



| UNIFACS

Projeto A3

Gerência de Mesas - Site

Documentação do Sistema

Modelos, métodos e técnicas da engenharia de software
Ciência da computação
Sistema de Informação

Integrantes

Lino Menezes de Castro
João Vitor Barreto Rios
Johan Victor Prazeres Santos
Bernardo Soledade Cunha
Átila Pereira

Unifacs - Santa Mônica
Feira de Santana
2024

Introdução

O objetivo deste sistema de gerenciamento de pedidos é fornecer uma solução eficiente e organizada para restaurantes. Gerenciar pedidos manualmente pode ser uma tarefa árdua e propensa a erros, impactando negativamente a experiência do cliente e a eficiência operacional do restaurante. Este sistema foi criado para automatizar o processo de gerenciamento de pedidos, proporcionando uma série de benefícios:

Redução de Erros: A entrada manual de pedidos pode levar a erros, como esquecer pedidos ou anotar informações incorretas. Com este sistema, os pedidos são registrados digitalmente, reduzindo a chance de erros humanos.

Aumento da Eficiência: O sistema permite que os garçons adicionem e visualizem pedidos de maneira rápida e eficiente, otimizando o tempo de atendimento e melhorando a experiência do cliente.

Organização e Acompanhamento: Com todos os pedidos registrados em um banco de dados, é fácil acompanhar o status dos pedidos e obter uma visão geral das operações do restaurante, facilitando a tomada de decisões informadas.

Facilidade de Uso: O sistema é intuitivo e fácil de usar, exigindo um treinamento mínimo para os funcionários, permitindo uma rápida adoção e utilização.

Melhoria no Atendimento ao Cliente: Com a capacidade de gerenciar pedidos de forma eficiente, os garçons podem se concentrar mais no atendimento ao cliente, melhorando a satisfação e fidelidade dos clientes.

Visão Geral

Este projeto é um site de gerenciamento de pedidos para um restaurante. O site possui três páginas principais: a página inicial, a página do cardápio fictício e a página de pedidos. A página de pedidos permite que o usuário adicione e delete pedidos, onde cada pedido contém o nome do prato, o nome do garçom que atendeu e o preço do pedido. A tabela de pedidos é conectada a um banco de dados para persistência dos dados.

Tecnologias Utilizadas

Frontend: HTML, CSS, JavaScript

Backend: Java Script

Banco de Dados: MySQL (ou outro banco de dados relacional)

Estrutura do Projeto

Página Inicial

- **Descrição:** Apresenta informações sobre o projeto.

Página do Cardápio Fictício

- **Descrição:** Apresenta um cardápio fictício do restaurante.

Página de Pedidos

- **Descrição:** Permite ao usuário adicionar novos pedidos e deletar pedidos existentes. Cada pedido inclui:
 - Nome do pedido
 - Garçom que atendeu
 - Preço do pedido

Requisitos do Sistema de Gerenciamento de Pedidos

Requisitos Funcionais

Cadastro de Pedidos

- O sistema deve permitir que os usuários adicionem novos pedidos.
- Cada pedido deve incluir:
 - Nome do pedido (pedido)
 - Nome do garçom que atendeu (nome_garçom)
 - Preço do pedido (valor)

Listagem de Pedidos

- O sistema deve exibir uma lista de todos os pedidos cadastrados.
- A lista deve ser atualizada automaticamente quando um novo pedido for adicionado ou deletado.

Pedidos Deletáveis

- O sistema deve permitir que os usuários deletem pedidos existentes.
- A deleção deve ser confirmada pelo usuário antes de ser realizada.

Página Inicial

- O sistema deve ter uma página inicial que apresenta informações sobre o projeto.

Página de Cardápio

- O sistema deve ter uma página que exibe um cardápio fictício do restaurante.

Requisitos Não Funcionais

Usabilidade

- A interface do usuário deve ser intuitiva e fácil de usar.
- O design deve ser responsivo, garantindo boa usabilidade em diferentes dispositivos e tamanhos de tela.

Performance

- O sistema deve ser capaz de processar e exibir pedidos de forma rápida e eficiente.
- A adição e deleção de pedidos devem ser realizadas sem atrasos perceptíveis.

Confiabilidade

- O sistema deve ser robusto, evitando falhas e comportamentos inesperados.
- Deve ser implementada validação de entrada de dados para garantir a integridade dos dados cadastrados.

Compatibilidade

- O sistema deve ser compatível com os principais navegadores web (Chrome, Firefox, Safari, Edge).
- O backend deve ser configurável para funcionar em diferentes ambientes de desenvolvimento e produção.

Escalabilidade

- A arquitetura do sistema deve permitir a escalabilidade para lidar com um aumento no número de pedidos e usuários.
- O banco de dados e o servidor devem ser capazes de suportar o crescimento do uso do sistema.

Benefícios do Sistema

- **Redução de Erros:** Menos erros na tomada de pedidos.
- **Eficiência:** Processo de pedidos mais rápido e organizado.
- **Acompanhamento:** Melhor gestão e controle de pedidos.
- **Atendimento ao Cliente:** Melhoria na experiência do cliente.

Requisitos do Sistema de Gerenciamento de Pedidos

Requisitos Funcionais

Cadastro de Pedidos

- O sistema deve permitir que os usuários adicionem novos pedidos.

- Cada pedido deve incluir:
 - Nome do pedido (pedido)
 - Nome do garçom que atendeu (nome_garçom)
 - Preço do pedido (valor)

Listagem de Pedidos

- O sistema deve exibir uma lista de todos os pedidos cadastrados.
- A lista deve ser atualizada automaticamente quando um novo pedido for adicionado ou deletado.

Pedidos Deletaveis

- O sistema deve permitir que os usuários deletem pedidos existentes.
- A deleção deve ser confirmada pelo usuário antes de ser realizada.

Página Inicial

- O sistema deve ter uma página inicial que apresenta informações sobre o projeto.

Página de Cardápio

- O sistema deve ter uma página que exibe um cardápio fictício do restaurante.

Requisitos Não Funcionais

Usabilidade

- A interface do usuário deve ser intuitiva e fácil de usar.
- O design deve ser responsivo, garantindo boa usabilidade em diferentes dispositivos e tamanhos de tela.

Performance

- O sistema deve ser capaz de processar e exibir pedidos de forma rápida e eficiente.
- A adição e deleção de pedidos devem ser realizadas sem atrasos perceptíveis.

Segurança

- O sistema deve proteger os dados dos pedidos contra acesso não autorizado.
- Devem ser implementadas práticas de segurança no desenvolvimento para prevenir vulnerabilidades comuns, como injeção de SQL.

Confiabilidade

- O sistema deve ser robusto, evitando falhas e comportamentos inesperados.
- Deve ser implementada validação de entrada de dados para garantir a integridade dos dados cadastrados.

Compatibilidade

- O sistema deve ser compatível com os principais navegadores web (Chrome, Firefox, Safari, Edge).
- O backend deve ser configurável para funcionar em diferentes ambientes de desenvolvimento e produção.

Escalabilidade

- A arquitetura do sistema deve permitir a escalabilidade para lidar com um aumento no número de pedidos e usuários.
- O banco de dados e o servidor devem ser capazes de suportar o crescimento do uso do sistema.