

Desafio técnico MixTI

Construção: Leonardo Gildo

Solicitante: Jorge Edson

Objetivo: Análise técnica do projeto para ver aptidão para vaga de programador Júnior.

1. Estrutura do Projeto

O projeto é uma aplicação de gerenciamento de vendas e estoque, feita em Windows Forms com persistência de dados no banco de dados SQL Server usando ADO .NET e Stored Procedures. Ele terá formulários para:

- **Cadastro de Produtos:** Adicionar novos produtos no estoque.
- **Registro de Vendas:** Registrar as vendas de produtos, reduzindo a quantidade do estoque.
- **Relatório de Vendas:** Exibir as vendas feitas, com detalhes de data, produto e quantidade vendida.
- **Relatório de Estoque:** Exibir o estoque atual de produtos disponíveis.

2. Entidades (Classes)

Defini duas entidades principais para representar os dados do negócio:

- **Produto:**
 - **Id:** Um identificador único para cada produto (gerado automaticamente).
 - **Nome:** O nome do produto.
 - **Preço:** O valor unitário do produto.
 - **Quantidade:** A quantidade disponível no estoque.
- **Venda:**
 - **Id:** Um identificador único para cada venda (gerado automaticamente).
 - **ProdutoId:** Refere-se ao produto vendido.
 - **Data:** Data da venda.
 - **QuantidadeVendida:** Quantidade do produto vendida.
 - **ValorTotal:** Preço total calculado com base na quantidade vendida e no preço do produto.

Essas classes correspondem diretamente às tabelas no banco de dados, facilitando a manipulação de dados na aplicação.

3. Banco de Dados SQL Server

Aqui, criei o banco de dados que vai armazenar as informações de produtos e vendas.

- **Tabelas:**
 - **Produto:** Armazena os produtos cadastrados, incluindo o Nome, Preço, e Quantidade de cada um.

- **Venda:** Armazena as vendas realizadas, incluindo o `ProdutoId` (referência ao produto vendido), a `Data` da venda, a `QuantidadeVendida`, e o `ValorTotal`.
- **Stored Procedures:** Utilizei Stored Procedures (SPs) para realizar operações no banco de dados de forma segura e eficiente. São elas:
 - `sp_InserirProduto`: Para inserir um novo produto no banco.
 - `sp_RegistrarVenda`: Para registrar uma venda. Ele calcula o valor total da venda com base no preço do produto e também atualiza a quantidade do produto no estoque.
 - `sp_RelatorioVendas`: Para exibir todas as vendas já realizadas.
 - `sp_RelatorioEstoque`: Para exibir todos os produtos cadastrados e suas quantidades em estoque.

As SPs garantem que o código SQL será executado no servidor de banco de dados com maior controle e segurança.

4. ADO .NET para Persistência de Dados

ADO .NET usei este conjunto de classes para conectar a aplicação ao banco de dados e executar comandos SQL.

Por exemplo, ao cadastrar um produto, o código C# vai chamar a Stored Procedure `sp_InserirProduto` passando os dados de nome, preço e quantidade do produto...

Aqui, o método `InserirProduto` conecta ao banco de dados e executa a Stored Procedure `sp_InserirProduto`, inserindo um novo produto.

- **SqlConnection:** Cria a conexão com o banco.
- **SqlCommand:** Representa a instrução SQL que será executada (no caso, a Stored Procedure).
- **Parameters:** Define os valores dos parâmetros usados pela Stored Procedure.

5. Interface Windows Forms

- **Cadastro de Produto:** Um formulário onde o usuário vai inserir o nome, o preço e a quantidade de um novo produto, e clicar em um botão "Salvar" para cadastrar o produto no banco.
- **Registro de Venda:** Um formulário onde o usuário escolhe um produto (de uma lista suspensa, por exemplo), insere a quantidade vendida, e clica em um botão "Registrar Venda". A aplicação vai calcular o valor total e atualizar o estoque no banco.
- **Relatórios:**
 - **Relatório de Vendas:** Mostra uma lista de todas as vendas feitas, com data, produto, quantidade e valor.
 - **Relatório de Estoque:** Mostra a lista de produtos e suas quantidades disponíveis no estoque.

6. Boas Práticas

- **Camadas de Separação:** A aplicação segue o conceito de separação de responsabilidades em diferentes camadas:
 - **DAL (Data Access Layer):** Camada de acesso aos dados, onde você lida com ADO .NET e consultas SQL.
 - **BLL (Business Logic Layer):** Camada de lógica de negócios, onde regras e cálculos específicos são aplicados.
 - **UI (User Interface):** A interface gráfica do Windows Forms.
- **Validação e Tratamento de Exceções:** Toda vez que o usuário submete dados, é importante validar a entrada (ex.: verificar se o nome do produto não está vazio). Além disso, é importante capturar erros inesperados com `try-catch` e informar o usuário adequadamente.

Conclusão

Essa aplicação simples cobre todo o fluxo de gerenciamento de produtos e vendas. A implementação usa ADO .NET para se comunicar com o banco SQL Server e permite que você cadastre produtos, registre vendas e visualize relatórios de vendas e estoque. O uso de boas práticas de desenvolvimento, como separação de camadas e tratamento de erros, garante que a aplicação seja organizada e fácil de manter.