# CENTRO UNIVERSITÁRIO BRASILEIRO (UNIBRA)

# ***Análise e Desenvolvimento de Sistemas***

# **Sistema web Sistema de gerenciamento de pedidos online (SGP) Para restaurante Mizu Oriente.**

**Recife -PE**

**2025**

**Sistema web Sistema de gerenciamento de pedidos online (SGP)**

Projeto apresentado ao curso de Análise e Desenvolvimento de Software, do Centro Universitário Brasileiro (UNIBRA). como parte dos Requisitos para obtenção do Título de Grau tecnólogo.

**Orientadora:** Pricylla Santos

**Recife -PE**

**2025**

SUMÁRIO

Introdução............................................................................ 4  
1.1 Propósito........................................................................ 4  
1.2 Escopo............................................................................ 4  
1.3 Definições, Acrônimos e Abreviações........................... 5

DESCRIÇÃO GERAL.......................................................... 5  
2.1 Visão Geral do Sistema................................................... 5  
2.2 Funções do Sistema......................................................... 5  
2.3 Restrições........................................................................ 6

REQUISITOS ESPECÍFICOS............................................. 6  
3.1 Requisitos Funcionais..................................................... 6  
3.1.1 Gestão de Usuários...................................................... 6  
3.1.2 Cardápio do Restaurante.............................................. 6  
3.1.3 Histórico de Pedidos.................................................... 7  
3.2 Requisitos Não Funcionais............................................. 7

3.3Desempenho....................................................................7

3.4Segurança........................................................................7

3.5Usabilidade......................................................................7

3.6Confiabilidade e Disponibilidade....................................7

3.7 Escalabilidade................................................................8

3.7 Compatibilidade.............................................................8

3.8 Manutabilidade...............................................................8

3.9Auditabilidade..................................................................8

4.0Portabilidade....................................................................8

4.1Requisitos de interface de usuário...................................8

4.2Criterio de aceitação........................................................9

Requisitos de interface de usuário ..................................... 14

Critérios de aceitação......................................................... 15

Considerações finais...........................................................16  
5.1 Riscos........................................................................... 16  
5.2 Anexos.......................................................................... 17

5.3 Referências.................................................................... 21

# ***Documentação de Requisitos do Sistema***

***1.Introdução***

O Mizu Oriente é um restaurante especializado em culinária oriental, com foco em temakis e pratos típicos japoneses. Localizado na Rua José Hipólito Cardoso, nº 17, atende clientes presencialmente e via delivery, operando das 11h às 15h e das 18h às 23h. Para aprimorar a experiência do cliente e otimizar processos internos, o restaurante busca desenvolver um Sistema de Gerenciamento de Pedidos Online (SGP). Este documento descreve os requisitos necessários para o desenvolvimento do sistema

**English**

**Mizu Oriente** is a restaurant specializing in Oriental cuisine, with a focus on temaki and traditional Japanese dishes. Located at Rua José Hipólito Cardoso, nº 17, it serves both dine-in and delivery customers, operating from 11:00 AM to 3:00 PM and 6:00 PM to 11:00 PM. To enhance the customer experience and streamline internal processes, the restaurant aims to develop an **Online Order Management System (OOMS)**. This document outlines the requirements necessary for the system's development

**1.1 Propósito**

Este documento define os requisitos funcionais e não funcionais do SGP, que permitirá aos usuários realizar pedidos online, acompanhar entregas em tempo real e gerenciar contas. A administração do restaurante poderá atualizar o cardápio, monitorar pedidos e ver o histórico de pedidos do cliente.

**1.2 Escopo**

O SGP será acessível via navegador web, com funcionalidades para:

* Visualização e gestão do cardápio;
* Realização de pedidos
* Acompanhamento de status de entrega;
* Administração de usuários e produtos.

**1.3 Definições, Acrônimos e Abreviações**

* SGP: Sistema de Gerenciamento de Pedidos.
* Admin: Administrador responsável pela gestão do sistema.
* Usuário: Cliente ou funcionário com acesso ao sistema.

***2.Descrição Geral***

**2.1 Visão Geral do Sistema**

O SGP permitirá que clientes naveguem pelo cardápio, adicionem itens ao carrinho, realizem pedidos e acompanhem entregas. Administradores terão acesso a funcionalidades avançadas, como gestão de produtos e relatórios de vendas.

**2.2 Funções do Sistema**

**.** Navegações e pesquisar os pedidos.

**.** Criação e gerenciamento de pedidos.

**.** Acompanhamento de status de entrega.

**.** Gestão de contas de usuário.

**.** Administração do cardápio e do sistema.

**2.3 Restrições**

**.** O sistema deve estar deve estar disponível 24/7, exceto durante a manutenção do sistema.

**.** A interface deve ser responsiva e funcionar em dispositivos móveis e desktops.

**.** O sistema deve garantir a segurança dos dados dos usuários.

***3.Requisitos Específicos***

**3.1 Requisitos Funcionais**

**3.1.1 Gestão de Usuários**

**. RF001:** O sistema permitirá que novos usuários se registrem com seu nome, sobrenome, senha, e-mail e telefone.

**. RF002:** O sistema permitirá que os usuários façam login usando suas credenciais.

**. RF003:** O administrador pode excluir a conta do usuário.

**. RF004:** O sistema permitirá que os usuários recuperem suas senhas via e-mail.

**3.1.2 Cardápio do Restaurante**

**. RF005:** O sistema permitirá que os usuários visualizem os itens do cardápio (Temaki, Sashimi, e etc.).

**. RF006:** O sistema permitirá exibir detalhes dos pratos, incluindo ingredientes e preços.

**. RF007:** O administrador permitirá adicionar, editar e remover itens do cardápio.

**. RF008:** O sistema permitirá que os usuários adicionem e remova itens do carrinho.

**3.1.3 Histórico de Pedidos**

**. RF009:** Na tela de pedidos o cliente pode ver o histórico dos seus pedidos finalizados e em andamentos.

**. RF010:** No histórico de pedidos o cliente também poderá acompanhar o desenvolvimento dos seus pedidos.

**. RF011:** Na tela de pedidos o cliente poderá cancelar o seu pedido.

**3.2 Requisitos Não Funcionais**

**3.2.1 Desempenho**

**. RNF001:** O sistema ele vai gerar os pedidos em até 3 segundos após a confirmação do pedido.

**. RNF002:** O sistema deve suportar as operações em horário de pico sem falhas.

**3.2.2. Segurança**

**. RNF003:** O usuário deve conseguir alterar a senha através do Email recebendo código único de alteração.

**. RNF004:** As senhas dos usuários devem ser protegidas utilizando criptografia robusto, garantindo a segurança de dados armazenados.

**. RNF005:** O sistema deve implementar o mecanismo de segurança para evitar injeções de comandos SQL.

**3.2.3 Usabilidade**

**. RNF005:** A interface do sistema deve se adaptar bem a diferentes tamanhos de aparelhos

**. RNF006:** A navegação deve ser simples, permitindo que o usuário conclua seu pedido com 5 a 8 cliques.

**3.2.4 Confiabilidade e Disponibilidade**

**. RNF007:** O sistema deve estar disponível 90% do tempo. Ficando inoperante apenas em atualizações.

**. RNF008:** Em caso de falha no sistema, os dados dos pedidos não devem ser perdidos.

**3.2.5 Escalabilidade**

**. RNF009:** O sistema deve suportar o aumento de usuários e pedidos sem comprometer o desempenho.

**3.2.6 Compatibilidade**

**. RNF010:** Deve funcionar com o navegador designado.

**. RNF011:** Deve ser compatível com Windows.

**3.2.7 Manutenibilidade**

**.** **RNF012:** O código deve ser modular e documentado para facilitar futuras atualizações ou correções.

**. RNF013:** O banco de dados deve ser estruturado de forma relacional, seguindo boas práticas de normalização.

**3.2.8 Auditabilidade**

**. RF014:** Todas as ações administrativas e pedidos realizados devem ser registrados em logs com data, hora e identificação do usuário responsável.

**3.2.9 Portabilidade**

**. RNF015:** O sistema deve ser executável em servidores com sistema operacional Windows, com suporte de MySQL e PHP.

**4. Requisitos de interface de usuário**

**. RUI001:** Tela inicial com visão geral de pedidos em andamento e histórico de pedidos.

**. RUI002:** Tela deve permitir que o usuário altere senha Email e forma de cadastro.

**. RUI003:** Usuário deve conseguir visualizar estado do pedido vendo em que etapa ele está do processo.

**5. Critério de Aceitação**

**.** O sistema deve permitir que o administrador alterar cardápio estado do pedido.

**.** A atualização de estado do pedido deve ocorrer de forma automática.

**6. Considerações Finais**

**6.1 Riscos**

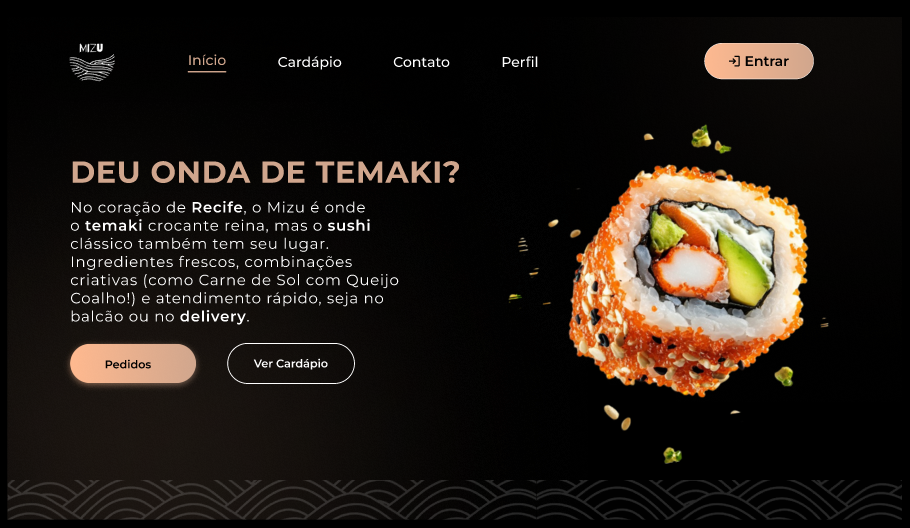
**.** Risco de resistência á mudanças dos funcionários caso o sistema seja pouco intuitivo.

**.** A integração com plataformas externas pode exigir futuras adaptações.

**Grupo**

**Integrantes**: José Nilton, Rafael Lucas, Vinicíus Tavares, Yan Lucas.

**6.1 Anexos**



Figura



Figura



Figura



Figura

Interface gráfica do usuário

O conteúdo gerado por IA pode estar incorreto.

Figura



Figura

Interface gráfica do usuário

O conteúdo gerado por IA pode estar incorreto.

Figura

Interface gráfica do usuário, Aplicativo

O conteúdo gerado por IA pode estar incorreto.

Figura

Interface gráfica do usuário, Site

O conteúdo gerado por IA pode estar incorreto.

Figura

Interface gráfica do usuário

O conteúdo gerado por IA pode estar incorreto.

Figura

**REDES**

Arquitetura e Interações de Rede do Sistema Visão Geral da Topologia

O sistema de pedidos web da temakeria adota uma topologia em estrela lógica, baseada na arquitetura cliente-servidor. Os usuários interagem com o sistema através de navegadores conectados à internet, e todas as requisições são centralizadas em um servidor web, que se comunica com um banco de dados para processar e armazenar as informações.

Interações Entre os Componentes

Cliente → Internet

O usuário acessa o sistema através de um navegador web, digitando o endereço do site. Essa ação gera uma requisição HTTP que é enviada através da internet.

Internet → Roteador / Firewall

A requisição atinge o roteador da infraestrutura de rede, que é protegida por um firewall. Este componente filtra o tráfego, permitindo apenas conexões autorizadas para garantir a segurança do sistema.

Roteador / Firewall → Servidor Web

Após passar pelo firewall, a requisição é direcionada ao servidor web, responsável por processar a lógica do sistema, autenticar usuários, gerenciar sessões e lidar com as funcionalidades do sistema de pedidos.

Servidor Web → Banco de Dados

Sempre que o sistema precisa de informações (como cardápio, dados do usuário ou status de pedidos), o servidor realiza uma consulta ao banco de dados.

Banco de Dados → Servidor Web

O banco de dados retorna os dados solicitados ao servidor, que os utiliza para compor a resposta à solicitação original.

Servidor Web → Cliente

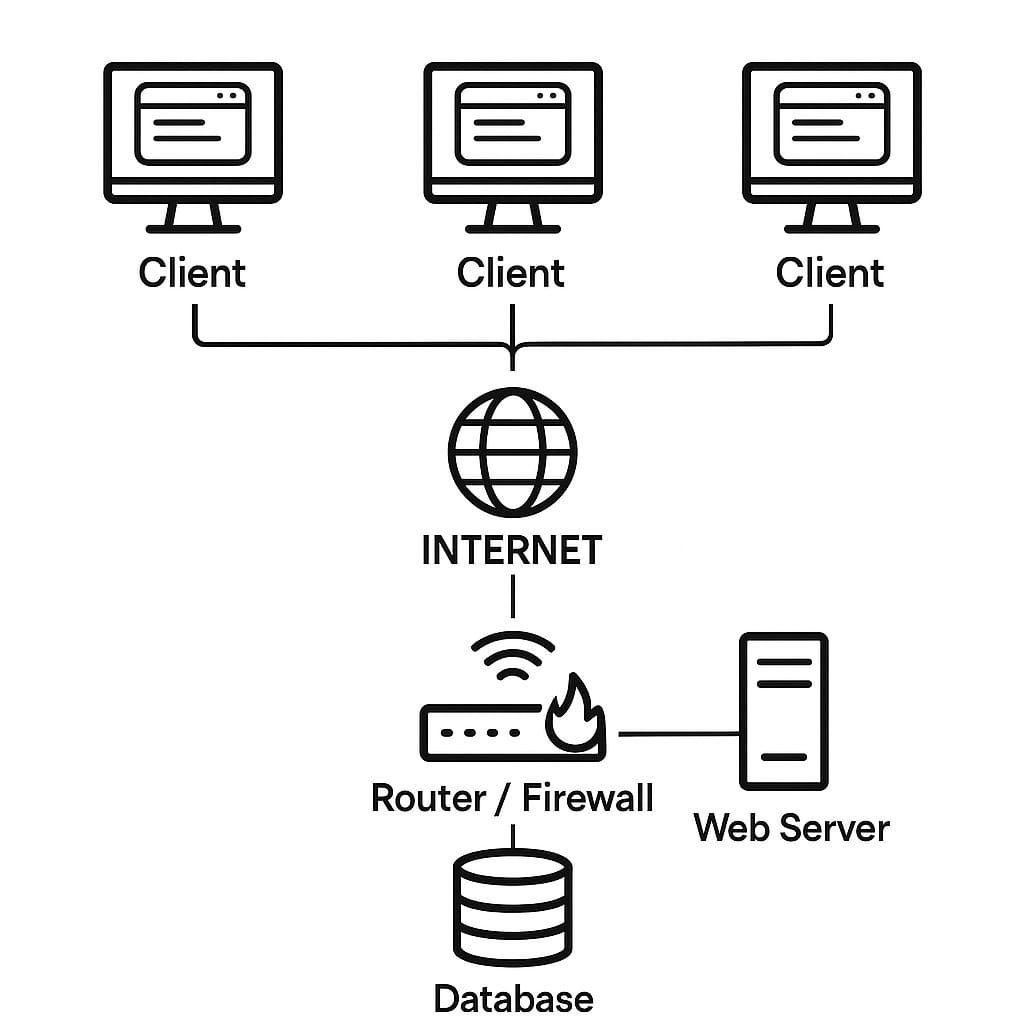
O servidor web prepara a resposta (página web renderizada, mensagens de confirmação, listas de itens, etc.) e a envia de volta ao cliente por meio da internet. O navegador então exibe essas informações ao usuário.

Fluxo Resumido da Comunicação

Cliente → Internet → Firewall → Servidor Web → Banco de Dados

Banco de Dados → Servidor Web → Internet → Cliente

**Como é mostrado na figura 11**



Figura

**Diagrama UML**

Diagrama de Classes

O Diagrama de Classes a seguir representa a estrutura do Sistema de Gerenciamento de Pedidos (SGP) do restaurante Mizu. Ele foi construído com base nas funcionalidades levantadas junto ao cliente, no formulário de requisitos e nos objetivos definidos na documentação oficial.

As principais entidades modeladas são:

•Usuário: Derivado da classe Pessoa, representa os diferentes perfis de usuários no sistema, com atributos distintos.

\* Cardápio: Armazena os produtos oferecidos pelo restaurante, com dados como descrição, preço e categoria, conforme definido nos requisitos RF005 a RF007.

\* Pedido: Relacionado à criação e controle de pedidos, incluindo quantidade, observações e valores, integrando as funcionalidades de compra e acompanhamento.

Esse diagrama traduz visualmente a lógica de funcionamento do sistema e será referência para o desenvolvimento posterior da base de dados e da codificação.

Importância do Diagrama de Classes no Projeto

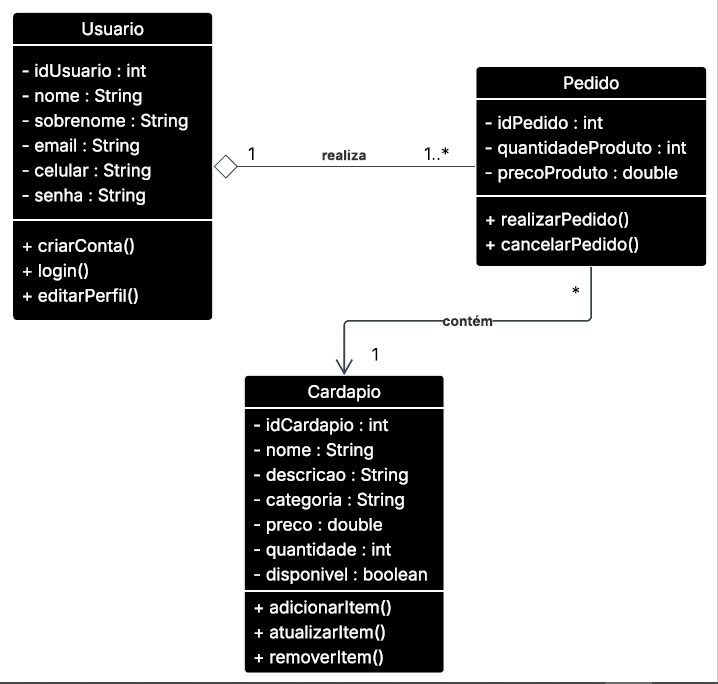
O Diagrama de Classes é um dos elementos mais relevantes da UML (Unified Modeling Language) para o desenvolvimento de sistemas orientados a objetos. Ele possibilita a representação clara das entidades do sistema, seus atributos, métodos e relacionamentos. Segundo Sommerville [1], essa representação estrutural melhora o entendimento técnico entre os membros da equipe e facilita a comunicação com os stakeholders.

Além disso, conforme destaca Pressman [2], o diagrama de classes permite mapear os requisitos do sistema diretamente em componentes reutilizáveis e bem definidos, contribuindo para a organização do código e a redução de erros durante o desenvolvimento.

Larman [3] reforça que essa modelagem promove uma abordagem iterativa e incremental, fundamental para projetos em que os requisitos ainda estão em refinamento — como no caso do restaurante Mizu, onde o sistema pode evoluir com a inclusão futura de módulos como feedback, promoções e reservas.

Assim, o Diagrama de Classes cumpre um papel essencial tanto na fase de análise quanto nas etapas de design e implementação, assegurando consistência técnica e alinhamento com os objetivos do projeto.

**Como é mostrado na figura 12**



Figura

**DER**

DER – Diagrama Entidade-Relacionamento

Conceito

O Diagrama Entidade-Relacionamento (DER) é uma ferramenta gráfica de modelagem conceitual utilizada para representar a estrutura lógica de um banco de dados. Criado por Peter Chen, o DER possibilita a identificação das entidades de um sistema, seus atributos e os relacionamentos existentes entre elas.

No Sistema de Gerenciamento de Pedidos (SGP) do restaurante Mizu Oriente, o DER é essencial para definir como os dados serão organizados, armazenados e relacionados dentro do banco de dados relacional, servindo como base sólida para a fase de implementação.

Importância para o Projeto

A utilização do DER no desenvolvimento do SGP traz diversas vantagens técnicas:

Organização dos dados: Permite visualizar a estrutura de dados do sistema de forma clara e lógica.

Segurança e integridade: Garante relacionamentos bem definidos, com integridade referencial, prevenindo erros de duplicação ou perda de dados (ELMASRI; NAVATHE, 2016).

Facilidade de manutenção: Estruturas bem modeladas favorecem futuras atualizações e escalabilidade (SILBERSCHATZ; KORTH; SUDARSHAN, 2011).

Base para bancos relacionais: A modelagem segue os princípios do modelo relacional proposto por Codd [6], que fundamenta o funcionamento de sistemas modernos de gerenciamento de banco de dados.

Aplicação no Sistema do Restaurante Mizu

Com base na análise dos requisitos funcionais do sistema, o DER do SGP contempla três entidades principais:

Usuário: Representa os clientes ou administradores que utilizam o sistema. Armazena dados como nome, telefone, e-mail e senha.

Cardápio: Contém os itens oferecidos pelo restaurante, como temakis e combinados, com informações de descrição, ingredientes e valores.

Pedido: Representa a solicitação feita pelo usuário, incluindo a data, horário, status e os produtos escolhidos.

Relacionamentos

Usuário (1:N) Pedido: Um único usuário pode realizar vários pedidos, mas cada pedido está vinculado exclusivamente a um usuário.

Cardápio (1:N) Pedido: Um item do cardápio pode aparecer em vários pedidos, mas cada pedido está relacionado a apenas um item do cardápio por vez.

Essa modelagem atende à lógica do sistema descrita nos requisitos, onde o cliente realiza pedidos individuais com base em um item do cardápio por vez, garantindo simplicidade no processo e integridade dos dados.

**Como é mostrado na figura 13**

**Diagrama

O conteúdo gerado por IA pode estar incorreto.**

Figura

**Referências**

**.** Especificações de usabilidade para aplicações de pedidos.

**.** Regras e políticas internas do restaurante.

**.** SOMMERVILLE, Ian. Engenharia de software. 9. ed. São Paulo: Pearson, 2011.

PRESSMAN, Roger S. Engenharia de software: uma abordagem profissional. 8. ed. Porto Alegre: McGraw-Hill, 2016.

LARMAN, Craig. Utilizando UML e padrões. 3. ed. Porto Alegre: Bookman, 2007.

**.** ELMASRI, R.; NAVATHE, S. B. Sistemas de Banco de Dados. 7. ed. São Paulo: Pearson, 2016.

SILBERSCHATZ, A.; KORTH, H. F.; SUDARSHAN, S. Sistemas de Banco de Dados. 6. ed. São Paulo: McGraw-Hill, 2011. CODD, E. F. A relational model of data for large shared data banks. Communications of the ACM, v. 13, n. 6, p. 377–387, 1970.