

```

//-----
#include <vcl.h>
#pragma hdrstop

#include "UnitFormMain.h"
#include "UnitUtils.h"
#include "UnitDMClients.h"
//-----
#pragma package(smart_init)
#pragma resource "*.dfm"
TformMain *formMain;
//-----
__fastcall TformMain::TformMain(TComponent* Owner)
: TForm(Owner)
{
    // Get Application Version
    strVersion = GetTSLVersion();
    Caption = Caption + " [ver." + strVersion + "]";
}
//-----
void __fastcall TformMain::FormShow(TObject *Sender)
{
    labelShowHideLogClick(Sender);

    buttonSync->SetFocus();
    buttonSync->Enabled = true;
}
//-----
void __fastcall TformMain::FormClose(TObject *Sender, TCloseAction &Action)
{
    if(dmClients->IsSyncing())
    {
        // Защита срещу затваряне на приложението при наличие на необслужени заявки.
        // За да се стигне до тази точка, обработката трябва да бъде по-продължителна (~ sec).
        Action = caNone;
        String str = "Cannot close while sync [Pending clients #" + IntToStr(dmClients->GetPending()) + "]";
        MessageDlg(str, mtError, TMsgDlgButtons() << mbOK, 0);
        return;
    }
}
//-----
void __fastcall TformMain::buttonSyncClick(TObject *Sender)
{
    // Send Batch of Requests
    dmClients->SendBatchOfReq();
}
//-----
void __fastcall TformMain::labelShowHideLogClick(TObject *Sender)
{
    if(memoLog->Visible)
    {
        // Hide Log
        labelShowHideLog->Caption = strShowHideLogCaptionShow;
        labelShowHideLog->Hint = strShowHideLogHintShow;

        memoLog->Hide();
        ClientHeight = ClientHeight - memoLog->Height;
    }
    else
    {
        // Show Log
        labelShowHideLog->Caption = strShowHideLogCaptionHide;
        labelShowHideLog->Hint = strShowHideLogHintHide;

        ClientHeight = ClientHeight + memoLog->Height;
        memoLog->Show();
    }

    Top = Screen->WorkAreaHeight - Height;
}
//-----
void __fastcall TformMain::miClearClick(TObject *Sender)
{
    int ExitCode = Application->MessageBox(L"Do you want to continue?", L"The System Log will be Cleared",
        MB_YESNO|MB_ICONWARNING);
}

```

```

    if (ExitCode == IDNO)
        return;

    memoLog->Clear();
    buttonSync->SetFocus();
}

//-----
void __fastcall TFormMain::miSaveClick(TObject *Sender)
{
    String strLogFileName = strLogFilePrefix + "." +
        FormatDateTime("yyyymmdd.hhmmss", Date() + Time()) +
        strLogFileExpension;

    try
    {
        memoLog->Lines->SaveToFile(strLogFileName);
        MessageDlg(strLogFileName + " saved", mtInformation, TMsgDlgButtons() << mbOK, 0);
    }
    catch (Exception& e)
    {
        MessageDlg(e.Message, mtError, TMsgDlgButtons() << mbOK, 0);
    }
}

//-----
void __fastcall TFormMain::miSaveTblClick(TObject *Sender)
{
    String strTblFileName = strTblFilePrefix + "." +
        FormatDateTime("yyyymmdd.hhmmss", Date() + Time()) +
        strTblFileExpension;

    try
    {
        dmClients->Save(strTblFileName);
    }
    catch (Exception& e)
    {
        MessageDlg(e.Message, mtError, TMsgDlgButtons() << mbOK, 0);
    }
}

//-----

```