

Sistema de Supermercados AED Market

Daniel Araújo, Gustavo Perdigão, Pedro Neto e Yago Simões

Sistemas de Informação – PUC MINAS CONTAGEM (PUC), Brasil

***Abstract:** This document contains all specifications related to the AED Market system, from the work concept to the results obtained through our own tests. This report aims to identify any processes during the academic project and understand the entire structure of our system, aiming to make its use and understanding more efficient.*

***Resumo:** Este documento contém todas as especificações relacionadas ao sistema AED Market, desde o conceito do trabalho até o resultado obtido por meio dos nossos próprios testes. Esse relatório tem o objetivo de identificar quaisquer processos durante a realização do projeto acadêmico e entender toda a estrutura do nosso sistema, visando tornar mais eficiente a utilização e compreensão do mesmo.*

1. Introdução:

O planejamento veio a partir do trabalho final em Algoritmos e Estruturas de Dados (AED), onde a ideia era que cada grupo deverá criar um sistema em C# para solucionar um problema definido em sala de aula, no caso, pensamos que seria melhor fazer um estoque de supermercado, onde foi criado um repositório na plataforma GitHub com todas as requisições que foram pedidas, onde cada pessoa commitou no Git com a sua determinada função implementada dentro daquela parte, como a parte de login e cadastro, e do menu.

2. Ideia do trabalho:

O sistema AED Market é uma aplicação voltada ao gerenciamento de produtos no estoque de mercados, possuindo todas as funções essenciais, como adicionar produtos, remover produtos, alterar dados de produtos cadastrados e possui sistema de login com dados criptografados, no qual cada usuário tem o seu próprio estoque para mexer com os determinados produtos;

Requisitos mínimos do sistema:

- Tela de login;
- Salvar dados em arquivos;
- Utilizar um método de ordenação;
- Utilizar Lista, Pilha ou Fila;
- Possuir estrutura de controle como Árvore ou Tabela Hash;

2.1 Escolha da Linguagem:-

A escolha do C# como linguagem principal de implementação (que foi obrigatório para esse trabalho) fundamentou-se em diversos fatores. A familiaridade da equipe com a linguagem, juntamente com robustez da plataforma .NET que ofereceu um ambiente

muito acessível para o desenvolvimento de um sistema como o AED Market.

2.2 Estrutura de dados no sistema AED Market:

O sistema incorpora de maneira inteligente estruturas de dados fundamentais, como listas, mais especificamente a lista duplamente encadeada, pois ela permite um melhor acesso aos produtos além de não ser limitada em tamanho por alguma variável. Para organizar e gerenciar os dados do estoque. As listas são utilizadas para representar coleções dinâmicas de produtos, permitindo operações eficientes de adição e remoção.

Em nosso projeto, a estrutura de navegação é meticulosamente desenhada para otimizar a experiência do usuário, focando na fácil acessibilidade e na clareza das funcionalidades disponíveis. Na página principal, priorizamos a exibição dos produtos em uma lista central, proporcionando uma visão rápida e abrangente do estoque. Para complementar esta visualização principal, as demais funcionalidades são estrategicamente posicionadas no lado direito da lista.

Essa abordagem busca maximizar a eficiência operacional, permitindo que os usuários naveguem entre produtos e, ao mesmo tempo, acessem instantaneamente ferramentas essenciais. Ao concentrar funcionalidades adicionais no lado direito, minimizamos distrações visuais, focando na lista de produtos como o ponto central da interação. A organização intencional visa proporcionar uma experiência intuitiva, onde ações como adicionar novos produtos, gerenciar estoque ou acessar relatórios são facilmente acessíveis sem prejudicar a visibilidade da lista principal de produtos. Este design visa simplificar a navegação, mantendo a eficácia e a usabilidade em primeiro plano para otimizar as operações no contexto de um supermercado.

3. Funções:

Dentro do projeto AED - Market, diversas funções foram estrategicamente implementadas para otimizar o fluxo de trabalho e atender aos requisitos do sistema de maneira eficiente. Várias dessas funções estão associadas ao cadastro e login de usuários, sendo integradas às classes que pertencem do código.

A funcionalidade de cadastro de usuários foi desenvolvida de forma muito intuitiva, permitindo que os usuários preencham informações essenciais, como nome de usuário e senha, através de interfaces gráficas com caixas de texto (textBox). Após a confirmação desses dados pelo usuário, o sistema procede à criação de arquivos na pasta do projeto, nos quais são armazenadas as informações fornecidas. Cada usuário é registrado em uma linha do arquivo correspondente, garantindo uma estrutura organizada e fácil manutenção.

De forma geral, as funções implementadas em C# no projeto AED - Market visam não apenas atender aos requisitos estabelecidos, mas também aprimorar a experiência do usuário, proporcionando eficiência, organização e segurança no gerenciamento de informações de cadastro e login.

4. Classes

No desenvolvimento do projeto AED – Market, a estrutura do código é fundamentada na utilização de classes, proporcionando uma organização eficiente e modularidade ao sistema. Cada classe é concebida com o propósito de definir propriedades, campos, eventos e métodos, representando entidades específicas que desempenham funções vitais para o funcionamento do sistema. Para ilustrar, considere a classe "Usuário". Esta classe é essencial para o gerenciamento de contas dentro do AED – Market. Nela, são definidos atributos como "nome" e "senha", elementos cruciais para autenticação no sistema. Métodos como "login" e "cadastrarUsuário" são implementados para facilitar a interação do usuário com a plataforma. Uma função adicional dessa classe é salvar, linha a linha, as informações do usuário em um arquivo, proporcionando persistência aos dados. Além disso, outras classes estão relacionadas aos produtos disponíveis no menu do AED – Market. A classe de produtos inclui propriedades como "código", "nome" e "quantidade", permitindo uma gestão eficaz do estoque. Métodos como "adicionarProduto" e "removerProduto" são desenvolvidos para possibilitar a manipulação dos produtos pelo usuário.

Usuarios

Uma classe que contém o nome de usuário a senha e um apontador para o próximo usuário da lista.

ListaUsuarios

A lista de usuários serve para controlar a lista, ela armazena um apontador para o primeiro e o ultimo usuário, quantidade de usuários salvos, e as funções de adicionar, lerUsuarios e salvarUsuarios.

A função de adicionar assim como seu nome indica adiciona um novo usuário a lista de usuários.

A função lerUsuarios carrega todos os usuários salvos no arquivo contas.txt e os adiciona na lista de usuários utilizando a função de adicionar.

A função salvarUsuarios salva todos os usuários no arquivo contas.txt para que possam ser carregados posteriormente, antes de salvar os usuários a quantidade de usuários salvos é escrita.

Produtos

A classe produtos armazena todos os dados de um produto, seu nome, preço, quantidade, código, próximo produto da lista e o produto anterior da lista.

Estoque

A classe estoque serve para interagir com os produtos cadastrados, ela adiciona, remove, localiza, lê os produtos salvos e salva os produtos cadastrados, a classe estoque armazena o primeiro produto, o último produto e quantidade de produtos armazenados.

Usuário

A classe usuária serve para realizar o processo de login, cadastro e direcionar a qual estoque o usuário tem acesso, além de armazenar dados dos usuários, como nome e senha, e garante que sejam criptografados ao serem salvos.

5. Telas produzidas e funcionamento da aplicação:

Inserir produtos

No processo de inserir um produto, o usuário deverá informar o código, o preço, o nome e a quantidade atual do mesmo.

Após isso o produto será adicionado a lista de produtos criada durante a execução do programa, após o fechamento da janela principal ou a troca para uma outra janela a lista de produtos é salva em um arquivo específico do usuário.

Remover produtos

Para remover um produto é necessário informar o seu código, esse processo é irreversível e apaga todos os dados relacionados ao produto.

Entretanto o processo assim como inserir produtos é salvo apenas quando a janela principal é fechada ou outra janela é aberta.

Alterar dados

Para editar o estoque, o usuário deve fornecer o código do produto desejado, e poderá editar todas as respectivas informações.

Assim como os processos anteriores a alteração ocorre quando a janela é fechada ou outra é aberta.

Sistema de login

O sistema de login permite o cadastro de usuário e para cada usuário fornece acesso a um "estoque" diferente.

Para o cadastro é necessário fornecer um nome de usuário e senha, que serão usados para realizar o login e o nome do arquivo de estoque corresponde ao nome de usuário com a adição de “_estoque.txt” no final, o arquivo é criado quando o usuário realiza o login pela primeira vez.

Os usuários têm seu login e senha criptografados para garantir a segurança.

Aqui estão todas as telas utilizadas no trabalho:

Tela de login:



AED Market: Entrar

Insira seus dados de login

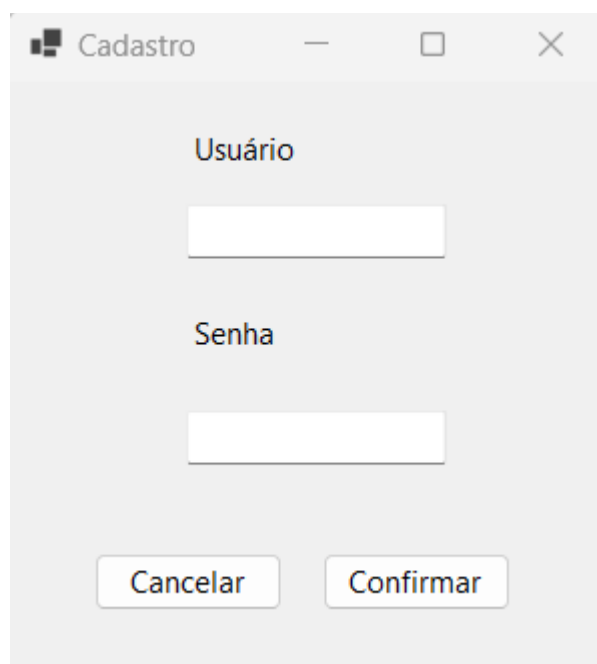
Usuário

Senha

[Cadastrar-se](#)

Sair Confirma

Tela de cadastro:



Cadastro

Usuário

Senha

Cancelar Confirmar

Tela principal:

[illegible]

Tela de excluir produto:

Os resultados obtidos no projeto AED – Market demonstram uma implementação bem-sucedida de funcionalidades, destacando a eficiência e a organização

Resultados Obtidos:

1. Cadastro e Login:

Implementação bem-sucedida de cadastro e login, utilizando arquivos e uma lista duplamente encadeada. Isso proporciona uma gestão segura e eficiente das informações dos usuários.

2. Menu de Estoque: Desenvolvimento de um menu interativo que exibe corretamente o estoque de cada usuário. Os produtos adicionados podem ser visualizados, editados ou removidos, proporcionando uma experiência completa de gerenciamento de estoque.

3. Dados: Utilização de arquivos para armazenar dados específicos de cada usuário. Essa abordagem garante a persistência das informações entre diferentes execuções do programa, contribuindo para a continuidade e consistência dos dados.

7. Pastas do Projeto:

1. Pasta ".vs":

A pasta ".vs" desempenha um papel vital no contexto do trabalho acadêmico, armazenando códigos e configurações essenciais relacionados ao projeto. Essa centralização promove consistência e integridade no código-fonte, além de fornecer um repositório único para informações cruciais.

2. Pasta "Docs": A pasta "Docs" desempenha um papel significativo na organização e preservação de documentos essenciais relacionados ao projeto acadêmico. Ao concentrar documentos neste local designado, como manuais, especificações e documentação técnica, a pasta "Docs" facilita a gestão eficiente das informações mais importantes.

8. Organização das funcionalidades

Em nosso projeto, a estrutura de navegação é meticulosamente desenhada para otimizar a experiência do usuário, focando na fácil acessibilidade e na clareza das funcionalidades disponíveis. Na página principal, priorizamos a exibição dos produtos em uma lista central, proporcionando uma visão rápida e abrangente do estoque. Para complementar esta visualização principal, as demais funcionalidades são estrategicamente posicionadas no lado direito da lista.

Essa abordagem busca maximizar a eficiência operacional, permitindo que os usuários naveguem entre produtos e, ao mesmo tempo, acessem instantaneamente ferramentas essenciais. Ao concentrar funcionalidades adicionais no lado direito, minimizamos distrações visuais, focando na lista de produtos como o ponto central da interação. A

organização intencional visa proporcionar uma experiência intuitiva, onde ações como adicionar novos produtos, gerenciar estoque ou acessar relatórios são facilmente acessíveis sem prejudicar a visibilidade da lista principal de produtos. Este design visa simplificar a navegação, mantendo a eficácia e a usabilidade em primeiro plano para otimizar as operações no contexto de um supermercado.

