Grande	12.00
Editar Copiar Remover	12	Suco	Bebida	Média	10.00
Editar Copiar Remover	13	Água	Bebida	Broto	5.00
Editar Copiar Remover	14	Cerveja	Bebida	Grande	15.00
Editar Copiar Remover	15	Batata Frita	Acompanhamento	Média	18.00
Editar Copiar Remover	16	Nuggets	Acompanhamento	Média	20.00
Editar Copiar Remover	17	Pudim	Sobremesa	Broto	12.00
Editar Copiar Remover	18	Brownie	Sobremesa	Broto	15.00
Editar Copiar Remover	19	Açaí	Sobremesa	Média	18.00
Editar Copiar Remover	20	Borda Recheada	Adicional	Broto	8.00

Autora: Joquebede

Figura 05 - A execução da query SELECT * FROM produtos; no phpMyAdmin retornou a listagem dos primeiros 10 itens do cardápio, filtrando especificamente a categoria de pizzas. A consulta válida a estrutura técnica da tabela, confirmando a correta atribuição de id_produto como chave primária e a precisão dos valores monetários na coluna preco. Esta visualização permite auditar a consistência dos dados inseridos, garantindo que as variações de tamanho (Grande e Broto) estejam devidamente mapeadas para o funcionamento do sistema.

Figura 06- Consulta Simples



SELECT * FROM insumos;

Perfil [Editar em linha] [Editar] [Demonstrar SQL] [Criar código PHP] [Atualizar]

Mostrar tudo | Número de linhas: 25 Filtrar linhas: Procurar nesta tabela

Opções extras

←T→	id_insumo (Identificador do insumo)	nome_insumo (Nome do insumo)	unidade (Unidade de medida)
Editar Copiar Remover	1	Calabresa	kg
Editar Copiar Remover	2	Mussarela	kg
Editar Copiar Remover	3	Presunto	kg
Editar Copiar Remover	4	Frango	kg
Editar Copiar Remover	5	Chocolate	kg
Editar Copiar Remover	6	Goiabada	kg
Editar Copiar Remover	7	Pepperoni	kg
Editar Copiar Remover	8	Queijo	kg
Editar Copiar Remover	9	Vegetais	kg
Editar Copiar Remover	10	Lombo	kg
Editar Copiar Remover	11	Refrigerante	L
Editar Copiar Remover	12	Suco	L
Editar Copiar Remover	13	Água	unidade
Editar Copiar Remover	14	Cerveja	unidade
Editar Copiar Remover	15	Batata	kg
Editar Copiar Remover	16	Nuggets	kg
Editar Copiar Remover	17	Pudim	kg
Editar Copiar Remover	18	Brownie	kg
Editar Copiar Remover	19	Açaí	kg
Editar Copiar Remover	20	Massa	kg

Autora: Joquebede

Figura 07- O comando SELECT * FROM insumos; foi processado para realizar a leitura integral da tabela de estoque, exibindo os componentes fundamentais para a produção. A consulta valida a padronização das unidades de medida, confirmando que os itens estão devidamente quantificados em quilogramas (**kg**) no banco de dados. Esta visualização técnica no **phpMyAdmin** assegura a integridade dos primeiros registros de insumos, permitindo o controle preciso do inventário necessário para a operação.

Figura 07 - Consulta Simples

```
SELECT * FROM `pagamentos`;
```

Perfil [Editar em linha] [Editar] [Demonstrar SQL] [Criar código PHP] [Atualizar]

Mostrar tudo | Número de linhas: 25 Filtrar linhas: Procurar nesta tabela Ordenar pela chave: Nenhu

Opções extras

←T→				id_pagamento Identificador do pagamento	id_pedido Pedido pago	tipo_pagamento Forma de pagamento	valor Valor pago	status Situação do pagamento
Editar	Copiar	Remover		1	1	Dinheiro	45.00	Pago
Editar	Copiar	Remover		2	2	Pix	40.00	Pago
Editar	Copiar	Remover		3	3	Cartão Crédito	50.00	Pago
Editar	Copiar	Remover		4	4	Cartão Débito	48.00	Pago
Editar	Copiar	Remover		5	5	iFood	30.00	Pago
Editar	Copiar	Remover		6	6	Uber Eats	28.00	Pago
Editar	Copiar	Remover		7	7	Rappi	32.00	Pago
Editar	Copiar	Remover		8	8	Carteira Digital	25.00	Pago
Editar	Copiar	Remover		9	9	Pix	20.00	Pago
Editar	Copiar	Remover		10	10	Cartão Crédito	55.00	Pago
Editar	Copiar	Remover		11	11	Dinheiro	45.00	Pago
Editar	Copiar	Remover		12	12	Pix	40.00	Pago
Editar	Copiar	Remover		13	13	Cartão Débito	38.00	Pago
Editar	Copiar	Remover		14	14	iFood	60.00	Pago

Autora: Joquebede

Figura 07 - A consulta realizada na tabela pagamentos do banco de dados sigep_pizzaria. O comando SELECT * FROM pagamentos; foi utilizado para exibir todos os registros cadastrados na tabela. Os dados apresentados incluem o identificador do pagamento, número do pedido, tipo de pagamento, valor e status. É possível observar diferentes formas de pagamento, como Dinheiro, Pix, Cartão de Crédito e iFood. Todos os registros exibidos possuem o status “Pago”, indicando pagamentos concluídos com sucesso. Dessa forma, a tabela pagamentos auxilia no controle financeiro e gerenciamento dos pagamentos do sistema SIGEP.

Figura 08 - Consulta Simples

```
SELECT * FROM estoque;
```

Perfil [Editar em linha] [Editar] [Demonstrar SQL] [Criar código PHP] [Atualizar]

Mostrar tudo | Número de linhas: 25 Filtrar linhas: Procurar nesta tabela Ordenar pela chave: Nenhum

Opções extras

←T→	id_estoque Identificador do estoque	id_insumo Insumo armazenado	quantidade Quantidade disponível	data_entrada Data de entrada no estoque	data_saida Data de saída do estoque
Editar Copiar Remover	1	1	20.00	2026-05-01	NULL
Editar Copiar Remover	2	2	30.00	2026-05-01	NULL
Editar Copiar Remover	3	3	15.00	2026-05-01	NULL
Editar Copiar Remover	4	4	25.00	2026-05-01	NULL
Editar Copiar Remover	5	5	10.00	2026-05-01	NULL
Editar Copiar Remover	6	6	12.00	2026-05-01	NULL
Editar Copiar Remover	7	7	18.00	2026-05-01	NULL
Editar Copiar Remover	8	8	20.00	2026-05-01	NULL
Editar Copiar Remover	9	9	15.00	2026-05-01	NULL
Editar Copiar Remover	10	10	14.00	2026-05-01	NULL
Editar Copiar Remover	11	11	40.00	2026-05-01	NULL
Editar Copiar Remover	12	12	20.00	2026-05-01	NULL
Editar Copiar Remover	13	13	50.00	2026-05-01	NULL
Editar Copiar Remover	14	14	60.00	2026-05-01	NULL
Editar Copiar Remover	15	15	10.00	2026-05-01	NULL
Editar Copiar Remover	16	16	20.00	2026-05-01	NULL
Editar Copiar Remover	17	17	12.00	2026-05-01	NULL
Editar Copiar Remover	18	18	10.00	2026-05-01	NULL
Editar Copiar Remover	19	19	15.00	2026-05-01	NULL
Editar Copiar Remover	20	20	50.00	2026-05-01	NULL

Autora: Joquebede

Figura 08 - A execução da instrução SQL SELECT * FROM estoque; permitiu a auditoria da movimentação física de produtos, apresentando os níveis atuais de armazenamento do sistema. A consulta valida a gestão temporal dos itens, confirmando o registro das colunas data_entrada e a presença de valores NULL em data_saida para produtos ainda disponíveis. Esta visualização no **phpMyAdmin** é essencial para monitorar a variável quantidade, assegurando que o controle de inventário esteja operando conforme a lógica de negócios da pizzaria.

Figura 09 - Consulta Simples

```
SELECT * FROM mesas;
```

Perfil [Editar em linha] [Editar] [Demonstrar SQL] [Criar código PHP] [Atualizar]

Mostrar tudo | Número de linhas: 25 Filtrar linhas: Procurar nesta tabela Ordenar p

Opções extras

←T→	id_mesa Identificador da mesa	numero_mesa Número físico da mesa	andar Andar da pizzaria	status Situação da mesa
Editar Copiar Remover	1	1	1º	Disponível
Editar Copiar Remover	2	2	1º	Reservada
Editar Copiar Remover	3	3	1º	Disponível
Editar Copiar Remover	4	4	1º	Indisponível
Editar Copiar Remover	5	5	1º	Reservada
Editar Copiar Remover	6	6	2º	Disponível
Editar Copiar Remover	7	7	2º	Reservada
Editar Copiar Remover	8	8	2º	Disponível
Editar Copiar Remover	9	9	2º	Indisponível
Editar Copiar Remover	10	10	2º	Disponível

Autora: Joquebede

Figura 09 - A instrução SQL SELECT * FROM mesas; foi executada para monitorar a disponibilidade física do salão, resultando na visualização técnica do status de ocupação em tempo real. A consulta válida a organização espacial do estabelecimento, confirmando o mapeamento das colunas numero_mesa, a localização por andar e a situação atual (Disponível, Reservada ou Indisponível). Esta análise no phpMyAdmin garante a integridade operacional da entidade, assegurando que o sistema de reservas consiga identificar corretamente a capacidade produtiva do ambiente físico.

Figura 10 - Consulta Simples

```
SELECT * FROM entregas;
```

Perfil [Editar em linha] [Editar] [Demonstrar SQL] [Criar código PHP] [Atualizar]

Mostrar tudo | Número de linhas: 25 Filtrar linhas: Procurar nesta tabela Ordenar pela cha

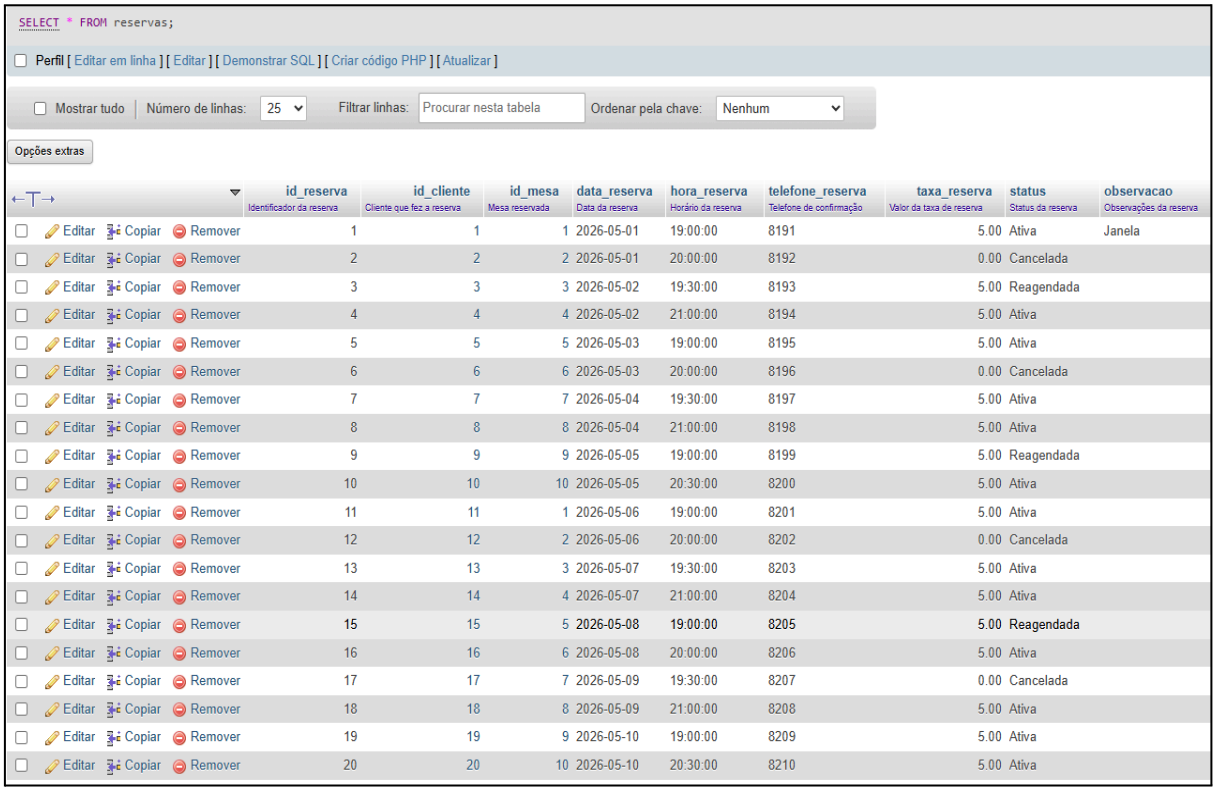
Opções extras

←T→	id_entrega Identificador da entrega	id_pedido Pedido entregue	endereco_entrega Endereço da entrega	status Status da entrega
Editar Copiar Remover	1	1	Rua A	Entregue
Editar Copiar Remover	2	2	Rua B	Saiu para entrega
Editar Copiar Remover	3	3	Rua C	Em preparo
Editar Copiar Remover	4	4	Rua D	Entregue
Editar Copiar Remover	5	5	Rua E	Saiu para entrega
Editar Copiar Remover	6	6	Rua F	Em preparo
Editar Copiar Remover	7	7	Rua G	Entregue
Editar Copiar Remover	8	8	Rua H	Saiu para entrega
Editar Copiar Remover	9	9	Rua I	Em preparo
Editar Copiar Remover	10	10	Rua J	Entregue
Editar Copiar Remover	11	11	Rua K	Entregue
Editar Copiar Remover	12	12	Rua L	Saiu para entrega
Editar Copiar Remover	13	13	Rua M	Em preparo
Editar Copiar Remover	14	14	Rua N	Entregue
Editar Copiar Remover	15	15	Rua O	Saiu para entrega
Editar Copiar Remover	16	16	Rua P	Entregue
Editar Copiar Remover	17	17	Rua Q	Em preparo
Editar Copiar Remover	18	18	Rua R	Entregue
Editar Copiar Remover	19	19	Rua S	Saiu para entrega
Editar Copiar Remover	20	20	Rua T	Entregue

Autora: Joquebede

Figura 10 - A instrução SQL SELECT * FROM entregas foi processada para monitorar o ciclo logístico externo, resultando na projeção de dados sobre o deslocamento de pedidos finalizados. A consulta validou a rastreabilidade da operação através da coluna status, permitindo o acompanhamento em tempo real entre as fases "Em preparo", "Saiu para entrega" e "Entregue". A estrutura técnica apresentada no phpMyAdmin confirma a integridade da chave estrangeira (id_pedido), assegurando que cada destino de entrega esteja diretamente vinculado à sua respectiva transação de venda.

Figura 11 - Consulta Simples



SELECT * FROM reservas;

Perfil [Editar em linha] [Editar] [Demonstrar SQL] [Criar código PHP] [Atualizar]

Mostrar tudo | Número de linhas: 25 Filtrar linhas: Procurar nesta tabela Ordenar pela chave: Nenhum

Opções extras

	id_reserva Identificador da reserva	id_cliente Cliente que fez a reserva	id_mesa Mesa reservada	data_reserva Data da reserva	hora_reserva Horário da reserva	telefone_reserva Telefone de confirmação	taxa_reserva Valor da taxa de reserva	status Status da reserva	observacao Observações da reserva
Editar Copiar Remover	1	1	1	2026-05-01	19:00:00	8191	5.00	Ativa	Janela
Editar Copiar Remover	2	2	2	2026-05-01	20:00:00	8192	0.00	Cancelada	
Editar Copiar Remover	3	3	3	2026-05-02	19:30:00	8193	5.00	Reagendada	
Editar Copiar Remover	4	4	4	2026-05-02	21:00:00	8194	5.00	Ativa	
Editar Copiar Remover	5	5	5	2026-05-03	19:00:00	8195	5.00	Ativa	
Editar Copiar Remover	6	6	6	2026-05-03	20:00:00	8196	0.00	Cancelada	
Editar Copiar Remover	7	7	7	2026-05-04	19:30:00	8197	5.00	Ativa	
Editar Copiar Remover	8	8	8	2026-05-04	21:00:00	8198	5.00	Ativa	
Editar Copiar Remover	9	9	9	2026-05-05	19:00:00	8199	5.00	Reagendada	
Editar Copiar Remover	10	10	10	2026-05-05	20:30:00	8200	5.00	Ativa	
Editar Copiar Remover	11	11	1	2026-05-06	19:00:00	8201	5.00	Ativa	
Editar Copiar Remover	12	12	2	2026-05-06	20:00:00	8202	0.00	Cancelada	
Editar Copiar Remover	13	13	3	2026-05-07	19:30:00	8203	5.00	Ativa	
Editar Copiar Remover	14	14	4	2026-05-07	21:00:00	8204	5.00	Ativa	
Editar Copiar Remover	15	15	5	2026-05-08	19:00:00	8205	5.00	Reagendada	
Editar Copiar Remover	16	16	6	2026-05-08	20:00:00	8206	5.00	Ativa	
Editar Copiar Remover	17	17	7	2026-05-09	19:30:00	8207	0.00	Cancelada	
Editar Copiar Remover	18	18	8	2026-05-09	21:00:00	8208	5.00	Ativa	
Editar Copiar Remover	19	19	9	2026-05-10	19:00:00	8209	5.00	Ativa	
Editar Copiar Remover	20	20	10	2026-05-10	20:30:00	8210	5.00	Ativa	

Autora: Joquebede

Figura 11 - A execução da query SELECT * FROM reservas; realizou a leitura da tabela de agendamentos, retornando registros que detalham a ocupação futura das mesas no sistema. A consulta valida o gerenciamento de eventos através das colunas de data_reserva, hora_reserva e status, permitindo o controle de situações como "Ativa", "Cancelada" ou "Reagendada". A análise técnica no phpMyAdmin confirma a correta aplicação de chaves estrangeiras (id_cliente e id_mesa) e o processamento da taxa_reserva, assegurando a integridade financeira e operacional do banco de dados.

4.2.5 LEGENDA TÉCNICA

A tabela abaixo define os termos técnicos utilizados na descrição das entidades:

PK (Primary Key): Chave primária, identificador exclusivo da tabela.

FK (Foreign Key): Chave estrangeira, indica relacionamento com outra tabela.

NOT NULL: O preenchimento do campo é obrigatório.

UNIQUE: O valor não pode se repetir na tabela.

AI (Auto Incremento): O sistema gera o número automaticamente.

4.2.6 DICIONÁRIO DE DADOS

Este dicionário descreve a estrutura das tabelas que compõem o banco de dados do sistema SIGEP, detalhando os tipos de dados, restrições e a finalidade de cada campo presente nas tabelas.

Tabela 06: Cargos

ENTIDADE: Cargos				
DESCRIÇÃO	Armazena os cargos disponíveis para os funcionários da pizzaria.			
VOLUME ESPERADO	Crescimento contínuo (10 iniciais)			
TEMPO DE RETENÇÃO	Permanente (dados mestres).			
ROTINA LIMPEZA	Não aplicável (tabela de referência).			
ATRIBUTOS				
ATRIBUTO	**NOME DO CAMPO**	**TIPO DE DADO**	**RESTRIÇÃO**	**DESCRIÇÃO**
Identificador do cargo	id_cargo	INT	PRIMARY KEY, AUTO_INCREMENT	Chave primária que identifica unicamente o cargo.

Nome do cargo	nome_cargo	VARCHAR (50)	NOT NULL	Nome descritivo do cargo (ex: 'Garçom', 'Gerente').

Autora: Sophia Roque

Tabela 07: Funcionários

ENTIDADE: Funcionários				
DESCRIÇÃO	Entidade responsável por armazenar dados dos colaboradores.			
VOLUME ESPERADO	12 registros (registro fixo inicial)			
TEMPO DE RETENÇÃO	Enquanto houver vínculo ativo			
ROTINA LIMPEZA	A cada 1 ano, fazer backup e limpar inativos			
ATRIBUTOS				
ATRIBUTO	**NOME DO CAMPO**	**TIPO DE DADO**	**RESTRIÇÃO**	**DESCRIÇÃO**
código do funcionário	id_funcionario	inteiro	PK	Identificador único.
nome completo	nome	varchar(100)	NOT NULL	Nome do colaborador.
login de acesso	usuario	varchar(50)	UNIQUE	Usuário do sistema.
senha	senha	varchar(255)	-	Senha de acesso.
código do cargo	id_cargo	inteiro	FK	Relacionamento com cargos.
turno	turno	enum	-	Período de trabalho.

Autora: Sophia Roque

Tabela 08: Clientes

ENTIDADE: Clientes	
DESCRIÇÃO	Cadastro de dados dos consumidores da pizzaria.

VOLUME ESPERADO	Crescimento contínuo (20 iniciais)			
TEMPO DE RETENÇÃO	Enquanto durar o relacionamento			
ROTINA LIMPEZA	Inativar após 2 anos sem pedidos			
ATRIBUTOS				
ATRIBUTO	**NOME DO CAMPO**	**TIPO DE DADO**	**RESTRIÇÃO**	**DESCRIÇÃO**
código do cliente	id_cliente	inteiro	PK	ID único do cliente.
nome	nome	varchar(100)	-	Nome do cliente.
telefone	telefone	varchar(20)	-	Contato telefônico.
e-mail	email	varchar(100)	-	Contato eletrônico.
endereço	endereco	varchar(200)	-	Local para entregas.

Autora: Sophia Roque

Tabela 09: Pedidos

ENTIDADE: Pedidos				
DESCRIÇÃO	Entidade responsável por armazenar os pedidos realizados.			
VOLUME ESPERADO	Crescimento contínuo (20 iniciais)			
TEMPO DE RETENÇÃO	2 anos para histórico de vendas			
ROTINA LIMPEZA	A cada 6 meses, arquivar pedidos finalizados			
ATRIBUTOS				
ATRIBUTO	**NOME DO CAMPO**	**TIPO DE DADO**	**RESTRIÇÃO**	**DESCRIÇÃO**
código do pedido	id_pedido	inteiro	PK	Identificador da venda.
código do cliente	id_cliente	inteiro	FK	Quem comprou.

tipo do pedido	tipo	enum	-	Presencial ou Delivery.
origem do pedido	origem	enum	-	Canal (iFood, Site, etc).
status	status	enum	-	Situação (Preparo, Saiu).
data e hora	data_hora	datetime	-	Momento do registro.

Autora: Sophia Roque

Tabela 10: Produtos

ENTIDADE: Produtos				
DESCRIÇÃO	Entidade responsável por armazenar o cardápio.			
VOLUME ESPERADO	Crescimento contínuo (20 iniciais)			
TEMPO DE RETENÇÃO	Vitalício enquanto estiver no menu			
ROTINA LIMPEZA	Revisão semestral de preços e itens			
ATRIBUTOS				
ATRIBUTO	**NOME DO CAMPO**	**TIPO DE DADO**	**RESTRIÇÃO**	**DESCRIÇÃO**
código do produto	id_produto	inteiro	PK	Identificador do produto.
nome do produto	nome_produto	varchar(100)	-	Nome da pizza/bebida.
categoria	categoria	varchar(50)	-	Pizza, Bebida, etc.
tamanho	tamanho	enum	-	Broto, Grande, etc.

Autora: Sophia Roque

Tabela 11: Insumos

ENTIDADE: Insumos				
DESCRIÇÃO	Cadastro das matérias-primas e ingredientes.			
VOLUME ESPERADO	Crescimento contínuo (20 iniciais)			
TEMPO DE RETENÇÃO	Vitalício enquanto houver uso			
ROTINA LIMPEZA	Revisão anual de itens ativos			
ATRIBUTOS				
ATRIBUTO	**NOME DO CAMPO**	**TIPO DE DADO**	**RESTRIÇÃO**	**DESCRIÇÃO**
código do insumo	id_insumo	inteiro	PK	Identificador da matéria-prima.
nome do insumo	nome_insumo	varchar(100)	NOT NULL	Nome (Ex: Calabresa, Massa).
unidade de medida	unidade	enum	-	kg, L, un, etc.

Autora: Sophia Roque

Tabela 12: Pagamentos

ENTIDADE: Pagamentos	
DESCRIÇÃO	Entidade responsável pelo registro financeiro das vendas.
VOLUME ESPERADO	1 registro por pedido
TEMPO DE RETENÇÃO	5 anos para fins contábeis
ROTINA LIMPEZA	Anual
ATRIBUTOS	

ATRIBUTO	NOME DO CAMPO	TIPO DE DADO	RESTRIÇÃO	DESCRIÇÃO
código do pagamento	id_pagamento	inteiro	PK	Identificador da transação.
código do pedido	id_pedido	inteiro	FK	Vínculo com a venda.
forma de pagamento	tipo_pagamento	enum	-	Pix, Cartão, Dinheiro.
valor pago	valor	decimal(10,2)	-	Total recebido.
situação	status	enum	-	Pago ou Pendente.

Autora: Sophia Roque

Tabela 13: Estoque

ENTIDADE: Estoque				
DESCRIÇÃO	Entidade responsável pelo controle de quantidades.			
VOLUME ESPERADO	20 registros (um por insumo)			
TEMPO DE RETENÇÃO	Histórico de 1 ano			
ROTINA LIMPEZA	A cada 3 meses, atualizar saldos			
ATRIBUTOS				
ATRIBUTO	**NOME DO CAMPO**	**TIPO DE DADO**	**RESTRIÇÃO**	**DESCRIÇÃO**
código do estoque	id_estoque	inteiro	PK	Identificador do registro.
código do insumo	id_insumo	inteiro	FK	Relacionamento com insumo.
quantidade	quantidade	decimal (10,2)	-	Saldo em estoque.

data de entrada	data_entrada	date	-	Última reposição.

Autora: Sophia Roque

Tabela 14: Mesas

ENTIDADE: Mesas				
DESCRIÇÃO	Controle das mesas físicas para atendimento presencial.			
VOLUME ESPERADO	10 registros fixos			
TEMPO DE RETENÇÃO	Vitalício			
ROTINA LIMPEZA	N/A (Apenas atualização de status)			
ATRIBUTOS				
ATRIBUTO	**NOME DO CAMPO**	**TIPO DE DADO**	**RESTRIÇÃO**	**DESCRIÇÃO**
código da mesa	id_mesa	inteiro	PK	Identificador da mesa.
número	numero_mesa	inteiro	-	Número físico da mesa.
andar	andar	enum	-	1º ou 2º andar.
situação	status	enum	-	Disponível, Ocupada, etc.

Autora: Sophia Roque

Tabela 15: Entregas

ENTIDADE: Entregas	
DESCRIÇÃO	Gestão logística de pedidos Delivery.
VOLUME ESPERADO	Conforme demanda diária

TEMPO DE RETENÇÃO	1 ano para histórico de rota			
ROTINA LIMPEZA	Semestral (arquivar entregas concluídas)			
ATRIBUTOS				
ATRIBUTO	**NOME DO CAMPO**	**TIPO DE DADO**	**RESTRIÇÃO**	**DESCRIÇÃO**
código da entrega	id_entrega	inteiro	PK	ID único da entrega.
código do pedido	id_pedido	inteiro	FK	Vínculo com o pedido.
endereço	endereco_entrega	varchar(200)	-	Destino da entrega.
status	status	enum	-	Saiu, Entregue, etc.

Autora: Sophia Roque

Tabela 16: Reservas

ENTIDADE: Reservas				
DESCRIÇÃO	Agendamentos de mesas para clientes.			
VOLUME ESPERADO	Crescimento contínuo (20 iniciais)			
TEMPO DE RETENÇÃO	1 ano			
ROTINA LIMPEZA	Mensal (limpar datas passadas)			
ATRIBUTOS				
ATRIBUTO	**NOME DO CAMPO**	**TIPO DE DADO**	**RESTRIÇÃO**	**DESCRIÇÃO**
código da reserva	id_reserva	inteiro	PK	Identificador da reserva.
código do cliente	id_cliente	inteiro	FK	Quem reservou.
mesa	id_mesa	inteiro	FK	Mesa escolhida.

data	data_reserva	date	-	Dia marcado.
hora	hora_reserva	time	-	Horário marcado.
status	status	enum	-	Ativa, Cancelada etc.

Autora: Sophia Roque

Tabela 17: Dispositivos

ENTIDADE: Dispositivos				
DESCRIÇÃO	Registra os dispositivos físicos utilizados no ecossistema da pizzaria			
VOLUME ESPERADO	Crescimento contínuo (20 iniciais)			
TEMPO DE RETENÇÃO	Permanente			
ROTINA LIMPEZA	Não aplicável (apenas alteração de status para 'Inativo' se o dispositivo for descartado).			
ATRIBUTOS				
ATRIBUTO	**NOME DO CAMPO**	**TIPO DE DADO**	**RESTRIÇÃO**	**DESCRIÇÃO**
Identificador do dispositivo	id_dispositivo	INT	PRIMARY KEY, AUTO_INCREMENT	Chave primária que identifica o dispositivo.
Tipo do dispositivo	tipo	ENUM	‘TV Cozinha', 'Painel Expedição', 'TV Expedição', 'Computador', 'Maquininha Cartão', 'Celular', 'Totem'	Categoria/tipo de uso do dispositivo no sistema.
Localização	localizacao	VARCHAR (100)	NULL	Descrição física de onde o dispositivo está instalado.

Status	status	ENUM	DEFAULT 'Ativo'	Define se o aparelho está 'Ativo' ou 'Inativo'.

Autora: Sophia Roque

Tabela 18: Itens_Pedido

ENTIDADE: Itens_Pedidos				
DESCRIÇÃO	Detalha os produtos e quantidades que compõem cada pedido individual.			
VOLUME ESPERADO	Crescimento contínuo (20 iniciais)			
TEMPO DE RETENÇÃO	5 anos (para fins fiscais e histórico de relatórios).			
ROTINA LIMPEZA	Arquivamento/Backup anual de registros antigos para otimização de performance.			
ATRIBUTOS				
ATRIBUTO	**NOME DO CAMPO**	**TIPO DE DADO**	**RESTRIÇÃO**	**DESCRIÇÃO**
Item do pedido	id_item	INT	PRIMARY KEY, AUTO_INCREMENT	Chave primária que identifica a linha do item.
Pedido relacionado	id_pedido	INT	FOREIGN KEY (Pedidos)	Chave estrangeira ligando o item ao seu respectivo pedido.
Produto do pedido	id_produto	INT	FOREIGN KEY (Produtos)	Chave estrangeira ligando o item ao catálogo de produtos.
Quantidade	quantidade	INT	NULL	Quantidade solicitada deste produto no pedido.
Observações	observacao	VARCHAR (200)	NULL	Notas adicionais do cliente (ex: "Sem cebola", "Borda recheada").

Autora: Sophia Roque

4.2.7 **CONSULTAS AVANÇADAS** (DML, DDL, DTL, DQL, DCL)

Exemplos de comandos utilizados no projeto:

DDL (Data Definition Language): CREATE TABLE Produtos (...);

DML (Data Manipulation Language): INSERT INTO Clientes (nome, telefone) VALUES ('João', '99123-4567');

DQL (Data Query Language): SELECT p.nome_produto, c.nome FROM Pedidos p JOIN Clientes c ON p.id_cliente = c.id_cliente.

Figura 01 - Consulta Avançada

✔ O MySQL retornou um conjunto vazio (ex. zero registros). (Consulta levou 0.0014 segundos.)

```
CREATE DATABASE sigep_pizzaria;
```

Figura 02 - Consulta Avançada

```
-- ==================================== -- ADMINISTRATIVO --
==================================== CREATE TABLE Cargos ( id_cargo INT AUTO_INCREMENT
PRIMARY KEY COMMENT 'Identificador do cargo', nome_cargo VARCHAR(50) NOT NULL COMMENT 'Nome do
cargo (Gerente, Garçom, etc.)' ) COMMENT='Tabela de cargos dos funcionários';
```

Figura 03 - Consulta Avançada

```
CREATE TABLE Funcionarios ( id_funcionario INT AUTO_INCREMENT PRIMARY KEY COMMENT
'Identificador do funcionário', nome VARCHAR(100) NOT NULL COMMENT 'Nome completo do
funcionário', usuario VARCHAR(50) UNIQUE COMMENT 'Login do sistema', senha VARCHAR(255)
COMMENT 'Senha de acesso', id_cargo INT COMMENT 'Cargo do funcionário', turno
ENUM('Manhã','Tarde','Noite') COMMENT 'Turno de trabalho', FOREIGN KEY (id_cargo) REFERENCES
Cargos(id_cargo) ) COMMENT='Tabela de funcionários da pizzaria';
```

Figura 04 - Consulta Avançada

```
-- ==================================== -- DISPOSITIVOS --
==================================== CREATE TABLE Dispositivos ( id_dispositivo INT
AUTO_INCREMENT PRIMARY KEY COMMENT 'Identificador do dispositivo', tipo ENUM( 'TV Cozinha',
'Painel Expedição', 'TV Expedição', 'Computador', 'Maquininha Cartão', 'Celular', 'Totem' )
COMMENT 'Tipo do dispositivo utilizado no sistema', localizacao VARCHAR(100) COMMENT 'Local
onde o dispositivo está', status ENUM('Ativo','Inativo') DEFAULT 'Ativo' COMMENT 'Status do
```

Figura 05 - Consulta Avançada

```
-- ==================================== -- DISPOSITIVOS -- ==================================== CREATE
TABLE Dispositivos ( id_dispositivo INT AUTO_INCREMENT PRIMARY KEY COMMENT 'Identificador do
dispositivo', tipo ENUM( 'TV Cozinha', 'Painel Expedição', 'TV Expedição', 'Computador', 'Maquininha
Cartão', 'Celular', 'Totem' ) COMMENT 'Tipo do dispositivo utilizado no sistema', localizacao
VARCHAR(100) COMMENT 'Local onde o dispositivo está', status ENUM('Ativo','Inativo') DEFAULT 'Ativo'
COMMENT 'Status do dispositivo' ) COMMENT='Dispositivos utilizados no funcionamento da pizzaria';
```

Figura 06 - Consulta Avançada

```
-- ==================================== -- CLIENTES --
==================================== CREATE TABLE Clientes ( id_cliente INT AUTO_INCREMENT
PRIMARY KEY COMMENT 'Identificador do cliente', nome VARCHAR(100) COMMENT 'Nome do cliente',
telefone VARCHAR(20) COMMENT 'Telefone para contato', email VARCHAR(100) COMMENT 'Email do
cliente', endereco VARCHAR(200) COMMENT 'Endereço para entrega' ) COMMENT='Cadastro de
clientes';
```

Figura 07 - Consulta Avançada

```
-- ==================================== -- MESAS -- ==================================== CREATE TABLE
Mesas ( id_mesa INT AUTO_INCREMENT PRIMARY KEY COMMENT 'Identificador da mesa', numero_mesa INT COMMENT
'Número físico da mesa', andar ENUM('1º','2º') COMMENT 'Andar da pizzaria', status
ENUM('Disponível','Indisponível','Reservada') COMMENT 'Situação da mesa' ) COMMENT='Mesas físicas da
pizzaria';
```

Figura 08 - Consulta Avançada

```
id_reserva INT AUTO_INCREMENT PRIMARY KEY COMMENT 'Identificador da reserva', id_cliente INT COMMENT 'Cliente que fez
a reserva', id_mesa INT COMMENT 'Mesa reservada', data_reserva DATE COMMENT 'Data da reserva', hora_reserva TIME
COMMENT 'Horário da reserva', telefone_reserva VARCHAR(20) COMMENT 'Telefone de confirmação', taxa_reserva
DECIMAL(10,2) COMMENT 'Valor da taxa de reserva', status ENUM('Ativa','Cancelada','Reagendada') COMMENT 'Status da
reserva', observacao VARCHAR(200) COMMENT 'Observações da reserva', FOREIGN KEY (id_cliente) REFERENCES
Clientes(id_cliente), FOREIGN KEY (id_mesa) REFERENCES Mesas(id_mesa) ) COMMENT='Controle de reservas de mesas';
```

Figura 09 - Consulta Avançada

```
-- ==================================== -- INSUMOS / ESTOQUE -- ==================================== CREATE TABLE Insumos (
id_insumo INT AUTO_INCREMENT PRIMARY KEY COMMENT 'Identificador do insumo', nome_insumo VARCHAR(100) COMMENT 'Nome do insumo',
unidade ENUM('ml','L','g','kg','unidade') COMMENT 'Unidade de medida' ) COMMENT='Matéria-prima utilizada na produção';
```

Figura 10 - Consulta Avançada

```
-- ==================================== -- PEDIDOS -- ==================================== CREATE TABLE Pedidos (
id_pedido INT AUTO_INCREMENT PRIMARY KEY COMMENT 'Identificador do pedido', id_cliente INT COMMENT 'Cliente que
realizou o pedido', id_mesa INT NULL COMMENT 'Mesa (se presencial)', tipo ENUM('Presencial','Delivery') COMMENT
'Tipo do pedido', origem ENUM( 'WhatsApp', 'iFood', 'Rappi', 'Uber Eats', 'Site', 'Balcão', 'Totem' ) COMMENT
'Origem do pedido', status ENUM( 'Pedido recebido', 'Em fila', 'Em preparo', 'No forno', 'Finalizado', 'Pronto',
'Saiu para entrega', 'Servindo mesa' ) COMMENT 'Status do pedido', data_hora DATETIME DEFAULT CURRENT_TIMESTAMP
```

Figura 11 - Consulta Avançada

```
-- ==================================== -- PAGAMENTOS -- ==================================== CREATE TABLE
Pagamentos ( id_pagamento INT AUTO_INCREMENT PRIMARY KEY COMMENT 'Identificador do pagamento', id_pedido INT COMMENT
'Pedido pago', tipo_pagamento ENUM( 'Dinheiro', 'Cartão Crédito', 'Cartão Débito', 'Pix', 'Carteira Digital',
'iFood', 'Rappi', 'Uber Eats' ) COMMENT 'Forma de pagamento', valor DECIMAL(10,2) COMMENT 'Valor pago', status
ENUM('Pago','Pendente') COMMENT 'Situação do pagamento', FOREIGN KEY (id_pedido) REFERENCES Pedidos(id_pedido) )
COMMENT='Pagamentos dos pedidos';
```

Figura 12 - Consulta Avançada

```
-- ==================================== -- ENTREGAS -- ==================================== CREATE TABLE Entregas (
id_entrega INT AUTO_INCREMENT PRIMARY KEY COMMENT 'Identificador da entrega', id_pedido INT COMMENT 'Pedido entregue',
endereco_entrega VARCHAR(200) COMMENT 'Endereço da entrega', status ENUM( 'Pedido recebido', 'Em preparo', 'Saiu para
entrega', 'Entregue' ) COMMENT 'Status da entrega', FOREIGN KEY (id_pedido) REFERENCES Pedidos(id_pedido) )
COMMENT='Controle de entregas';
```

Figura 13 - Consulta Avançada

```
-- ==================================== -- CARGOS -- ==================================== INSERT INTO Cargos
(nome_cargo) VALUES ('Gerente'),('Pizzaiolo'),('Auxiliar Cozinha'),('Caixa'), ('Garçom'),('Entregador'),
('Limpeza'),('Supervisor'), ('Atendente'),('Estoquista');
```

Figura 14 - Consulta Avançada

```
-- ==================================== -- FUNCIONÁRIOS (12) -- ==================================== INSERT
INTO Funcionarios (nome, usuario, senha, id_cargo, turno) VALUES ('Claudio','gerente','123',1,'Manhã'),
('Marcos','p1','123',2,'Noite'), ('André','p2','123',2,'Noite'), ('Ana','c1','123',3,'Manhã'),
('Bruno','c2','123',3,'Tarde'), ('Carla','cx1','123',4,'Tarde'), ('Renata','cx2','123',4,'Noite'),
('José','g1','123',5,'Noite'), ('Lucas','g2','123',5,'Tarde'), ('Ruan','e1','123',6,'Noite'),
('Sonia','l1','123',7,'Manhã'), ('Lucas2','s1','123',8,'Manhã');
```

Figura 15 - Consulta Avançada

```
-- ==================================== -- DISPOSITIVOS -- ==================================== INSERT INTO
Dispositivos (tipo, localizacao, status) VALUES ('TV Cozinha','Cozinha 1','Ativo'), ('TV Cozinha','Cozinha
2','Ativo'), ('Painel Expedição','Expedição','Ativo'), ('Maquininha Cartão','Caixa','Ativo'), ('Maquininha
Cartão','Caixa','Ativo'), ('Celular','Salão','Ativo'), ('Celular','Salão','Ativo'),
('Celular','Delivery','Ativo'), ('Celular','Delivery','Ativo'), ('Totem','Entrada','Ativo'),
('Totem','Entrada','Ativo'), ('TV Expedição','Bar','Ativo'), ('Maquininha Cartão','Salão VIP','Ativo'),
```

Figura 16 - Consulta Avançada

```
-- ==================================== -- PRODUTOS -- ==================================== INSERT INTO
Produtos (nome_produto,categoria,tamanho,preco) VALUES ('Calabresa','Pizza','Grande',45),
('Mussarela','Pizza','Grande',40), ('Portuguesa','Pizza','Grande',50), ('Frango
Catupiry','Pizza','Grande',48), ('Chocolate','Pizza','Broto',30), ('Romeu e Julieta','Pizza','Broto',28),
('Pepperoni','Pizza','Grande',55), ('4 Queijos','Pizza','Grande',52), ('Vegetariana','Pizza','Grande',49),
('Lombo','Pizza','Grande',53), ('Refrigerante','Bebida','Grande',12), ('Suco','Bebida','Média',10),
```

Figura 17 - Consulta Avançada

```
-- ==================================== -- INSUMOS -- ==================================== INSERT INTO
Insumos (nome_insumo, unidade) VALUES ('Calabresa','kg'),('Mussarela','kg'),('Presunto','kg'),('Frango','kg'),
('Chocolate','kg'),('Goiabada','kg'),('Pepperoni','kg'),('Queijo','kg'), ('Vegetais','kg'),('Lombo','kg'),
('Refrigerante','L'),('Suco','L'), ('Água','unidade'),('Cerveja','unidade'),('Batata','kg'),('Nuggets','kg'),
('Pudim','kg'),('Brownie','kg'),('Açaí','kg'),('Massa','kg');
```

Figura 18 - Consulta Avançada

```
-- ==================================== -- ESTOQUE -- ==================================== INSERT INTO
Estoque (id_insumo, quantidade, data_entrada, data_saida) VALUES (1,20,'2026-05-01',NULL),(2,30,'2026-05-
01',NULL), (3,15,'2026-05-01',NULL),(4,25,'2026-05-01',NULL), (5,10,'2026-05-01',NULL),(6,12,'2026-05-
01',NULL), (7,18,'2026-05-01',NULL),(8,20,'2026-05-01',NULL), (9,15,'2026-05-01',NULL),(10,14,'2026-05-
01',NULL), (11,40,'2026-05-01',NULL),(12,20,'2026-05-01',NULL), (13,50,'2026-05-01',NULL),(14,60,'2026-05-
01',NULL), (15,10,'2026-05-01',NULL),(16,20,'2026-05-01',NULL), (17,12,'2026-05-01',NULL),(18,10,'2026-05-
01',NULL), (19,15,'2026-05-01',NULL),(20,50,'2026-05-01',NULL);
```

Figura 19 - Consulta Avançada

```
-- ==================================== -- PEDIDOS (20) -- ==================================== INSERT INTO
Pedidos (id_cliente,id_mesa,tipo,origem,status) VALUES (1,1,'Presencial','Totem','Finalizado'),
(2,2,'Presencial','Balcão','Em preparo'), (3,3,'Presencial','Site','Servindo mesa'),
(4,4,'Delivery','iFood','Saiu para entrega'), (5,5,'Delivery','Uber Eats','Finalizado'),
(6,6,'Delivery','Rappi','Em preparo'), (7,7,'Presencial','Totem','Finalizado'),
(8,8,'Presencial','Balcão','Servindo mesa'), (9,9,'Presencial','Site','Em preparo'),
(10,10,'Delivery','iFood','Saiu para entrega'), (11,1,'Presencial','Totem','Finalizado'),
```

Figura 20 - Consulta Avançada

```
-- ==================================== -- PAGAMENTOS -- ==================================== INSERT INTO
Pagamentos (id_pedido,tipo_pagamento,valor,status) VALUES (1,'Dinheiro',45,'Pago'),(2,'Pix',40,'Pago'),
(3,'Cartão Crédito',50,'Pago'),(4,'Cartão Débito',48,'Pago'), (5,'iFood',30,'Pago'),(6,'Uber
Eats',28,'Pago'), (7,'Rappi',32,'Pago'),(8,'Carteira Digital',25,'Pago'), (9,'Pix',20,'Pago'),(10,'Cartão
Crédito',55,'Pago'), (11,'Dinheiro',45,'Pago'),(12,'Pix',40,'Pago'), (13,'Cartão Débito',38,'Pago'),
(14,'iFood',60,'Pago'), (15,'Uber Eats',42,'Pago'),(16,'Rappi',35,'Pago'), (17,'Pix',25,'Pago'),(18,'Cartão
```

Figura 21 - Consulta Avançada

```
-- ==================================== -- ENTREGAS (20) -- ==================================== INSERT
INTO Entregas (id_pedido,endereco_entrega,status) VALUES (1,'Rua A','Entregue'), (2,'Rua B','Saiu para
entrega'), (3,'Rua C','Em preparo'), (4,'Rua D','Entregue'), (5,'Rua E','Saiu para entrega'), (6,'Rua F','Em
preparo'), (7,'Rua G','Entregue'), (8,'Rua H','Saiu para entrega'), (9,'Rua I','Em preparo'), (10,'Rua
J','Entregue'), (11,'Rua K','Entregue'), (12,'Rua L','Saiu para entrega'), (13,'Rua M','Em preparo'),
(14,'Rua N','Entregue'), (15,'Rua O','Saiu para entrega'), (16,'Rua P','Entregue'), (17,'Rua Q','Em
preparo'), (18,'Rua R','Entregue'), (19,'Rua S','Saiu para entrega'), (20,'Rua T','Entregue');
```

Figura 22 - Consulta Avançada

```
-- ==================================== -- RESERVAS (20) -- ==================================== INSERT
INTO Reservas (id_cliente,id_mesa,data_reserva,hora_reserva,telefone_reserva,taxa_reserva,status,observacao)
VALUES (1,1,'2026-05-01','19:00','8191',5,'Ativa','Janela'), (2,2,'2026-05-
01','20:00','8192',0,'Cancelada',''), (3,3,'2026-05-02','19:30','8193',5,'Reagendada',''), (4,4,'2026-05-
02','21:00','8194',5,'Ativa',''), (5,5,'2026-05-03','19:00','8195',5,'Ativa',''), (6,6,'2026-05-
03','20:00','8196',0,'Cancelada',''), (7,7,'2026-05-04','19:30','8197',5,'Ativa',''), (8,8,'2026-05-
04','21:00','8198',5,'Ativa',''), (9,9,'2026-05-05','19:00','8199',5,'Reagendada',''), (10,10,'2026-05-
```

4.2.8 FASE DE EXECUÇÃO

4.2.9 Quadro Financeiro

Quadro Financeiro: Levantamento de custos de insumos (Calabresa, Massa, etc.) e preços de venda para cálculo de lucro.

Tabela 19: Quadro Financeiro

Ordem	Descrição dos materiais	Unidade	Quantidade	Valor Unitário (R$)	Valor Parcial (R$)
MATERIAL DE EXPEDIENTE					
1	Papel A4	Resma 500 fls	2	R$ 35,00	R$ 70,00
2	Impressão do Relatório	Páginas	1	R$ 60,00	R$ 60,00
3	Encadernação	Unidades	4	R$ 7,00	R$ 28,00
Subtotal (R$)158,00					
MATERIAL PARA PESQUISA					
1	Apostilas Cetam	Disponibilizado	-	R$ 00,00	R$ 00,00
2	Software Xampp	Livre	-	R$ 00,00	R$ 00,00
3	Software brModelo	Livre	-	R$ 00,00	R$ 00,00
4	Software Draw.io	Livre	-	R$ 00,00	R$ 00,00
5	Plataforma Google Drive Sheets	Gratuito	-	R$ 00,00	R$ 00,00
6	Plataforma ChatGPT	Gratuito	-	R$ 00,00	R$ 00,00
7	Plataforma Gemini	Gratuito	-	R$ 00,00	R$ 00,00
8	VS Code	Gratuito	-	R$ 00,00	R$ 00,00
9	Microsoft PowerPoint	Gratuito	-	R$ 00,00	R$ 00,00
10	GitHub	Gratuito	-	R$ 00,00	R$ 00,00
Subtotal (R$) 00,00					
VALOR TOTAL DO PROJETO (R$)158,00					

4.3 CRONOGRAMA: O projeto foi executado entre abril e maio de 2026, passando pelas fases de levantamento de requisitos, modelagem e implementação SQL.

Tabela 20: Cronograma

ATIVIDADES	DATAS/DIAS/SEMANAS									
	1	2	3	4	5	6	7	8	9	10
Definição da temática/área do projeto e Seleção de Material para a pesquisa.	X	X	X	X	X					
Caracterização da Problemática e Elaboração da Justificativa Técnica.		X	X							
Definição da Área de Abrangência.			X							
Definição dos objetivos e das metas do projeto.			X	X						
Definição das fases de execução e da Metodologia.				X	X	X	X	X	X	
Especificação dos órgãos envolvidos com o projeto.					X					
Descrição dos mecanismos e normas de execução.					X	X				
Elaboração do orçamento do projeto.						X				
Aplicação do estudo e Elaboração do Projeto.						X	X	X	X	
Apresentação oral do Projeto.										X

5. CONCLUSÃO

A conclusão do sistema SIGEP consolidou nosso aprendizado prático sobre como organizar e manipular dados de forma profissional. Durante o desenvolvimento, ficou claro que a teoria estudada em sala de aula só ganha vida quando precisamos resolver os problemas reais de um estabelecimento, como o controle de uma pizzaria que possui múltiplos setores funcionando ao mesmo tempo.

Para o nosso grupo, o desafio técnico mais acentuado foi o desenvolvimento dos scripts SQL. A criação dos códigos de criação de tabelas e, principalmente, dos comandos de inserção e consulta, exigiu uma atenção minuciosa. Foi difícil garantir que todas as chaves estrangeiras estivessem conectadas corretamente, evitando que o banco de dados aceitasse informações erradas ou ficasse travado por erros de sintaxe. Passamos bastante tempo revisando cada linha de código no VS Code para assegurar que a estrutura que montamos no papel funcionasse sem falhas dentro do phpMyAdmin.

Por outro lado, tivemos facilidades que ajudaram a compensar essas dificuldades técnicas. Como pontos positivos, entregamos um projeto que realmente funciona, com um banco de dados capaz de registrar pedidos, controlar o estoque de insumos e gerenciar as entregas de delivery com segurança. Como ponto negativo, notamos que o tempo necessário para corrigir os erros de programação foi maior do que o planejado, o que exigiu um esforço redobrado da equipe nas semanas finais. No entanto, o resultado final demonstra que estamos capacitados para atuar na área técnica, compreendendo como um banco de dados deve ser estruturado e mantido para apoiar o dia a dia de uma empresa. O uso do pacote XAMPP foi essencial por ser uma ferramenta prática, que permitiu transformar nossos computadores em servidores locais de forma rápida e estável. Além disso, o apoio de ferramentas de inteligência artificial foi um diferencial, pois nos ajudou a entender erros de código mais rapidamente e a organizar a documentação de maneira mais clara e técnica, funcionando como um suporte extra para nossas pesquisas.

6. REFERÊNCIAS

APACHE FRIENDS. XAMPP. Disponível em: https://www.apachefriends.org/. Acesso em: 4 maio 2026.

BRMODELO. Ferramenta de modelagem de dados relacional. Versão 3.1. Disponível em: http://www.sisweb.com.br/brmodelo/. Acesso em: 4 maio 2026.

CETAM – CENTRO DE EDUCAÇÃO TECNOLÓGICA DO AMAZONAS. Apostilas do curso Técnico em Informática. Manaus: CETAM, 2026.

CETAM – CENTRO DE EDUCAÇÃO TECNOLÓGICA DO AMAZONAS. Material didático da disciplina de Banco de Dados. Manaus: CETAM, 2026.

CETAM – CENTRO DE EDUCAÇÃO TECNOLÓGICA DO AMAZONAS. Projeto Pedagógico de Curso. Manaus, 2026.

DIAGRAMS.NET. Draw.io. [S.l.]: JGraph Ltd, 2026. Disponível em: https://app.diagrams.net/. Acesso em: 5 maio 2026.

GITHUB, INC. GitHub: plataforma de hospedagem e controle de versões de projetos de software. Disponível em: https://github.com/. Acesso em: 6 maio 2026.

GITHUB, INC. GitHub Documentation. Disponível em: https://docs.github.com/. Acesso em: 6 maio 2026.

GOOGLE LLC. Google Docs (Documentos Google). Disponível em: https://workspace.google.com/products/docs/. Acesso em: 4 maio 2026.

GOOGLE LLC. Google Drive – armazenamento em nuvem e sincronização de arquivos. Disponível em: https://www.google.com/drive/. Acesso em: 4 maio 2026.

GOOGLE LLC. Google Sheets (Planilhas Google). Disponível em: https://www.google.com/sheets/about/. Acesso em: 4 maio 2026.

GOOGLE. Gemini: assistente de inteligência artificial generativa. Versão 1.5 Flash. Mountain View, 2026. Disponível em: https://gemini.google.com/. Acesso em: 4 maio 2026.

MICROSOFT. Microsoft PowerPoint. Redmond: Microsoft Corporation, 2026. Software de apresentação.

MICROSOFT. Visual Studio Code (VS Code). Redmond: Microsoft Corporation, 2026. Disponível em: https://code.visualstudio.com/. Acesso em: 4 maio 2026.

MYSQL. MySQL Documentation. Disponível em: https://dev.mysql.com/doc/. Acesso em: 4 maio 2026.

OPENAI. ChatGPT: modelo de linguagem baseado em inteligência artificial. Versão GPT-5.3. San Francisco, 2026. Disponível em: https://chat.openai.com/. Acesso em: 1 maio 2026.

7. APÊNDICE A

REGISTRO DE OBSERVAÇÕES DE PRÁTICAS PROFISSIONAIS

DADOS DOS ESTUDANTES

Estudante 1: Joquebede Barboza Santos Estudante 2: Sophia Roque Martins Estudante 3: Suelen Tavares dos Santos Estudante 4: Suely Nascimento Costa	Turma: _____ Turma: _____ Turma: _____ Turma: _____
Unidade Escolar/Município: Manaus / AM	Curso: Técnico em Informática
Componente Curricular:	
Instrutor Responsável:	

LOCAL DE REALIZAÇÃO DA PRÁTICA PROFISSIONAL

Local:
Setor envolvido (visitado/observado):

REGISTRO DOS PROCEDIMENTOS/FATOS OBSERVADOS

Data da Observação: / /	Horário: de hs às hs
Registros: _____ _____ _____ _____ _____ _____ _____	

Data: ___/___/_____ 1. _____ 2. _____ 3. _____ 4. _____ Assinatura dos Estudantes	Data: ___/___/_____ _____ Assinatura do Instrutor

Observação: Os registros de observações servirão para a elaboração do Relatório Técnico ou do Projeto Técnico.

8. ANEXO

Scripts SQL

```
CREATE DATABASE sigep_pizzaria;

USE sigep_pizzaria;

-- ====================================

– TABELA CREATE

– ADMINISTRATIVO

-- ====================================

CREATE TABLE Cargos (

   id_cargo INT AUTO_INCREMENT PRIMARY KEY COMMENT 'Identificador do cargo',

   nome_cargo VARCHAR(50) NOT NULL COMMENT 'Nome do cargo (Gerente, Garçom, etc.)'

) COMMENT='Tabela de cargos dos funcionários';


CREATE TABLE Funcionarios (

        id_funcionario INT AUTO_INCREMENT PRIMARY KEY COMMENT 'Identificador do
funcionário',

   nome VARCHAR(100) NOT NULL COMMENT 'Nome completo do funcionário',

   usuario VARCHAR(50) UNIQUE COMMENT 'Login do sistema',

   senha VARCHAR(255) COMMENT 'Senha de acesso',

   id_cargo INT COMMENT 'Cargo do funcionário',

   turno ENUM('Manhã','Tarde','Noite') COMMENT 'Turno de trabalho',

   FOREIGN KEY (id_cargo) REFERENCES Cargos(id_cargo)

) COMMENT='Tabela de funcionários da pizzaria';


-- ====================================

-- DISPOSITIVOS

-- ====================================
```

```
CREATE TABLE Dispositivos (
        id_dispositivo INT AUTO_INCREMENT PRIMARY KEY COMMENT 'Identificador do dispositivo',
   tipo ENUM(
      'TV Cozinha',
      'Painel Expedição',
      'TV Expedição',
      'Computador',
      'Maquininha Cartão',
      'Celular',
      'Totem'
   ) COMMENT 'Tipo do dispositivo utilizado no sistema',
   localizacao VARCHAR(100) COMMENT 'Local onde o dispositivo está',
   status ENUM('Ativo','Inativo') DEFAULT 'Ativo' COMMENT 'Status do dispositivo'
) COMMENT='Dispositivos utilizados no funcionamento da pizzaria';


-- ======================================
-- CLIENTES
-- ======================================
CREATE TABLE Clientes (
   id_cliente INT AUTO_INCREMENT PRIMARY KEY COMMENT 'Identificador do cliente',
   nome VARCHAR(100) COMMENT 'Nome do cliente',
   telefone VARCHAR(20) COMMENT 'Telefone para contato',
   email VARCHAR(100) COMMENT 'Email do cliente',
   endereco VARCHAR(200) COMMENT 'Endereço para entrega'
) COMMENT='Cadastro de clientes';
-- ======================================
```

```
-- MESAS
-- ====================================

CREATE TABLE Mesas (
    id_mesa INT AUTO_INCREMENT PRIMARY KEY COMMENT 'Identificador da mesa',
    numero_mesa INT COMMENT 'Número físico da mesa',
    andar ENUM('1º','2º') COMMENT 'Andar da pizzaria',
    status ENUM('Disponível','Indisponível','Reservada') COMMENT 'Situação da mesa'
) COMMENT='Mesas físicas da pizzaria';
-- ====================================
-- RESERVAS
-- ====================================
CREATE TABLE Reservas (
    id_reserva INT AUTO_INCREMENT PRIMARY KEY COMMENT 'Identificador da reserva',
    id_cliente INT COMMENT 'Cliente que fez a reserva',
    id_mesa INT COMMENT 'Mesa reservada',
    data_reserva DATE COMMENT 'Data da reserva',
    hora_reserva TIME COMMENT 'Horário da reserva',
    telefone_reserva VARCHAR(20) COMMENT 'Telefone de confirmação',
    taxa_reserva DECIMAL(10,2) COMMENT 'Valor da taxa de reserva',
    status ENUM('Ativa','Cancelada','Reagendada') COMMENT 'Status da reserva',
    observacao VARCHAR(200) COMMENT 'Observações da reserva',
    FOREIGN KEY (id_cliente) REFERENCES Clientes(id_cliente),
    FOREIGN KEY (id_mesa) REFERENCES Mesas(id_mesa)
) COMMENT='Controle de reservas de mesas';

-- ====================================
```

```
-- PRODUTOS
-- ====================================
CREATE TABLE Produtos (
    id_produto INT AUTO_INCREMENT PRIMARY KEY COMMENT 'Identificador do produto',
    nome_produto VARCHAR(100) COMMENT 'Nome do produto',
    categoria VARCHAR(50) COMMENT 'Categoria (Pizza, Bebida, etc.)',
    tamanho ENUM('Broto','Média','Grande','Família') COMMENT 'Tamanho do produto',
    preco DECIMAL(10,2) COMMENT 'Preço do produto'
) COMMENT='Cardápio da pizzaria';
-- ====================================
-- INSUMOS / ESTOQUE
-- ====================================
CREATE TABLE Insumos (
    id_insumo INT AUTO_INCREMENT PRIMARY KEY COMMENT 'Identificador do insumo',
    nome_insumo VARCHAR(100) COMMENT 'Nome do insumo',
    unidade ENUM('ml','L','g','kg','unidade') COMMENT 'Unidade de medida'
) COMMENT='Matéria-prima utilizada na produção';


CREATE TABLE Estoque (
    id_estoque INT AUTO_INCREMENT PRIMARY KEY COMMENT 'Identificador do estoque',
    id_insumo INT COMMENT 'Insumo armazenado',
    quantidade DECIMAL(10,2) COMMENT 'Quantidade disponível',
    data_entrada DATE COMMENT 'Data de entrada no estoque',
    data_saida DATE COMMENT 'Data de saída do estoque',
    FOREIGN KEY (id_insumo) REFERENCES Insumos(id_insumo)
) COMMENT='Controle de estoque de insumos';
-- ====================================
```

```
-- PEDIDOS
-- ====================================
CREATE TABLE Pedidos (
    id_pedido INT AUTO_INCREMENT PRIMARY KEY COMMENT 'Identificador do pedido',
    id_cliente INT COMMENT 'Cliente que realizou o pedido',
    id_mesa INT NULL COMMENT 'Mesa (se presencial)',
    tipo ENUM('Presencial','Delivery') COMMENT 'Tipo do pedido',

    origem ENUM(
        'WhatsApp',
        'iFood',
        'Rappi',
        'Uber Eats',
        'Site',
        'Balcão',
        'Totem'
    ) COMMENT 'Origem do pedido',

    status ENUM(
        'Pedido recebido',
        'Em fila',
        'Em preparo',
        'No forno',
        'Finalizado',
        'Pronto',
        'Saiu para entrega',
        'Servindo mesa'
```

```
    ) COMMENT 'Status do pedido',

        data_hora DATETIME DEFAULT CURRENT_TIMESTAMP COMMENT 'Data e hora do pedido',

    FOREIGN KEY (id_cliente) REFERENCES Clientes(id_cliente),
    FOREIGN KEY (id_mesa) REFERENCES Mesas(id_mesa)
) COMMENT='Pedidos realizados na pizzaria';
-- ======================================
-- ITENS DO PEDIDO
-- ======================================
CREATE TABLE Itens_Pedido (
    id_item INT AUTO_INCREMENT PRIMARY KEY COMMENT 'Item do pedido',
    id_pedido INT COMMENT 'Pedido relacionado',
    id_produto INT COMMENT 'Produto do pedido',
    quantidade INT COMMENT 'Quantidade do item',
    observacao VARCHAR(200) COMMENT 'Observações do cliente',
    FOREIGN KEY (id_pedido) REFERENCES Pedidos(id_pedido),
    FOREIGN KEY (id_produto) REFERENCES Produtos(id_produto)
) COMMENT='Itens que compõem cada pedido';
-- ======================================
-- PAGAMENTOS
-- ======================================
CREATE TABLE Pagamentos (
        id_pagamento INT AUTO_INCREMENT PRIMARY KEY COMMENT 'Identificador do pagamento',
    id_pedido INT COMMENT 'Pedido pago',
    tipo_pagamento ENUM(
```

```
        'Dinheiro',
        'Cartão Crédito',
        'Cartão Débito',
        'Pix',
        'Carteira Digital',
        'iFood',
        'Rappi',
        'Uber Eats'
    ) COMMENT 'Forma de pagamento',
    valor DECIMAL(10,2) COMMENT 'Valor pago',
    status ENUM('Pago','Pendente') COMMENT 'Situação do pagamento',
    FOREIGN KEY (id_pedido) REFERENCES Pedidos(id_pedido)
) COMMENT='Pagamentos dos pedidos';
-- ====================================
-- ENTREGAS
-- ====================================
CREATE TABLE Entregas (
    id_entrega INT AUTO_INCREMENT PRIMARY KEY COMMENT 'Identificador da entrega',
    id_pedido INT COMMENT 'Pedido entregue',
    endereco_entrega VARCHAR(200) COMMENT 'Endereço da entrega',
    status ENUM(
        'Pedido recebido',
        'Em preparo',
        'Saiu para entrega',
        'Entregue'
    ) COMMENT 'Status da entrega',
    FOREIGN KEY (id_pedido) REFERENCES Pedidos(id_pedido)
```

```
) COMMENT='Controle de entregas';
USE sigep_pizzaria;
-- ====================================
-- TABELA INSERTS
– CARGOS
-- ====================================
INSERT INTO Cargos (nome_cargo) VALUES
('Gerente'),('Pizzaiolo'),('Auxiliar Cozinha'),('Caixa'),
('Garçom'),('Entregador'),('Limpeza'),('Supervisor'),
('Atendente'),('Estoquista');
-- ====================================
-- FUNCIONÁRIOS (12)
-- ====================================
INSERT INTO Funcionarios (nome, usuario, senha, id_cargo, turno) VALUES
('Claudio','gerente','123',1,'Manhã'),
('Marcos','p1','123',2,'Noite'),
('André','p2','123',2,'Noite'),
('Ana','c1','123',3,'Manhã'),
('Bruno','c2','123',3,'Tarde'),
('Carla','cx1','123',4,'Tarde'),
('Renata','cx2','123',4,'Noite'),
('José','g1','123',5,'Noite'),
('Lucas','g2','123',5,'Tarde'),
('Ruan','e1','123',6,'Noite'),
('Sonia','l1','123',7,'Manhã'),
('Lucas2','s1','123',8,'Manhã');
-- ====================================
```

```
-- DISPOSITIVOS
-- ====================================
INSERT INTO Dispositivos (tipo, localizacao, status) VALUES
('TV Cozinha','Cozinha 1','Ativo'),
('TV Cozinha','Cozinha 2','Ativo'),
('Painel Expedição','Expedição','Ativo'),
('Maquininha Cartão','Caixa','Ativo'),
('Maquininha Cartão','Caixa','Ativo'),
('Celular','Salão','Ativo'),
('Celular','Salão','Ativo'),
('Celular','Delivery','Ativo'),
('Celular','Delivery','Ativo'),
('Totem','Entrada','Ativo'),
('Totem','Entrada','Ativo'),
('TV Expedição','Bar','Ativo'),
('Maquininha Cartão','Salão VIP','Ativo'),
('Computador','Administração','Ativo'),
('Celular','Gestão','Ativo'),
('TV Expedição','Salão','Ativo'),
('Celular','Cozinha','Ativo'),
('Maquininha Cartão','Reserva','Ativo'),
('Celular','Backup','Ativo'),
('Totem','Entrada Externa','Ativo');
-- ====================================
-- CLIENTES (20)
-- ====================================
INSERT INTO Clientes (nome, telefone, email, endereco) VALUES
```

```
('Cliente 1','8191','c1@email.com','Rua A'),
('Cliente 2','8192','c2@email.com','Rua B'),
('Cliente 3','8193','c3@email.com','Rua C'),
('Cliente 4','8194','c4@email.com','Rua D'),
('Cliente 5','8195','c5@email.com','Rua E'),
('Cliente 6','8196','c6@email.com','Rua F'),
('Cliente 7','8197','c7@email.com','Rua G'),
('Cliente 8','8198','c8@email.com','Rua H'),
('Cliente 9','8199','c9@email.com','Rua I'),
('Cliente 10','8200','c10@email.com','Rua J'),
('Cliente 11','8201','c11@email.com','Rua K'),
('Cliente 12','8202','c12@email.com','Rua L'),
('Cliente 13','8203','c13@email.com','Rua M'),
('Cliente 14','8204','c14@email.com','Rua N'),
('Cliente 15','8205','c15@email.com','Rua O'),
('Cliente 16','8206','c16@email.com','Rua P'),
('Cliente 17','8207','c17@email.com','Rua Q'),
('Cliente 18','8208','c18@email.com','Rua R'),
('Cliente 19','8209','c19@email.com','Rua S'),
('Cliente 20','8210','c20@email.com','Rua T');
-- ====================================
-- MESAS (10)
-- ====================================
INSERT INTO Mesas (numero_mesa, andar, status) VALUES
(1,'1º','Disponível'),(2,'1º','Reservada'),(3,'1º','Disponível'),
(4,'1º','Indisponível'),(5,'1º','Reservada'),
(6,'2º','Disponível'),(7,'2º','Reservada'),
```

```
(8,'2º','Disponível'),(9,'2º','Indisponível'),(10,'2º','Disponível');

-- ====================================

-- PRODUTOS

-- ====================================

INSERT INTO Produtos (nome_produto,categoria,tamanho,preco) VALUES

('Calabresa','Pizza','Grande',45),

('Mussarela','Pizza','Grande',40),

('Portuguesa','Pizza','Grande',50),

('Frango Catupiry','Pizza','Grande',48),

('Chocolate','Pizza','Broto',30),

('Romeu e Julieta','Pizza','Broto',28),

('Pepperoni','Pizza','Grande',55),

('4 Queijos','Pizza','Grande',52),

('Vegetariana','Pizza','Grande',49),

('Lombo','Pizza','Grande',53),

('Refrigerante','Bebida','Grande',12),

('Suco','Bebida','Média',10),

('Água','Bebida','Broto',5),

('Cerveja','Bebida','Grande',15),

('Batata Frita','Acompanhamento','Média',18),

('Nuggets','Acompanhamento','Média',20),

('Pudim','Sobremesa','Broto',12),

('Brownie','Sobremesa','Broto',15),

('Açaí','Sobremesa','Média',18),

('Borda Recheada','Adicional','Broto',8);
```

```
-- ====================================
-- INSUMOS
-- ====================================
INSERT INTO Insumos (nome_insumo, unidade) VALUES
('Calabresa','kg'),('Mussarela','kg'),('Presunto','kg'),('Frango','kg'),
('Chocolate','kg'),('Goiabada','kg'),('Pepperoni','kg'),('Queijo','kg'),
('Vegetais','kg'),('Lombo','kg'),('Refrigerante','L'),('Suco','L'),
('Água','unidade'),('Cerveja','unidade'),('Batata','kg'),('Nuggets','kg'),
('Pudim','kg'),('Brownie','kg'),('Açaí','kg'),('Massa','kg');
-- ====================================
-- ESTOQUE
-- ====================================
INSERT INTO Estoque (id_insumo, quantidade, data_entrada, data_saida) VALUES
(1,20,'2026-05-01',NULL),(2,30,'2026-05-01',NULL),
(3,15,'2026-05-01',NULL),(4,25,'2026-05-01',NULL),
(5,10,'2026-05-01',NULL),(6,12,'2026-05-01',NULL),
(7,18,'2026-05-01',NULL),(8,20,'2026-05-01',NULL),
(9,15,'2026-05-01',NULL),(10,14,'2026-05-01',NULL),
(11,40,'2026-05-01',NULL),(12,20,'2026-05-01',NULL),
(13,50,'2026-05-01',NULL),(14,60,'2026-05-01',NULL),
(15,10,'2026-05-01',NULL),(16,20,'2026-05-01',NULL),
(17,12,'2026-05-01',NULL),(18,10,'2026-05-01',NULL),
(19,15,'2026-05-01',NULL),(20,50,'2026-05-01',NULL);
-- ====================================
-- PEDIDOS (20)
-- ====================================
INSERT INTO Pedidos (id_cliente,id_mesa,tipo,origem,status) VALUES
```

```
(1,1,'Presencial','Totem','Finalizado'),
(2,2,'Presencial','Balcão','Em preparo'),
(3,3,'Presencial','Site','Servindo mesa'),
(4,4,'Delivery','iFood','Saiu para entrega'),
(5,5,'Delivery','Uber Eats','Finalizado'),
(6,6,'Delivery','Rappi','Em preparo'),
(7,7,'Presencial','Totem','Finalizado'),
(8,8,'Presencial','Balcão','Servindo mesa'),
(9,9,'Presencial','Site','Em preparo'),
(10,10,'Delivery','iFood','Saiu para entrega'),
(11,1,'Presencial','Totem','Finalizado'),
(12,2,'Presencial','Balcão','Em preparo'),
(13,3,'Delivery','iFood','Saiu para entrega'),
(14,4,'Delivery','Rappi','Em preparo'),
(15,5,'Presencial','Site','Finalizado'),
(16,6,'Delivery','Uber Eats','Saiu para entrega'),
(17,7,'Presencial','Totem','Servindo mesa'),
(18,8,'Presencial','Balcão','Finalizado'),
(19,9,'Delivery','iFood','Em preparo'),
(20,10,'Delivery','Rappi','Finalizado');
-- ====================================
-- ITENS PEDIDO (20)
-- ====================================
INSERT INTO Itens_Pedido (id_pedido,id_produto,quantidade,observacao) VALUES
(1,1,1,'Sem cebola'),(2,2,1,'Borda recheada'),
(3,3,1,'Extra queijo'),(4,4,1,'Sem tomate'),
(5,5,1,'Bem assada'),(6,6,1,'Pouco sal'),
```

```
(7,7,1,'Sem orégano'),(8,8,1,'Molho extra'),
(9,9,1,'Sem azeite'),(10,10,1,'Com gelo'),
(11,1,2,'Família'),(12,2,1,'Normal'),
(13,3,1,'Sem borda'),(14,4,1,'Extra frango'),
(15,5,1,'Doce leve'),(16,6,1,'Sem açúcar'),
(17,7,1,'Apimentada'),(18,8,1,'Sem queijo'),
(19,9,1,'Vegana'),(20,10,1,'Caprichada');
-- ====================================
-- PAGAMENTOS
-- ====================================
INSERT INTO Pagamentos (id_pedido,tipo_pagamento,valor,status) VALUES
(1,'Dinheiro',45,'Pago'),(2,'Pix',40,'Pago'),
(3,'Cartão Crédito',50,'Pago'),(4,'Cartão Débito',48,'Pago'),
(5,'iFood',30,'Pago'),(6,'Uber Eats',28,'Pago'),
(7,'Rappi',32,'Pago'),(8,'Carteira Digital',25,'Pago'),
(9,'Pix',20,'Pago'),(10,'Cartão Crédito',55,'Pago'),
(11,'Dinheiro',45,'Pago'),(12,'Pix',40,'Pago'),
(13,'Cartão Débito',38,'Pago'),(14,'iFood',60,'Pago'),
(15,'Uber Eats',42,'Pago'),(16,'Rappi',35,'Pago'),
(17,'Pix',25,'Pago'),(18,'Cartão Crédito',50,'Pago'),
(19,'Dinheiro',30,'Pago'),(20,'Carteira Digital',22,'Pago');
-- ====================================
-- ENTREGAS (20)
-- ====================================
INSERT INTO Entregas (id_pedido,endereco_entrega,status) VALUES
(1,'Rua A','Entregue'),
(2,'Rua B','Saiu para entrega'),
```

```
(3,'Rua C','Em preparo'),

(4,'Rua D','Entregue'),

(5,'Rua E','Saiu para entrega'),

(6,'Rua F','Em preparo'),

(7,'Rua G','Entregue'),

(8,'Rua H','Saiu para entrega'),

(9,'Rua I','Em preparo'),

(10,'Rua J','Entregue'),

(11,'Rua K','Entregue'),

(12,'Rua L','Saiu para entrega'),

(13,'Rua M','Em preparo'),

(14,'Rua N','Entregue'),

(15,'Rua O','Saiu para entrega'),

(16,'Rua P','Entregue'),

(17,'Rua Q','Em preparo'),

(18,'Rua R','Entregue'),

(19,'Rua S','Saiu para entrega'),

(20,'Rua T','Entregue');

-- ====================================

-- RESERVAS (20)

-- ====================================

INSERT INTO Reservas

(id_cliente,id_mesa,data_reserva,hora_reserva,telefone_reserva,taxa_reserva,status,observacao
)

VALUES

(1,1,'2026-05-01','19:00','8191',5,'Ativa','Janela'),

(2,2,'2026-05-01','20:00','8192',0,'Cancelada',''),
```

```
(3,3,'2026-05-02','19:30','8193',5,'Reagendada',''),
(4,4,'2026-05-02','21:00','8194',5,'Ativa',''),
(5,5,'2026-05-03','19:00','8195',5,'Ativa',''),
(6,6,'2026-05-03','20:00','8196',0,'Cancelada',''),
(7,7,'2026-05-04','19:30','8197',5,'Ativa',''),
(8,8,'2026-05-04','21:00','8198',5,'Ativa',''),
(9,9,'2026-05-05','19:00','8199',5,'Reagendada',''),
(10,10,'2026-05-05','20:30','8200',5,'Ativa',''),
(11,1,'2026-05-06','19:00','8201',5,'Ativa',''),
(12,2,'2026-05-06','20:00','8202',0,'Cancelada',''),
(13,3,'2026-05-07','19:30','8203',5,'Ativa',''),
(14,4,'2026-05-07','21:00','8204',5,'Ativa',''),
(15,5,'2026-05-08','19:00','8205',5,'Reagendada',''),
(16,6,'2026-05-08','20:00','8206',5,'Ativa',''),
(17,7,'2026-05-09','19:30','8207',0,'Cancelada',''),
(18,8,'2026-05-09','21:00','8208',5,'Ativa',''),
(19,9,'2026-05-10','19:00','8209',5,'Ativa',''),
(20,10,'2026-05-10','20:30','8210',5,'Ativa','');
```

```
USE sigep_pizzaria;

-- ====================================
-- TABELA SELECT
-- ====================================
SELECT * FROM Cargos;
SELECT * FROM Funcionarios;
SELECT * FROM Clientes;
SELECT * FROM Mesas;
SELECT * FROM Produtos;
SELECT * FROM Insumos;
SELECT * FROM Estoque;
SELECT * FROM Pedidos;
SELECT * FROM Itens_Pedido;
SELECT * FROM Pagamentos;
```

```
SELECT * FROM Entregas;
SELECT * FROM Reservas;
```

Figura 23 - Consulta Avançada

Autora: Suelen Tavares

```
-- ====================================
-- PEDIDO COMPLETO (COM ITENS E CLIENTE)
-- ====================================
SELECT
    p.id_pedido,
    c.nome AS cliente,
    m.numero_mesa,
    p.tipo,
    p.origem,
    p.status,
    pr.nome_produto,
    i.quantidade,
    i.observacao
FROM Pedidos p
JOIN Clientes c ON p.id_cliente = c.id_cliente
LEFT JOIN Mesas m ON p.id_mesa = m.id_mesa
JOIN Itens_Pedido i ON p.id_pedido = i.id_pedido
JOIN Produtos pr ON i.id_produto = pr.id_produto
ORDER BY p.id_pedido;
```

Figura 24 - Consulta Avançada

```
SELECT p.id_pedido, c.nome AS cliente, m.numero_mesa, p.tipo, p.origem, p.status, pr.nome_produto, i.quantidade,
i.observacao FROM Pedidos p JOIN Clientes c ON p.id_cliente = c.id_cliente LEFT JOIN Mesas m ON p.id_mesa =
m.id_mesa JOIN Itens_Pedido i ON p.id_pedido = i.id_pedido JOIN Produtos pr ON i.id_produto = pr.id_produto ORDER BY
p.id_pedido;
```

Perfil [Editar em linha] [Editar] [Demonstrar SQL] [Criar código PHP] [Atualizar]

Mostrar tudo | Número de linhas: 25 Filtrar linhas: Procurar nesta tabela

Opções extras

d_pedido 1	cliente	numero_mesa	tipo	origem	status	nome_produto	quantidade	observacao
1	Cliente 1	1	Presencial	Totem	Finalizado	Calabresa	1	Sem cebola
2	Cliente 2	2	Presencial	Balcão	Em preparo	Mussarela	1	Borda recheada
3	Cliente 3	3	Presencial	Site	Servindo mesa	Portuguesa	1	Extra queijo
4	Cliente 4	4	Delivery	iFood	Saiu para entrega	Frango Catupiry	1	Sem tomate
5	Cliente 5	5	Delivery	Uber Eats	Finalizado	Chocolate	1	Bem assada
6	Cliente 6	6	Delivery	Rappi	Em preparo	Romeu e Julieta	1	Pouco sal
7	Cliente 7	7	Presencial	Totem	Finalizado	Pepperoni	1	Sem orégano
8	Cliente 8	8	Presencial	Balcão	Servindo mesa	4 Queijos	1	Molho extra
9	Cliente 9	9	Presencial	Site	Em preparo	Vegetariana	1	Sem azeite
10	Cliente 10	10	Delivery	iFood	Saiu para entrega	Lombo	1	Com gelo
11	Cliente 11	1	Presencial	Totem	Finalizado	Calabresa	2	Família
12	Cliente 12	2	Presencial	Balcão	Em preparo	Mussarela	1	Normal
13	Cliente 13	3	Delivery	iFood	Saiu para entrega	Portuguesa	1	Sem borda
14	Cliente 14	4	Delivery	Rappi	Em preparo	Frango Catupiry	1	Extra frango
15	Cliente 15	5	Presencial	Site	Finalizado	Chocolate	1	Doce leve
16	Cliente 16	6	Delivery	Uber Eats	Saiu para entrega	Romeu e Julieta	1	Sem açúcar
17	Cliente 17	7	Presencial	Totem	Servindo mesa	Pepperoni	1	Apimentada
18	Cliente 18	8	Presencial	Balcão	Finalizado	4 Queijos	1	Sem queijo
19	Cliente 19	9	Delivery	iFood	Em preparo	Vegetariana	1	Vegana
20	Cliente 20	10	Delivery	Rappi	Finalizado	Lombo	1	Caprichada

Autora: Suelen Tavares

```
-- ====================================
-- ENTREGAS COM CLIENTE
-- ====================================
SELECT
    e.id_entrega,
    p.id_pedido,
    c.nome AS cliente,
    e.endereco_entrega,
    e.status
FROM Entregas e
JOIN Pedidos p ON e.id_pedido = p.id_pedido
JOIN Clientes c ON p.id_cliente = c
.id_cliente;
```

Figura 25 - Consulta Avançada

```
SELECT e.id_entrega, p.id_pedido, c.nome AS cliente, e.endereco_entrega, e.status FROM Entregas e JOIN Pedidos p ON
e.id_pedido = p.id_pedido JOIN Clientes c ON p.id_cliente = c.id_cliente;
```

Perfil [Editar em linha] [Editar] [Demonstrar SQL] [Criar código PHP] [Atualizar]

Mostrar tudo | Número de linhas: 25 Filtrar linhas: Procurar nesta tabela

Opções extras

id_entrega	id_pedido	cliente	endereco_entrega	status
1	1	Cliente 1	Rua A	Entregue
2	2	Cliente 2	Rua B	Saiu para entrega
3	3	Cliente 3	Rua C	Em preparo
4	4	Cliente 4	Rua D	Entregue
5	5	Cliente 5	Rua E	Saiu para entrega
6	6	Cliente 6	Rua F	Em preparo
7	7	Cliente 7	Rua G	Entregue
8	8	Cliente 8	Rua H	Saiu para entrega
9	9	Cliente 9	Rua I	Em preparo
10	10	Cliente 10	Rua J	Entregue
11	11	Cliente 11	Rua K	Entregue
12	12	Cliente 12	Rua L	Saiu para entrega
13	13	Cliente 13	Rua M	Em preparo
14	14	Cliente 14	Rua N	Entregue
15	15	Cliente 15	Rua O	Saiu para entrega
16	16	Cliente 16	Rua P	Entregue
17	17	Cliente 17	Rua Q	Em preparo
18	18	Cliente 18	Rua R	Entregue
19	19	Cliente 19	Rua S	Saiu para entrega
20	20	Cliente 20	Rua T	Entregue

Autora: Suelen Tavares

```
-- ====================================
-- RESERVAS COM CLIENTE E MESA
-- ====================================
SELECT
    r.id_reserva,
    c.nome AS cliente,
    m.numero_mesa,
    r.data_reserva,
    r.hora_reserva,
    r.status
FROM Reservas r
JOIN Clientes c ON r.id_cliente = c.id_cliente
JOIN Mesas m ON r.id_mesa = m.id_mesa;
```

Figura 26 - Consulta Avançada

```
SELECT r.id_reserva, c.nome AS cliente, m.numero_mesa, r.data_reserva, r.hora_reserva, r.status FROM
Reservas r JOIN Clientes c ON r.id_cliente = c.id_cliente JOIN Mesas m ON r.id_mesa = m.id_mesa;
```

Perfil [Editar em linha] [Editar] [Demonstrar SQL] [Criar código PHP] [Atualizar]

Mostrar tudo | Número de linhas: 25 Filtrar linhas: Procurar nesta tabela

Opções extras

id_reserva	cliente	numero_mesa	data_reserva	hora_reserva	status
1	Cliente 1	1	2026-05-01	19:00:00	Ativa
11	Cliente 11	1	2026-05-06	19:00:00	Ativa
2	Cliente 2	2	2026-05-01	20:00:00	Cancelada
12	Cliente 12	2	2026-05-06	20:00:00	Cancelada
3	Cliente 3	3	2026-05-02	19:30:00	Reagendada
13	Cliente 13	3	2026-05-07	19:30:00	Ativa
4	Cliente 4	4	2026-05-02	21:00:00	Ativa
14	Cliente 14	4	2026-05-07	21:00:00	Ativa
5	Cliente 5	5	2026-05-03	19:00:00	Ativa
15	Cliente 15	5	2026-05-08	19:00:00	Reagendada
6	Cliente 6	6	2026-05-03	20:00:00	Cancelada
16	Cliente 16	6	2026-05-08	20:00:00	Ativa
7	Cliente 7	7	2026-05-04	19:30:00	Ativa
17	Cliente 17	7	2026-05-09	19:30:00	Cancelada
8	Cliente 8	8	2026-05-04	21:00:00	Ativa
18	Cliente 18	8	2026-05-09	21:00:00	Ativa
9	Cliente 9	9	2026-05-05	19:00:00	Reagendada
19	Cliente 19	9	2026-05-10	19:00:00	Ativa
10	Cliente 10	10	2026-05-05	20:30:00	Ativa
20	Cliente 20	10	2026-05-10	20:30:00	Ativa

Autora: Suelen Tavares

```
-- ====================================
-- ESTOQUE COM NOME DO INSUMO
-- ====================================

SELECT
    i.nome_insumo,
    e.quantidade,
    e.data_entrada
FROM Estoque e
JOIN Insumos i ON e.id_insumo = i.id_insumo;
```

Figura 27 - Consulta Avançada

```
SELECT i.nome_insumo, e.quantidade, e.data_entrada FROM Estoque e JOIN
Insumos i ON e.id_insumo = i.id_insumo;
```

Perfil [Editar em linha] [Editar] [Demonstrar SQL] [Criar código PHP] [Atualizar]

Mostrar tudo | Número de linhas: 25 Filtrar linhas: Procurar nesta t

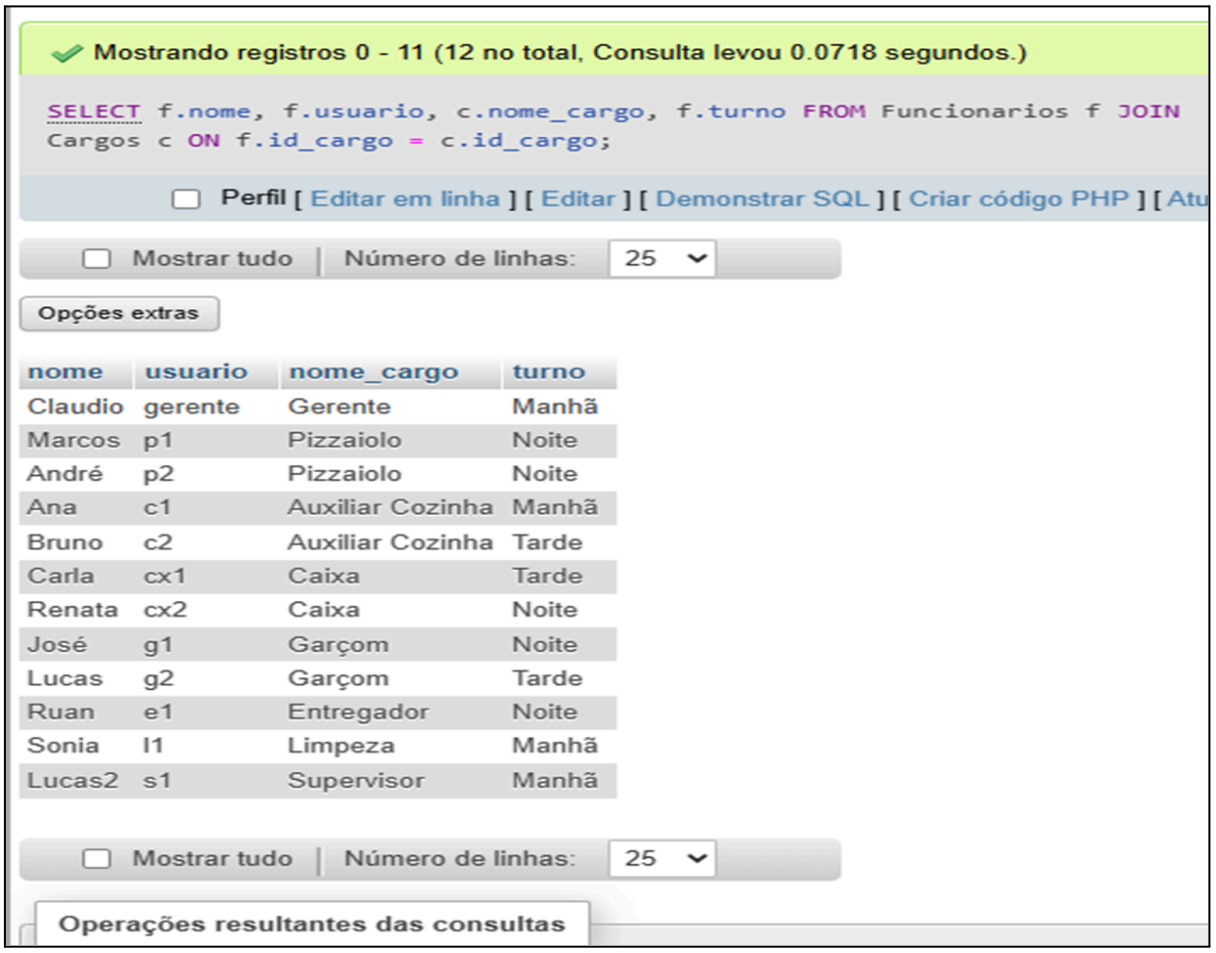
Opções extras

nome_insumo	quantidade	data_entrada
Calabresa	20.00	2026-05-01
Mussarela	30.00	2026-05-01
Presunto	15.00	2026-05-01
Frango	25.00	2026-05-01
Chocolate	10.00	2026-05-01
Goiabada	12.00	2026-05-01
Pepperoni	18.00	2026-05-01
Queijo	20.00	2026-05-01
Vegetais	15.00	2026-05-01
Lombo	14.00	2026-05-01
Refrigerante	40.00	2026-05-01
Suco	20.00	2026-05-01
Água	50.00	2026-05-01
Cerveja	60.00	2026-05-01
Batata	10.00	2026-05-01
Nuggets	20.00	2026-05-01
Pudim	12.00	2026-05-01
Brownie	10.00	2026-05-01
Açaí	15.00	2026-05-01
Massa	50.00	2026-05-01

Autora: Suelen Tavares

```
-- ====================================
-- FUNCIONÁRIOS COM CARGO
-- ====================================
SELECT
    f.nome,
    f.usuario,
    c.nome_cargo,
    f.turno
FROM Funcionarios f
JOIN Cargos c ON f.id_cargo = c.id_cargo;
```

Figura 28 - Consulta Avançada



nome	usuario	nome_cargo	turno
Claudio	gerente	Gerente	Manhã
Marcos	p1	Pizzaiolo	Noite
André	p2	Pizzaiolo	Noite
Ana	c1	Auxiliar Cozinha	Manhã
Bruno	c2	Auxiliar Cozinha	Tarde
Carla	cx1	Caixa	Tarde
Renata	cx2	Caixa	Noite
José	g1	Garçom	Noite
Lucas	g2	Garçom	Tarde
Ruan	e1	Entregador	Noite
Sonia	l1	Limpeza	Manhã
Lucas2	s1	Supervisor	Manhã

Autora: Suelen Tavares

```
-- ====================================
-- PEDIDOS COM PAGAMENTO
-- ====================================
SELECT
    p.id_pedido,
    c.nome AS cliente,
    p.tipo,
    p.status,
    pg.tipo_pagamento,
    pg.valor,
    pg.status AS status_pagamento
FROM Pedidos p
JOIN Clientes c ON p.id_cliente = c.id_cliente
JOIN Pagamentos pg ON p.id_pedido = pg.id_pedido;
```

Figura 29 - Consulta Avançada

```
SELECT p.id_pedido, c.nome AS cliente, p.tipo, p.status, pg.tipo_pagamento, pg.valor, pg.status AS
status_pagamento FROM Pedidos p JOIN Clientes c ON p.id_cliente = c.id_cliente JOIN Pagamentos pg
ON p.id_pedido = pg.id_pedido;
```

Perfil [Editar em linha] [Editar] [Demonstrar SQL] [Criar código PHP] [Atualizar

Mostrar tudo | Número de linhas: 25 Filtrar linhas: Procurar nesta tabela

Opções extras

id_pedido	cliente	tipo	status	tipo_pagamento	valor	status_pagamento
1	Cliente 1	Presencial	Finalizado	Dinheiro	45.00	Pago
2	Cliente 2	Presencial	Em preparo	Pix	40.00	Pago
3	Cliente 3	Presencial	Servindo mesa	Cartão Crédito	50.00	Pago
4	Cliente 4	Delivery	Saiu para entrega	Cartão Débito	48.00	Pago
5	Cliente 5	Delivery	Finalizado	iFood	30.00	Pago
6	Cliente 6	Delivery	Em preparo	Uber Eats	28.00	Pago
7	Cliente 7	Presencial	Finalizado	Rappi	32.00	Pago
8	Cliente 8	Presencial	Servindo mesa	Carteira Digital	25.00	Pago
9	Cliente 9	Presencial	Em preparo	Pix	20.00	Pago
10	Cliente 10	Delivery	Saiu para entrega	Cartão Crédito	55.00	Pago
11	Cliente 11	Presencial	Finalizado	Dinheiro	45.00	Pago
12	Cliente 12	Presencial	Em preparo	Pix	40.00	Pago
13	Cliente 13	Delivery	Saiu para entrega	Cartão Débito	38.00	Pago
14	Cliente 14	Delivery	Em preparo	iFood	60.00	Pago
15	Cliente 15	Presencial	Finalizado	Uber Eats	42.00	Pago
16	Cliente 16	Delivery	Saiu para entrega	Rappi	35.00	Pago
17	Cliente 17	Presencial	Servindo mesa	Pix	25.00	Pago
18	Cliente 18	Presencial	Finalizado	Cartão Crédito	50.00	Pago
19	Cliente 19	Delivery	Em preparo	Dinheiro	30.00	Pago
20	Cliente 20	Delivery	Finalizado	Carteira Digital	22.00	Pago

Autora: Suelen Tavares