unit Projeto\_Delphi;

interface

uses

Winapi.Windows, Winapi.Messages, System.SysUtils, System.Variants, System.Classes, Vcl.Graphics,

Vcl.Controls, Vcl.Forms, Vcl.Dialogs, Vcl.ExtCtrls, Vcl.StdCtrls;

type

TForm1 = class(TForm)

Painel: TPanel;

Titulo: TLabel;

Button2: TButton;

procedure IncluirClick(Sender: TObject);

procedure IncluirPFClick(Sender: TObject);

procedure FormCreate(Sender: TObject);

procedure FormDestroy(Sender: TObject);

procedure Button2Click(Sender: TObject);

procedure IncluirDisciplinaClick(Sender: TObject);

private

listaCPFs: TStringList;

listaCodigo : TStringList;

listacodigoaluno : TStringList;

listacodigodisciplina : TStringlist;

listamateria : TStringlist;

public

end;

var

Form1: TForm1;

implementation

{$R \*.dfm}

uses Main;

procedure TForm1.Button2Click(Sender: TObject);

begin

Form2.Show;

Form1.Hide;

end;

procedure TForm1.FormCreate(Sender: TObject);

begin

//Bloco do Professor

listaCPFs := TStringList.Create;

listaCodigo := TStringList.Create;

//CPF PESSOAL DE TESTE

listaCPFs.Add('20210857757');

//CODIGO DE TESTE

listaCodigo.Add('12345');

//Bloco do Aluno

//Criação Da listas

listacodigoaluno := TStringList.Create;

//Codigo do aluno

listacodigoaluno.Add('12345');

//Bloco de disciplina

listacodigodisciplina := TStringList.Create;

listamateria := TStringList.Create;

end;

procedure TForm1.FormDestroy(Sender: TObject);

begin

listaCPFs.Free;

listaCodigo.Free;

listacodigoaluno.Free;

listacodigodisciplina.Free;

listamateria.Free;

end;

procedure TForm1.IncluirPFClick(Sender: TObject);

var

cpfDigitado, nomepf, codigoPF: string;

cpfExiste, codigoExiste: Boolean;

i: Integer;

begin

nomepf := EditNomePF.Text;

cpfDigitado := EditCPF.Text;

codigoPF := EditCodigoPF.Text;

// Verifica se o CPF tem 11 caracteres

if Length(cpfDigitado) <> 11 then

begin

ShowMessage('O CPF deve ter 11 dígitos.');

Exit;

end;

// Verifica se o código está vazio

if codigoPF = '' then

begin

ShowMessage('O código não pode estar vazio.');

Exit;

end;

// Verifica se o CPF contém apenas números

for i := 1 to Length(cpfDigitado) do

begin

if not (cpfDigitado[i] in ['0'..'9']) then

begin

ShowMessage('O CPF deve conter apenas números.');

Exit;

end;

end;

// Verifica se o CPF já está cadastrado

cpfExiste := listaCPFs.IndexOf(cpfDigitado) <> -1;

if cpfExiste then

begin

ShowMessage('Este CPF já está cadastrado.');

Exit;

end;

// Verifica se o código já está cadastrado

codigoExiste := listaCodigo.IndexOf(codigoPF) <> -1;

if codigoExiste then

begin

ShowMessage('Este código já está cadastrado no sistema.');

Exit;

end;

// Se chegou até aqui, adiciona os dados

listaCPFs.Add(cpfDigitado);

listaCodigo.Add(codigoPF);

ShowMessage('Professor cadastrado com sucesso.');

end;

procedure TForm1.IncluirDisciplinaClick(Sender: TObject);

var codDisciplina,nomeDisciplina: String;

begin

codDisciplina := EditCodigoDisciplina.Text;

nomeDisciplina := EditDisciplina.Text;

if (codDisciplina = '') or (nomeDisciplina = '') then begin

ShowMessage('Preencha Todos os campos da disciplina');

exit;

end;

//Sempre adiciona uma disciplina

listacodigodisciplina.Add(codDisciplina);

listamateria.Add(nomeDisciplina);

ShowMessage('Disciplina cadastrada com sucesso');

end;

procedure TForm1.IncluirClick(Sender: TObject);

//Declara as variaveis do botão do painel alunos

var codigoaluno:String;

codigoal:Boolean;

begin

codigoaluno :=EditCodigo.Text;

//Verifica se tem algum codigo cadastrado no sistema dos alunos

codigoal := listacodigoaluno.Indexof(codigoaluno) <> -1;

if codigoal then begin

ShowMessage('Este Codigo já está cadastrado no sistema');

end else begin

listacodigoaluno.Add(codigoaluno);

ShowMessage('Codigo cadastrado com sucesso');

end;

end;

end.