Dim the main application form when a dialog box is displayed

Dialog windows you use to display critical information to the user are, in most cases, displayed modally. A modal form is one where the application can't continue to run until the modal (information) form is closed. Delphi's *ShowMessage*, *InputBox* and *MessageDlg*, for example, display a modal form to the user waiting for some action. Your custom dialogs are displayed using the *ShowModal* method of a form. To emphasize the importance of a modal form and the information it presents, you could gray out the main form of the application when the modal form is activated. Here's how to add a dim-out effect to your main form when modal forms are waiting for the user input.

Have a Main Form of your application Add a new form to the project. This will be the "dimmer" form. Drop the *TApplicationEvents* component on the main form. Handle the *OnModalBegin* and *OnModalEnd* events. The *TDimmerForm* is created at application startup - and since it is not the main form, it will not be displayed initially. The *Display* procedure aligns the dimmer form above the main form and gets displayed before any modal form is shown. This happens in the *TApplicationEvent*'s *OnModalBegin* event. *OnModalEnd* ensures that dimmer form is hidden until needed next time.

```
procedure TDimmerForm.FormCreate(Sender: TObject) ;
begin
  AlphaBlend := true;
  AlphaBlendValue := 128;
  BorderStyle := bsNone;
end;
```

The above is the *OnCreate* event handler for the dimmer form. By using the *AlphaBlend* property and the *AlphaBlendValue* you can make the form translucent. As it will be displayed over the main form, we want it to create the dim-out effect. *BorderStyle* ensures this form has no border, no caption bar, no title buttons. The *Display* procedure aligns the dimmer form above the main form:

```
procedure TDimmerForm.Display;
begin
  with Self do
  begin
    Left := MainForm.Left;
    Top := MainForm.Top;
    Width := MainForm.Width;
    Height := MainForm.Height;
    Show;
  end;
end;
```

Finally, handle the OnModalBegin and *OnModalEnd* events of the *TApplicationEvents* component on the main form:

```
//TApplicationEvents OnModalBegin
procedure TMainForm.ApplicationEvents1ModalBegin(Sender: TObject) ;
begin
  DimmerForm.Display;
end;

//TApplicationEvents OnModalEnd
procedure TMainForm.ApplicationEvents1ModalEnd(Sender: TObject) ;
begin
  DimmerForm.Hide;
end;
```

That's it!

In MDI (multiple document interface) Delphi applications, more than one form can be opened within a single parent window - main form. MDI application are ideal candidates for the dimmer form :)

Author:	Unknown
Contributor:	topellina