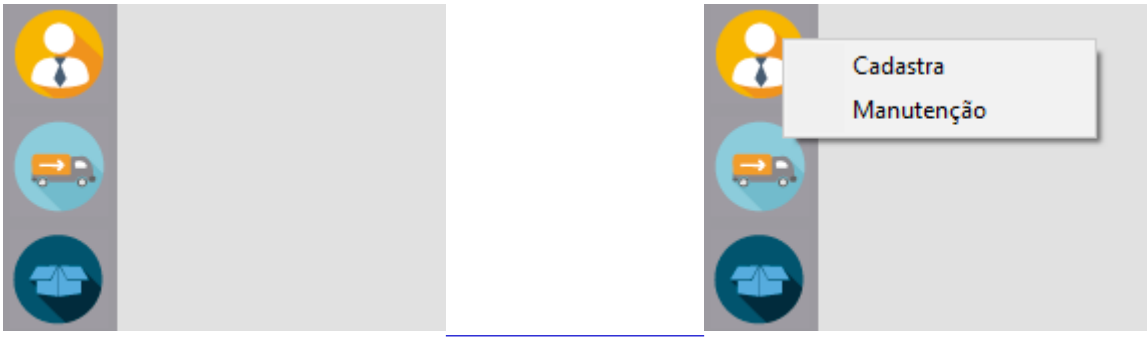
Adicionar clique direito do mouse nos botões laterais do sistema

Esse guia tem o intuito de mostrar os processos para a adição de um PopUpMenu de cadastro e manutenção nos botões laterais.



Alterações feitas na unit FormPri:

Na procedure AdicionarBotao, foi adicionado um PopupMenu que recebe 'nil' ao ser criado, para que não quebre os botões que não receberem o PopUp. Segue o exemplo do que foi feito:

```
procedure HabilitaToolBarATALHOS; virtual;
procedure DestroiBotoesDeAtalho;
procedure AdicionarBotao(AIndiceImagem: Integer; ANome: String;
  AEventoOnClick: TNotifyEvent; const AHint: String; APopupMenu: TPopupMenu = nil); virtual;
procedure CriarBotoes;virtual;
```

Na procedure AdicionarBotao, a linha que cria o botão nas laterais recebe a propriedade de um PopupMenu e foi adicionado na chamada da procedure, o parâmetro do PopUpMenu.

```
procedure TFORM_W.AdicionarBotao(AIndiceImagem: Integer; ANome: String;
  AEventoOnClick: TNotifyEvent; const AHint: String; APopupMenu: TPopupMenu);
var
  Botão: TToolButton;
begin
  Botão            := TToolButton.Create(TB_ATALHOS);
  Botão.Parent     := TB_ATALHOS;
  Botão.ImageIndex := AIndiceImagem;
  Botão.Name       := ANome;
  Botão.OnClick    := AEventoOnClick;
  Botão.Hint       := AHint;
  Botão.Width      := LARGURA_BOTAO;
  Botão.Height     := ALTURA_BOTAO;
  Botão.PopupMenu  := APopupMenu;
end;
```

Os passos até agora, não precisam ser refeitos, pois a FormPri é uma tela pai, logo, as outras telas herdam dela. Foi apresentado apenas para informação do que foi feito.

Na unit Princip:

Na procedure CriarBotoes, logo abaixo da declaração dos ícones usados nas laterais, é necessário criar uma função que vai receber o MenuItem do cadastro e da manutenção e vai retornar um PopUpMenu.

Nessa função é feita a criação dinâmica tanto do PopUp como dos itens dele para que não sejam valores fixos e sim, de acordo com o clique do mouse.

A seguir está um exemplo de como a função pode ser feita:

```
function CriarPopUpMenu(ACadastro, AManutencao: TMenuItem): TPopupMenu;
var
  LPopupMenu: TPopupMenu;
  LMenuAuxiliar: TMenuItem;
begin
  LPopupMenu := TPopupMenu.Create(nil);
  LPopupMenu.Name := 'PP_Atalho' + AManutencao.Name;
  if (ACadastro.Enabled) then
  begin
    LMenuAuxiliar := TMenuItem.Create(ACadastro);
    LMenuAuxiliar.Caption := ACadastro.Caption;
    LMenuAuxiliar.OnClick := ACadastro.OnClick;
    LPopupMenu.Items.Add(LMenuAuxiliar);
  end;

  LMenuAuxiliar := TMenuItem.Create(AManutencao);
  LMenuAuxiliar.Caption := AManutencao.Caption;
  LMenuAuxiliar.OnClick := AManutencao.OnClick;
  LPopupMenu.Items.Add(LMenuAuxiliar);
  Result := LPopupMenu;
end;
```

Ainda no Princip, devemos passar como parâmetro da procedure AdiconarBotao, a nossa função criada no passo anterior e dentro dos parâmetros da nossa função, passar o MenuItem do cadastro e da manutenção do botão específico.

A seguir, está uma demonstração de como deve ser adicionado os parâmetros na função AdicionarBotao, para que os botões, assim que criados, possam chamar o PopUp do cadastro e da

manutenção dele mesmo.

```
if Manuteno44.Enabled then
  AdicionarBotao(iIconeClientes, 'Manuteno44', Manuteno44Click, 'Clientes', CriarPopUpMenu(Cadastra44, Manuteno44));

if ManutFornec.Enabled then
  AdicionarBotao(iIconeFornecedores, 'ManutFornec', ManutFornecClick, 'Fornecedores', CriarPopUpMenu(CadFornec, ManutFornec));

if Pesquisa1.Enabled then
  AdicionarBotao(iIconeProdutos, 'Pesquisa1', Pesquisa1Click, 'Produtos', CriarPopUpMenu(Cadastra14, Pesquisa1));

if Manuteno9.Enabled then
  AdicionarBotao(iIconePlanoFiscal, 'Manuteno9', Manuteno9Click, 'Plano Fiscal', CriarPopUpMenu(Cadastra12, Manuteno9));

if ManutOrcamVD.Enabled then
  AdicionarBotao(iIconeOrcamento, 'ManutOrcamVD', ManutOrcamVDClick, 'Orçamentos', CriarPopUpMenu(CadOrcamVD, ManutOrcamVD));
```

Lembrando que essa implementação não vai quebrar as outras telas que não utilizarem a melhoria pois foi feita de maneira dinâmica, então não implementa-la é uma opção, porém por ser uma melhoria interna, é interessante utiliza-la.

Revision #14
Created 20 September 2022 17:00:43 by Felipe De Paula Vieira Da Silva
Updated 16 November 2022 14:11:49