



Uni Senai PR
CURSO: Engenharia de Software
DISCIPLINA: Estrutura de Dados
PROFESSORA: Mateus Ramos Pereira
Desafio prático
ALUNOS:
- Bernardo Canestraro
- Gabriel dos Santos da Luz
- Gabriel França
- Gustavo Andrés



Projeto Prático - Estrutura de dados

Introdução

Contextualização do Projeto

Com o avanço da tecnologia e a crescente demanda por sistemas informatizados na área da saúde, a informatização de clínicas médicas tornou-se essencial para otimizar atendimentos, organizar informações de pacientes e reduzir o tempo de espera. Nesse contexto, o presente projeto propõe o desenvolvimento de um sistema simples de gerenciamento de pacientes para uma clínica médica, utilizando a linguagem de programação C# e a plataforma Windows Forms, com aplicação prática de estruturas de dados como fila e lista encadeada.

Objetivo Geral

Desenvolver um sistema com interface gráfica que simule o gerenciamento de pacientes em uma clínica médica, utilizando estruturas de dados para organizar e manipular informações de forma eficiente.

Objetivos Específicos

Implementar uma estrutura de fila para gerenciar a ordem de atendimento dos pacientes;

Implementar uma lista encadeada para armazenar e organizar o histórico de pacientes atendidos;

Criar uma interface gráfica intuitiva utilizando Windows Forms;

Validar inserções de pacientes na fila;

Permitir o atendimento e remoção de pacientes da fila;

Exibir o histórico de atendimentos realizados.



Fundamentação Teórica

Estruturas de Dados Utilizadas

Fila (Queue):

Uma fila é uma estrutura de dados linear que segue o princípio FIFO (First In, First Out), ou seja, o primeiro elemento inserido é o primeiro a ser removido. Essa estrutura é ideal para representar a ordem de chegada dos pacientes.

Lista Encadeada (Linked List):

A lista encadeada é uma estrutura de dados composta por nós que armazenam dados e apontam para o próximo elemento da lista. Permite inserções e remoções dinâmicas, sendo útil para armazenar o histórico de atendimentos.

Justificativa do Uso

O uso da fila permite que os pacientes sejam atendidos na ordem correta de chegada, simulando o comportamento real de uma recepção médica. Já a lista encadeada é eficaz para registrar os atendimentos realizados, com facilidade para adicionar novos dados conforme o sistema evolui, sem desperdiçar memória.

Metodologia

Ferramentas Utilizadas

Visual Studio 2022

Linguagem C#

Plataforma Windows Forms

.NET Framework

Forma de Desenvolvimento



O projeto foi desenvolvido individualmente, permitindo ao desenvolvedor aplicar na prática os conhecimentos adquiridos em sala de aula sobre lógica de programação, orientação a objetos e estruturas de dados.

Etapas de Implementação

Definição das estruturas de dados (fila e lista encadeada);

Criação das classes base: **Paciente**, **FilaDeAtendimento**, **HistoricoAtendimento**;

Desenvolvimento da interface com Windows Forms;

Integração entre interface gráfica e lógica de dados;

Testes e validações;

Documentação do projeto.

Descrição do Projeto

Funcionamento Geral do Sistema

O sistema inicia com uma tela principal onde o usuário pode realizar ações como adicionar paciente à fila, atender paciente (remover da fila e adicionar ao histórico) e visualizar histórico de atendimentos. A interface é simples e objetiva, voltada para facilitar o uso por recepcionistas de clínicas.

Funcionalidades

Adicionar Paciente: Insere um novo paciente na fila de espera;

Atender Paciente: Remove o primeiro paciente da fila e o adiciona ao histórico;

Visualizar Histórico: Exibe uma lista com todos os pacientes já atendidos;



INOVA + (Repositório Digital)

Limpar Campos: Apaga os campos de entrada após a operação;

Mensagens de Validação: Avisos ao usuário em caso de campos vazios ou fila vazia.

Uso das Estruturas de Dados

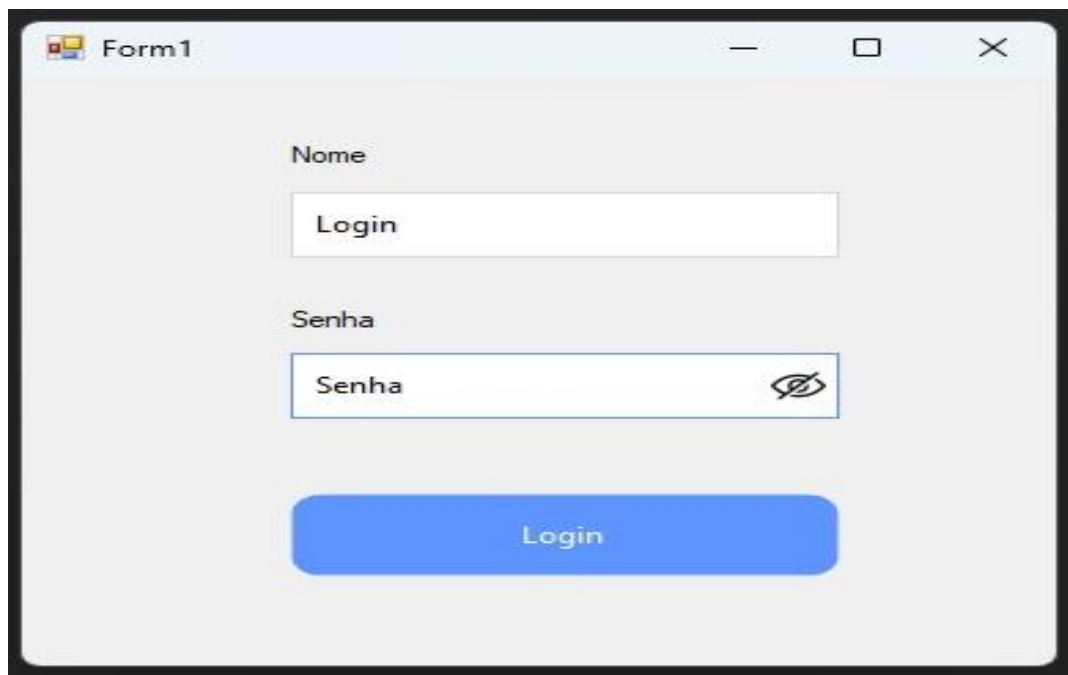
FilaDeAtendimento: Utilizada para gerenciar a ordem dos pacientes aguardando;

HistoricoAtendimento (Lista Encadeada): Usada para registrar os pacientes já atendidos;

Essas estruturas estão integradas com os botões da interface, que manipulam diretamente os objetos de cada classe.

Telas do Windows Forms

Figura 1 – Tela de login



The image shows a Windows Forms application window titled "Form1". Inside the window, there is a login form with the following elements:

- A label "Nome" above a text box containing the text "Login".
- A label "Senha" above a text box containing the text "Senha". To the right of the text box is a small icon of an eye with a diagonal line through it, typically used to toggle password visibility.
- A blue button with the text "Login" centered on it, located below the password field.



INOVA + (Repositório Digital)

Figura 2 – Tela Inicial do Sistema

The screenshot shows a web application window titled 'Form2' with a sidebar on the left containing a menu icon and two 'Formulário' (Form) buttons. The main area is titled 'Atendimento Médico' and features a table with patient data. A blue 'Atender' button is located in the top right corner of the main area.

Nome	Pressão Arterial	Temperatura	Oxigenação	Prioridade
Bruno	25	36	45	Vermelha
Pedro	10	10	10	Vermelha

Figura 3 – Cadastro de Paciente na Fila

The screenshot shows the 'Atendimento Médico' screen with a form for registering a patient. The form includes input fields for 'Nome', 'Pressão Arterial', 'Temperatura Corporal', and 'Nível de Oxigenação', each with a corresponding label. A blue 'Salvar' button is at the bottom. A success message dialog box is displayed on the right, indicating that the patient has been added to the queue.

Nome: Pedro

Pressão Arterial: 10

Temperatura Corporal: 10

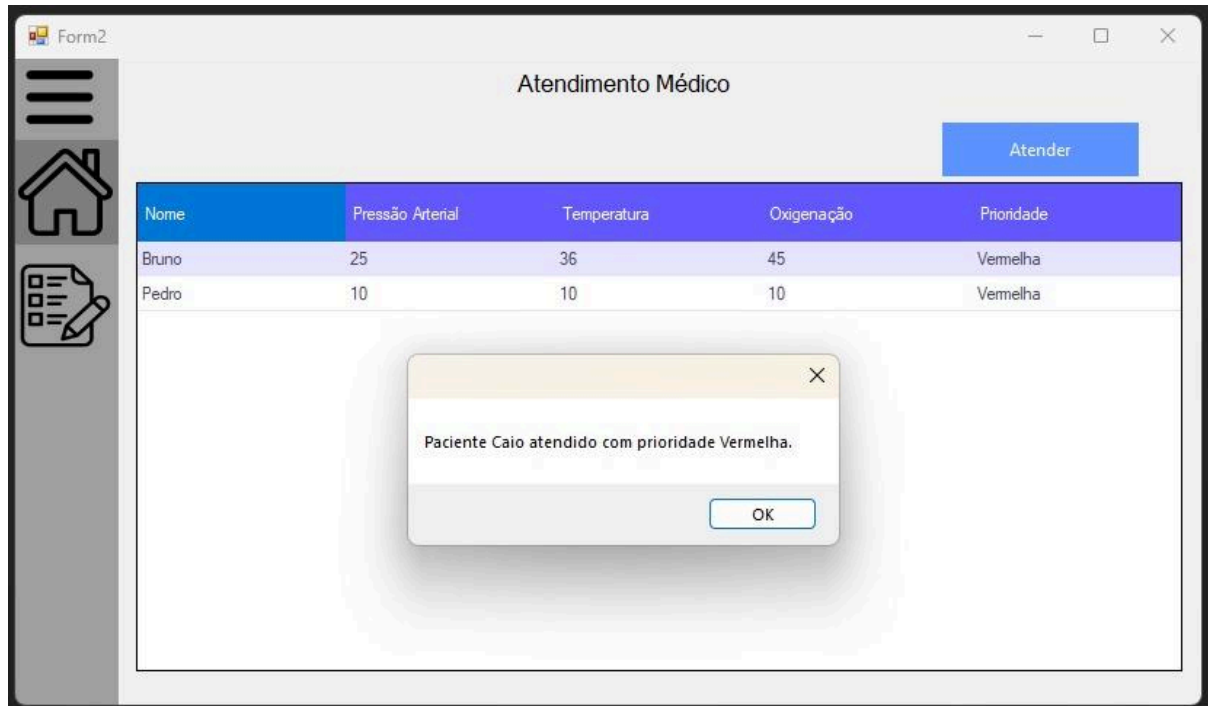
Nível de Oxigenação: 10

Salvar

Sucesso: Paciente adicionado à fila de triagem.

OK

Figura 4 – Lista de Espera e Histórico



Nome	Pressão Arterial	Temperatura	Oxigenação	Prioridade
Bruno	25	36	45	Vermelha
Pedro	10	10	10	Vermelha

Paciente Caio atendido com prioridade Vermelha.

Conclusão

O sistema proposto oferece uma solução prática e funcional para gerenciamento de atendimentos em uma clínica médica, aplicando conceitos fundamentais de estruturas de dados. A implementação com Windows Forms permitiu criar uma interface amigável, enquanto o uso de fila e lista encadeada garantiu uma organização lógica e eficiente das informações. Esse projeto reforça a importância da aplicação de teoria da computação em situações reais e contribui para a formação prática em desenvolvimento de software.