

الملاحق

الملحق (أ) واجهات النظام

تدقسم واجهات النظام الى جزئين هما :

1. واجهات نظام رئيسية وهى :
 - i. واجهات خاصة بجزئية الزبون فى النظام .
 - ii. واجهات خاصة بجزئية المخدم فى النظام .
2. واجهات نظام مساعدة.

1. واجهات رئيسية للنظام

- i. واجهات خاصه بجزئية الزبون فى النظام



شاشة دخول الزبون

عند فتح الجزئية الخاصة بالزبون تظهر هذه الواجهة وعلى المستخدم إدخال إسم المستخدم وكلمة المرور، حيث تتم المقارنة مع المعلومات الموجوده فى قاعدة بيانات الجامعة ويتم الاحتفاظ برقم الجهاز وزمن الدخول.



شاشة الزبون الرئيسية

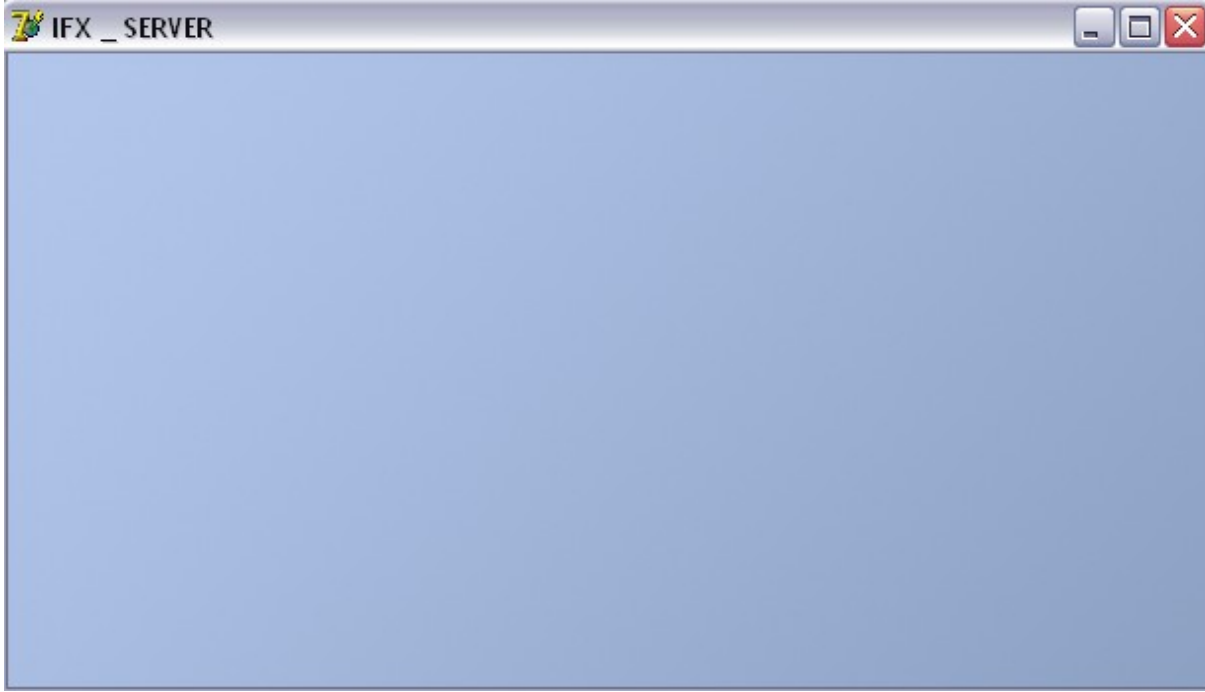
عند تشغيل جزئية الزبون الخاصه بنظام التسجيل الم قترح بنجاح تظهر هذه الشاشة في خلفية واجهة الزبون وعبرها يتم إرسال رسالتى خدمة طلب الدفعية وخدمة طلب الرصيد اليومي ويتم الرد مباشرة من قبل المخدم ، وأيضاً عبر هذه الشاشة يتم إرسال خدمة طلب الجامعة من البنك وذلك بإستخدام زمن معين (Timer) ثم يتم الرد من قبل المخدم.

ii. واجهات خاصة بجزئية المخدم فى النظام



شاشة الدخول للمخدم

عند فتح الجزئية الخاصة بالمخدم تظهر هذه الواجهة وعلى المستخدم إدخال إسم المستخدم وكلمة المرور، حيث تتم المقارنة مع المعلومات الموجوده فى قاعدة بيانات البنك ويتم الاحتفاظ برقم الجهاز وزمن الدخول.



شاشة الزبون الرئيسية

تعتبر هذه الشاشة هي الرئيسية فى جزئية المخدم ، وفى هذه الخطوة المخدم فى حالة استعداد كما يجب على المخدم فتح خدمة الويب الخاصة به (WebService) بناءً على الهيكل المحدد حسب حوجة النظام .

2. واجهات مساعده للنظام



شاشة توضيح ملخص الإرسال



شاشة لمراقبة البيانات

تظهر هاتان الشاشتان بعد دخول المستخدم لجزئية المخدم بنجاح ، من خلالها يمكن تشغيل وإيقاف الشاشة الرئيسية للمخدم وأيضاً تشغيل وإيقاف خدمة الويب، مع مراعاة تشغيل شاشة المخدم أولاً قبل تشغيل خدمة الويب.

المهمة الأساسية لهذه الجزئية توضيح ملخص الإرسال ومراقبة البيانات.

الملحق (ب)

الملفات المستخدمة في النظام

1. ملف مساعد : هو ملف يحتوي على طريقة الربط بقاعدة البيانات (Database)

ConStrDB.txt

```
Provider=SQLOLEDB.1;Integrated Security=SSPI;Persist Security  
Info=False;Initial Catalog=IFX;Data Source.=.
```

2. ملف جداول نظام التسجيل الإلكتروني المقترح : تم

إضافة ثلاثة جداول خاصة تتبع لقاعدة بيانات نظام التسجيل الإلكتروني المقترح، وهي كما يلي :

جدول رسالة خدمة طلاب الجامعة من البنك

UniSvcRq	
اسم الحقل	نوع البيانات
ID	Int
MsgName	(nvarchar(20
BankNO	Int
UniNO	Int
StdID	(nvarchar(20
CollNO	Int
Amt	Int
WriteAmt	(nvarchar(20
CurCode	(nvarchar(20
PayAmt	Int
RemAmt	Int
SentMessageDateTime	Datetime

جدول رسالة خدمة طلب الدفعية بواسطة الطالب

StPaymentRq	
اسم الحقل	نوع البيانات
ID	Int
MsgName	(nvarchar(20
BankNO	Int
UniNO	Int
StdID	(nvarchar(20
CollNO	Int
AccNo	Int
Amt	Int
WriteAmt	(nvarchar(20
CurCode	(nvarchar(20
PayAmtNow	Int
RemAmt	Int
SentMessageDateTime	Datetime

جدول رسالة خدمة طلب الرصيد اليومي

StatementBalanceRq	
اسم الحقل	نوع البيانات
ID	Int
MsgName	(nvarchar(20
BankNO	Int
UniNO	Int
AmtDay	Int
WriteAmtDay	(nvarchar(20
CurCode	(nvarchar(20
SentMessageDateTime	Datetime

3. ملف خدمة الويب (Web service) : هو ملف يحتوي على تقنية

خدمة الويب حسب الهيكل المطلوب فى رسائل الخدمات المستخدمة فى نظام التسجيل
الالكترونى المقترح، وهو كالاتى:

```
<?xml version="1.0" encoding="utf-8" ?>
<definitions xmlns="http://schemas.xmlsoap.org/wsdl/"
xmlns:xs="http://www.w3.org/2001/XMLSchema"
name="IMySOAPWebServiceservice" targetNamespace="http://tempuri.org/"
xmlns:tns="http://tempuri.org/"
```

```

xmlns:soap="http://schemas.xmlsoap.org/wsdl/soap/"
  xmlns:soapenc="http://schemas.xmlsoap.org/soap/encoding/"
xmlns:mime="http://schemas.xmlsoap.org/wsdl/mime/"
  xmlns:ns1="urn:MySOAPWebServiceIntf">
- <types>
- <xs:schema targetNamespace="urn:MySOAPWebServiceIntf"
  xmlns="urn:MySOAPWebServiceIntf">
- <xs:complexType name="AllBankSvcRq">
- <xs:sequence>
  <xs:element name="NewSignonRq" type="ns1:SignonRq" />
  <xs:element name="NewBankSvcRq" type="ns1:BankSvcRq" />
</xs:sequence>
</xs:complexType>
- <xs:complexType name="SignonRq">
- <xs:sequence>
  <xs:element name="NewSignonPswd" type="ns1:SignonPswd" />
  <xs:element name="NewCustLongPref" type="xs:string" />
</xs:sequence>
</xs:complexType>
- <xs:complexType name="SignonPswd">
- <xs:sequence>
  <xs:element name="NewCustPswd" type="ns1:CustPswd" />
  <xs:element name="NewCustId" type="ns1:CustId" />
</xs:sequence>
</xs:complexType>
- <xs:complexType name="CustPswd">
- <xs:sequence>
  <xs:element name="password" type="xs:string" />
</xs:sequence>
</xs:complexType>
- <xs:complexType name="CustId">
- <xs:sequence>
  <xs:element name="LoginId" type="xs:string" />
</xs:sequence>
</xs:complexType>
- <xs:complexType name="BankSvcRq">
- <xs:sequence>
  <xs:element name="BRqUID" type="xs:int" />
  <xs:element name="NewTmBank" type="ns1:TerminalBank" />
  <xs:element name="NewTmUniv" type="ns1:TerminalUniv" />
  <xs:element name="NewSAddRq" type="ns1:StuAddRq" />
</xs:sequence>
</xs:complexType>
- <xs:complexType name="TerminalBank">
- <xs:sequence>
  <xs:element name="BankNo" type="xs:int" />
  <xs:element name="BCity" type="xs:string" />
  <xs:element name="BCountry" type="xs:string" />
  <xs:element name="BName" type="xs:string" />
  <xs:element name="BTel" type="xs:string" />
</xs:sequence>
</xs:complexType>
- <xs:complexType name="TerminalUniv">
- <xs:sequence>
  <xs:element name="UnivNo" type="xs:int" />

```



```

<xs:element name="UCity" type="xs:string" />
<xs:element name="UCountry" type="xs:string" />
<xs:element name="UName" type="xs:string" />
<xs:element name="UTel" type="xs:string" />
</xs:sequence>
</xs:complexType>
<xs:complexType name="StuAddRq">
<xs:sequence>
<xs:element name="RqUID" type="xs:int" />
<xs:element name="NewCurAmt" type="ns1:CurAmt" />
<xs:element name="NewStuInfo" type="ns1:StuInfo" />
</xs:sequence>
</xs:complexType>
<xs:complexType name="CurAmt">
<xs:sequence>
<xs:element name="Amt" type="xs:int" />
<xs:element name="WriteAmt" type="xs:string" />
<xs:element name="CurCode" type="xs:string" />
<xs:element name="RemAmt" type="xs:int" />
<xs:element name="PayAmt" type="xs:int" />
<xs:element name="tax" type="xs:int" />
</xs:sequence>
</xs:complexType>
<xs:complexType name="StuInfo">
<xs:sequence>
<xs:element name="StNo" type="xs:int" />
<xs:element name="StName" type="xs:string" />
<xs:element name="CollNo" type="xs:int" />
<xs:element name="CollName" type="xs:string" />
<xs:element name="StStatus" type="xs:string" />
</xs:sequence>
</xs:complexType>
<xs:complexType name="BankSvcRs">
<xs:sequence>
<xs:element name="NewStatus" type="ns1:Status" />
<xs:element name="NewAllBankSvcRq" type="ns1:AllBankSvcRq" />
</xs:sequence>
</xs:complexType>
<xs:complexType name="Status">
<xs:sequence>
<xs:element name="StatusCode" type="xs:string" />
<xs:element name="SeverityId" type="xs:int" />
<xs:element name="StatusDesc" type="xs:string" />
</xs:sequence>
</xs:complexType>
<xs:complexType name="AllBankSvcRqM3">
<xs:sequence>
<xs:element name="NewSignonRq" type="ns1:SignonRq" />
<xs:element name="NewBankSvcRqM3" type="ns1:BankSvcRqM3" />
</xs:sequence>
</xs:complexType>
<xs:complexType name="BankSvcRqM3">
<xs:sequence>
<xs:element name="NewTmBank" type="ns1:TerminalBank" />
<xs:element name="NewTmUniv" type="ns1:TerminalUniv" />

```

```

<xs:element name="NewPmtAddRq" type="ns1:PmtAddRq" />
</xs:sequence>
</xs:complexType>
<xs:complexType name="PmtAddRq">
<xs:sequence>
<xs:element name="NewRqUID" type="xs:int" />
<xs:element name="NewPmtInfo" type="ns1:PmtInfo" />
</xs:sequence>
</xs:complexType>
<xs:complexType name="PmtInfo">
<xs:sequence>
<xs:element name="NewStuInfo" type="ns1:StuInfo" />
<xs:element name="NewCurAmt" type="ns1:CurAmt" />
<xs:element name="NewPayMethod" type="ns1:PayMethod" />
</xs:sequence>
</xs:complexType>
<xs:complexType name="PayMethod">
<xs:sequence>
<xs:element name="NewDepAcclIdFrom" type="ns1:DepAcclIdFrom" />
<xs:element name="NewDepAcclIdTo" type="ns1:DepAcclIdTo" />
</xs:sequence>
</xs:complexType>
<xs:complexType name="DepAcclIdFrom">
<xs:sequence>
<xs:element name="AccNo" type="xs:int" />
<xs:element name="AccTypName" type="xs:string" />
<xs:element name="BankName" type="xs:string" />
</xs:sequence>
</xs:complexType>
<xs:complexType name="DepAcclIdTo">
<xs:sequence>
<xs:element name="AccNo" type="xs:int" />
<xs:element name="AccTypName" type="xs:string" />
</xs:sequence>
</xs:complexType>
<xs:complexType name="BankSvcRsM4">
<xs:sequence>
<xs:element name="NewStatus" type="ns1:Status" />
<xs:element name="NewAllBankSvcRqM3" type="ns1:AllBankSvcRqM3" />
</xs:sequence>
</xs:complexType>
<xs:complexType name="AllBankSvcRqM5">
<xs:sequence>
<xs:element name="NewSignonRq5" type="ns1:SignonRq" />
<xs:element name="NewBankSvcRqM5" type="ns1:BankSvcRqM5" />
</xs:sequence>
</xs:complexType>
<xs:complexType name="BankSvcRqM5">
<xs:sequence>
<xs:element name="RqUID" type="xs:int" />
<xs:element name="NewAccAddRq" type="ns1:AccAddRq" />
</xs:sequence>
</xs:complexType>
<xs:complexType name="AccAddRq">
<xs:sequence>

```

```

<xs:element name="NewAcInfo" type="ns1:AcInfo" />
</xs:sequence>
</xs:complexType>
<xs:complexType name="AcInfo">
<xs:sequence>
<xs:element name="AccNo" type="xs:int" />
<xs:element name="AccTypeName" type="xs:string" />
</xs:sequence>
</xs:complexType>
<xs:complexType name="BankSvcRsM6">
<xs:sequence>
<xs:element name="NewStatus" type="ns1:Status" />
<xs:element name="NewBankSvcRqM5RS" type="ns1:BankSvcRqM5RS" />
</xs:sequence>
</xs:complexType>
<xs:complexType name="BankSvcRqM5RS">
<xs:sequence>
<xs:element name="NewAllBankSvcRqM5" type="ns1:AllBankSvcRqM5" />
<xs:element name="NewCurAmtDay" type="ns1:CurAmtDay" />
</xs:sequence>
</xs:complexType>
<xs:complexType name="CurAmtDay">
<xs:sequence>
<xs:element name="Amt" type="xs:int" />
<xs:element name="WriteAmt" type="xs:string" />
<xs:element name="CurCode" type="xs:string" />
</xs:sequence>
</xs:complexType>
</xs:schema>
</types>
<message name="MsgSvcRq0Request">
<part name="SendBankSvcRq" type="ns1:AllBankSvcRq" />
</message>
<message name="MsgSvcRq0Response">
<part name="return" type="ns1:BankSvcRs" />
</message>
<message name="MsgSvcRqM31Request">
<part name="SendBankSvcRqM3" type="ns1:AllBankSvcRqM3" />
</message>
<message name="MsgSvcRqM31Response">
<part name="return" type="ns1:BankSvcRsM4" />
</message>
<message name="MsgSvcRqM52Request">
<part name="SendBankSvcRqM5" type="ns1:AllBankSvcRqM5" />
</message>
<message name="MsgSvcRqM52Response">
<part name="return" type="ns1:BankSvcRsM6" />
</message>
<portType name="IMySOAPWebService">
<operation name="MsgSvcRq">
<input message="tns:MsgSvcRq0Request" />
<output message="tns:MsgSvcRq0Response" />
</operation>
<operation name="MsgSvcRqM3">
<input message="tns:MsgSvcRqM31Request" />

```

```

<output message="tns:MsgSvcRqM31Response" />
</operation>
- <operation name="MsgSvcRqM5">
  <input message="tns:MsgSvcRqM52Request" />
  <output message="tns:MsgSvcRqM52Response" />
</operation>
</portType>
- <binding name="IMySOAPWebServicebinding"
  type="tns:IMySOAPWebService">
  <soap:binding style="rpc" transport="http://schemas.xmlsoap.org/soap/http" />
- <operation name="MsgSvcRq">
  <soap:operation soapAction="urn:MySOAPWebServiceIntf-
    IMySOAPWebService#MsgSvcRq" style="rpc" />
- <input message="tns:MsgSvcRq0Request">
  <soap:body use="encoded"
    encodingStyle="http://schemas.xmlsoap.org/soap/encoding/"
    namespace="urn:MySOAPWebServiceIntf-IMySOAPWebService" />
</input>
- <output message="tns:MsgSvcRq0Response">
  <soap:body use="encoded"
    encodingStyle="http://schemas.xmlsoap.org/soap/encoding/"
    namespace="urn:MySOAPWebServiceIntf-IMySOAPWebService" />
</output>
</operation>
- <operation name="MsgSvcRqM3">
  <soap:operation soapAction="urn:MySOAPWebServiceIntf-
    IMySOAPWebService#MsgSvcRqM3" style="rpc" />
- <input message="tns:MsgSvcRqM31Request">
  <soap:body use="encoded"
    encodingStyle="http://schemas.xmlsoap.org/soap/encoding/"
    namespace="urn:MySOAPWebServiceIntf-IMySOAPWebService" />
</input>
- <output message="tns:MsgSvcRqM31Response">
  <soap:body use="encoded"
    encodingStyle="http://schemas.xmlsoap.org/soap/encoding/"
    namespace="urn:MySOAPWebServiceIntf-IMySOAPWebService" />
</output>
</operation>
- <operation name="MsgSvcRqM5">
  <soap:operation soapAction="urn:MySOAPWebServiceIntf-
    IMySOAPWebService#MsgSvcRqM5" style="rpc" />
- <input message="tns:MsgSvcRqM52Request">
  <soap:body use="encoded"
    encodingStyle="http://schemas.xmlsoap.org/soap/encoding/"
    namespace="urn:MySOAPWebServiceIntf-IMySOAPWebService" />
</input>
- <output message="tns:MsgSvcRqM52Response">
  <soap:body use="encoded"
    encodingStyle="http://schemas.xmlsoap.org/soap/encoding/"
    namespace="urn:MySOAPWebServiceIntf-IMySOAPWebService" />
</output>
</operation>
</binding>
- <service name="IMySOAPWebServiceservice">
- <port name="IMySOAPWebServicePort"
  binding="tns:IMySOAPWebServicebinding">

```

```
<soap:address location="http://localhost:1024/soap/IMySOAPWebService" />  
</port>  
</service>  
</definitions>
```

الملحق (ج)

ملفات شيفرة النظام

بالإضافة إلى الملفات والمكتبات المستخدمة ضمن نظام التسجيل الإلكتروني المقتراح ، تم إنجاز الآتي:

1. تعديل شيفرة واجهة خدمة الويب ((MySOAPWebServiceIntf لتتواءم مع نظام التسجيل الإلكتروني المقتراح وذلك ضمن الملف ((MySOAPServer.dpr.
2. إنشاء نموذج ((MyAppServerSoapDataModule يحتوي على كل الدوال والإجراءات التي يحتاج إليها المخدم في نظام التسجيل الإلكتروني المقتراح وذلك ضمن الملف ((MySOAPServer.dpr.
3. إنشاء نموذج ((MySOAPClientForm يحتوي على كل الدوال والإجراءات التي يحتاج إليها الزبون في نظام التسجيل الإلكتروني المقتراح وذلك ضمن الملف ((MySOAPClient.dpr.

MySOAPWebServiceIntf

```
{ Invokable interface IMySOAPWebService }
unit MySOAPWebServiceIntf;
interface
uses InvokeRegistry, Types, XSBuiltIns;
type
CustId = Class(TRemotable) // This is an actual class definition :
private
    CustLoginID : String;
published
    property LoginId: String read CustLoginID write CustLoginID;
end;
CustPswd = Class(TRemotable) // This is an actual class definition :
private
    Pswd : String;
published
    property password: String read Pswd write Pswd;
end;
SignonPswd = Class(TRemotable) // This is an actual class definition :
Private
```

```

CCustId:CustId;
CCustPswd:CustPswd;
Published
  property NewCustPswd: CustPswd read CCustPswd write CCustPswd;
  property NewCustId: CustId read CCustId write CCustId;
end;
SignonRq = Class(TRemotable) // This is an actual class definition :
private
  CSignonPswd:SignonPswd;
  ClientDt:Date;
  CustLongPref:string;
published
  property NewSignonPswd: SignonPswd read CSignonPswd write CSignonPswd;
  property NewClientDt: date read ClientDt write ClientDt;
  property NewCustLongPref: string read CustLongPref write CustLongPref;
end;
SignOffRs = Class(TRemotable) // This is an actual class definition :
private
  CCustIdRs:CustId;
published
  property NewCustIdRs: CustId read CCustIdRs write CCustIdRs;
end;
TerminalBank = Class(TRemotable) // This is an actual class definition :
private
  SBCity : String;
  SBCountry : String;
  SBName : String;
  SBTel : String;
  IBank :integer;
published
  property BankNo: integer read IBank write IBank;
  property BCity: String read SBCity write SBCity;
  property BCountry: String read SBCountry write SBCountry;
  property BName: string read SBName write SBName;
  property BTel: string read SBTel write SBTel;
end;
TerminalUniv = Class(TRemotable) // This is an actual class definition :
private
  SUCity : String;

```

```

    SUCountry : String;
    SUName : String;
    SUTel : String;
IUniv :integer;
published
    property UnivNo: integer read IUniv write IUniv;
    property UCity: String read SUCity write SUCity;
    property UCountry: String read SUCountry write SUCountry;
    property UName: string read SUName write SUName;
    property UTel: string read SUTel write SUTel;
end;
StuInfo = Class(TRemotable) // This is an actual class definition :
private
    SStName : String;
    SCollName : String;
    SStStatus : String;
    IStNo :integer;
    ICollNo :integer;
published
    property StNo: integer read IStNo write IStNo;
    property StName: String read SStName write SStName;
    property CollNo: integer read ICollNo write ICollNo;
    property CollName: string read SCollName write SCollName;
    property StStatus: string read SStStatus write SStStatus;
end;
CurAmt = Class(TRemotable) // This is an actual class definition :
private
    SWriteAmt : String;
    SCurCode : String;
    IAmt :integer;
    IRemAmt :integer;
    IPayAmt :integer;
    Itax :integer;
published
    property Amt: integer read IAmt write IAmt;
    property WriteAmt: String read SWriteAmt write SWriteAmt;
    property CurCode: String read SCurCode write SCurCode;
    property RemAmt: integer read IRemAmt write IRemAmt;
    property PayAmt: integer read IPayAmt write IPayAmt;

```



```

    property tax: integer read Itax write Itax;
end;
StuAddrRq = Class(TRemotable) // This is an actual class definition :
private
    // SAddrRq : String;
    IAddrRq :integer;
    CStuInfo:StuInfo;
    CCurAmt:CurAmt;
published
    property RqUID: integer read IAddrRq write IAddrRq;
    property NewCurAmt: CurAmt read CCurAmt write CCurAmt;
    property NewStuInfo: StuInfo read CStuInfo write CStuInfo;
end;
BankSvcRq = Class(TRemotable) // This is an actual class definition :
private
    // SAddrRq : String;
    IBSvcRq :integer;
    CTerminalBank:TerminalBank;
    CTerminalUniv:TerminalUniv;
    CStuAddrRq:StuAddrRq;
published
    property BRqUID: integer read IBSvcRq write IBSvcRq;
    property NewTmBank: TerminalBank read CTerminalBank write CTerminalBank;
    property NewTmUniv: TerminalUniv read CTerminalUniv write CTerminalUniv;
    property NewSAddrRq: StuAddrRq read CStuAddrRq write CStuAddrRq;
end;
Status = Class(TRemotable) // This is an actual class definition :
private
    SStatusCode : String;
    IStatus :integer;
    SStatusDesc : String;
published
    property StatusCode: String read SStatusCode write SStatusCode;
    property SeverityId: integer read IStatus write IStatus;
    property StatusDesc: String read SStatusDesc write SStatusDesc;
end;
AllBankSvcRq = Class(TRemotable) // This is an actual class definition :
private
    CSignonRq :SignonRq;

```

```

    CBankSvcRq:BankSvcRq;
published
    property NewSignonRq: SignonRq read CSignonRq write CSignonRq;

property NewBankSvcRq: BankSvcRq read CBankSvcRq write CBankSvcRq;
end;
BankSvcRs = Class(TRemotable) // This is an actual class definition :
private
    CStatus :Status;
    CAllBankSvcRq:AllBankSvcRq;
published
    property NewStatus: Status read CStatus write CStatus;
    property NewAllBankSvcRq: AllBankSvcRq read CAllBankSvcRq write CAllBankSvcRq;
end;
//*****Msq2
DepAccIdFrom = Class(TRemotable) // This is an actual class definition :
private
    IAcctId :integer;
    SAcctType : String;
    SBankName : String;
published
    property AccNo: integer read IAcctId write IAcctId;
    property AccTypeName: String read SAcctType write SAcctType;
    property BankName: String read SBankName write SBankName;
end;
DepAcctIdTo = Class(TRemotable) // This is an actual class definition :
private
    IAcctId :integer;
    SAcctType : String;
published
    property AccNo: integer read IAcctId write IAcctId;
    property AccTypeName: String read SAcctType write SAcctType;
end;
PayMethod = Class(TRemotable) // This is an actual class definition :
private
    DDueDt : date;
    CDepAccIdFrom :DepAccIdFrom;
    CDepAcctIdTo:DepAcctIdTo;
published

```

```

property NewDepAccIdFrom: DepAccIdFrom read CDepAccIdFrom write CDepAccIdFrom;
property NewDepAcctIdTo: DepAcctIdTo read CDepAcctIdTo write CDepAcctIdTo;
property NewDueDt: DueDt read DDueDt write DDueDt;
end;
PmtInfo = Class(TRemotable) // This is an actual class definition :
private
  CStuInfo : StuInfo;
  CCurAmt :CurAmt;
  CPayMethod:PayMethod;
published
  property NewStuInfo: StuInfo read CStuInfo write CStuInfo;
  property NewCurAmt: CurAmt read CCurAmt write CCurAmt;
  property NewPayMethod: PayMethod read CPayMethod write CPayMethod;
end;
PmtAddRq = Class(TRemotable) // This is an actual class definition :
private
  IRqUID :integer;
  CPmtInfo : PmtInfo;
published
  property NewRqUID: integer read IRqUID write IRqUID;
  property NewPmtInfo: PmtInfo read CPmtInfo write CPmtInfo;
end;
BankSvcRqM3 = Class(TRemotable) // This is an actual class definition :
private
  CTerminalBank:TerminalBank;
  CTerminalUniv:TerminalUniv;
  CPmtAddRq:PmtAddRq;
published
  property NewTmBank: TerminalBank read CTerminalBank write CTerminalBank;
  property NewTmUniv: TerminalUniv read CTerminalUniv write CTerminalUniv;
  property NewPmtAddRq: PmtAddRq read CPmtAddRq write CPmtAddRq;
end;
AllBankSvcRqM3 = Class(TRemotable) // This is an actual class definition :

private
  CSignonRq :SignonRq;
  CBankSvcRqM3:BankSvcRqM3;
published
  property NewSignonRq: SignonRq read CSignonRq write CSignonRq;

```

```

    property NewBankSvcRqM3: BankSvcRqM3 read CBankSvcRqM3 write CBankSvcRqM3;
end;

BankSvcRsM4 = Class(TRemotable) // This is an actual class definition :
Private
    CStatus :Status;
    CAllBankSvcRqM3:AllBankSvcRqM3;
published
    property NewStatus: Status read CStatus write CStatus;
    property NewAllBankSvcRqM3: AllBankSvcRqM3 read CAllBankSvcRqM3 write CAllBankSvcRqM3;
end;
//***** Msq5
AccInfo = Class(TRemotable) // This is an actual class definition :
private
    SAccType  : String;
    DAccAvailDt : Date;
    IAccNo    :integer;
published
    property AccNo: integer read IAccNo write IAccNo;
    property AccTypName: String read SAccType write SAccType;
    property AccAvailDt: String read DAccAvailDt write DAccAvailDt;
end;
AccAddRq = Class(TRemotable) // This is an actual class definition :
private
    CAccInfo  : AccInfo;
published
    property NewAccInfo: AccInfo read CAccInfo write CAccInfo;
end;
BankSvcRqM5 = Class(TRemotable) // This is an actual class definition :
private
    IRquID   : integer;
    CAccAddRq:AccAddRq;
published
    property RquID: integer read IRquID write IRquID;
    property NewAccAddRq: AccAddRq read CAccAddRq write CAccAddRq;
end;
CurAmtDay = Class(TRemotable) // This is an actual class definition :
private

```

```

IAmt : integer;
SWriteAmt: String;
SCurCode : String;
published
property Amt: integer read IAmt write IAmt;
property WriteAmt: string read SWriteAmt write SWriteAmt;
property CurCode: string read SCurCode write SCurCode;
end;
AllBankSvcRqM5 = Class(TRemotable) // This is an actual class definition :
private
CSignonRq5 :SignonRq;
CBankSvcRqM5:BankSvcRqM5;
published
property NewSignonRq5: SignonRq read CSignonRq5 write CSignonRq5;
property NewBankSvcRqM5: BankSvcRqM5 read CBankSvcRqM5 write CBankSvcRqM5;
end;
BankSvcRqM5RS = Class(TRemotable) // This is an actual class definition :
private
CAllBankSvcRqM5:AllBankSvcRqM5;
CCurAmtDay:CurAmtDay;
published
property NewAllBankSvcRqM5: AllBankSvcRqM5 read CAllBankSvcRqM5 write CAllBankSvcRqM5;
property NewCurAmtDay: CurAmtDay read CCurAmtDay write CCurAmtDay;
end;
BankSvcRsM6 = Class(TRemotable) // This is an actual class definition :
private
CStatus :Status;
CBankSvcRqM5RS:BankSvcRqM5RS;
published
property NewStatus: Status read CStatus write CStatus;
property NewBankSvcRqM5RS: BankSvcRqM5RS read CBankSvcRqM5RS write CBankSvcRqM5RS;
end;
IMySOAPWebService = interface(IInvokable)
['{4BF42BCF-294B-4D28-93D8-8311FA8B190E}']
function MsgSvcRq(SendBankSvcRq : AllBankSvcRq) : BankSvcRs; stdcall;
function MsgSvcRqM3(SendBankSvcRqM3 : AllBankSvcRqM3) : BankSvcRsM4; stdcall;
function MsgSvcRqM5(SendBankSvcRqM5 : AllBankSvcRqM5) : BankSvcRsM6; stdcall;
end;

```

implementation

```

uses MyAppServerSoapDataModuleU;
initialization
{ Invokable interfaces must be registered }
InvRegistry.RegisterInterface(TypeInfo(IMySOAPWebService));
end.

```

MyAppServerSoapDataModule

```

Unit MyAppServerSoapDataModuleU;
interface
uses SysUtils, Classes, InvokeRegistry, Midas, SOAPMidas, SOAPDm, MySOAPWebServiceIntf,
DB, ADODB;
type
IMyAppServerSoapDataModule = interface(IAppServerSOAP)
[ '{A6BAD421-FD89-436E-93DB-724E7E15A2AB}' ]
end;
TMyAppServerSoapDataModule = class(TSoapDataModule, IMyAppServerSoapDataModule,
IAppServerSOAP, IAppServer, IMySOAPWebService)
Insert_ElemMsg1: TADOStoredProc;
SP_UpdateStInfo: TADOStoredProc;
private

public
Path:string;
ConnStr:string;
function MsgSvcRq(SendBankSvcRq : AllBankSvcRq) : BankSvcRs; stdcall;
function MsgSvcRqM3(SendBankSvcRqM3 : AllBankSvcRqM3) : BankSvcRsM4; stdcall;
function MsgSvcRqM5(SendBankSvcRqM5 : AllBankSvcRqM5) : BankSvcRsM6; stdcall;
end;
implementation

{$R *.DFM}
{ TMyAppServerSoapDataModule }
function TMyAppServerSoapDataModule.MsgSvcRq(SendBankSvcRq : AllBankSvcRq) : BankSvcRs; stdcall;
var
f1: textfile;
ImpBankSvcRs :BankSvcRs;
begin
GetDir(0,Path);
Assignfile(f1,path+'\ConStrDB.txt');
Reset(f1);
readln(f1,ConnStr);
ImpBankSvcRs := BankSvcRs.Create;
ImpBankSvcRs.NewAllBankSvcRq := AllBankSvcRq.Create;
ImpBankSvcRs.NewAllBankSvcRq.NewSignonRq := SignonRq.Create;
ImpBankSvcRs.NewAllBankSvcRq.NewSignonRq.NewSignonPswd := SignonPswd.Create;
ImpBankSvcRs.NewAllBankSvcRq.NewSignonRq.NewSignonPswd.NewCustPswd:= CustPswd.Create;
ImpBankSvcRs.NewAllBankSvcRq.NewSignonRq.NewSignonPswd.NewCustId := CustId.Create;
ImpBankSvcRs.NewAllBankSvcRq.NewBankSvcRq := BankSvcRq.Create;
ImpBankSvcRs.NewAllBankSvcRq.NewBankSvcRq.NewTmBank := TerminalBank.Create;
ImpBankSvcRs.NewAllBankSvcRq.NewBankSvcRq.NewTmUniv := TerminalUniv.Create;
ImpBankSvcRs.NewAllBankSvcRq.NewBankSvcRq.NewSAddRq := StuAddRq.Create;
ImpBankSvcRs.NewAllBankSvcRq.NewBankSvcRq.NewSAddRq.NewStuInfo := StuInfo.Create;
ImpBankSvcRs.NewAllBankSvcRq.NewBankSvcRq.NewSAddRq.NewCurAmt := CurAmt.Create;
with Insert_ElemMsg1 do
begin

```

```

close;
ConnectionString:=ConnStr;
Parameters.Refresh;
Parameters.ParamByName('@BNO').Value := SendBankSvcRq.NewBankSvcRq.NewTmBank.BankNO;
Parameters.ParamByName('@BName').Value := SendBankSvcRq.NewBankSvcRq.NewTmBank.BName;
Parameters.ParamByName('@BCountry').Value :=
SendBankSvcRq.NewBankSvcRq.NewTmBank.BCountry;
Parameters.ParamByName('@BTel').Value := SendBankSvcRq.NewBankSvcRq.NewTmBank.BTel;
Parameters.ParamByName('@BCity').Value := SendBankSvcRq.NewBankSvcRq.NewTmBank.BCity;
Parameters.ParamByName('@UNO').Value := SendBankSvcRq.NewBankSvcRq.NewTmUniv.UniNO;
Parameters.ParamByName('@UName').Value := SendBankSvcRq.NewBankSvcRq.NewTmUniv.UName;
Parameters.ParamByName('@UCountry').Value :=
SendBankSvcRq.NewBankSvcRq.NewTmUniv.UCountry;
Parameters.ParamByName('@UTel').Value := SendBankSvcRq.NewBankSvcRq.NewTmUniv.UTel;
Parameters.ParamByName('@UCity').Value := SendBankSvcRq.NewBankSvcRq.NewTmUniv.UCity;
Parameters.ParamByName('@StNo').Value:=
SendBankSvcRq.NewBankSvcRq.NewSAddRq.NewStuInfo.StNO;
Parameters.ParamByName('@StName').Value :=
SendBankSvcRq.NewBankSvcRq.NewSAddRq.NewStuInfo.StName;
Parameters.ParamByName('@CollNo').Value :=
SendBankSvcRq.NewBankSvcRq.NewSAddRq.NewStuInfo.CollNo;
Parameters.ParamByName('@StStatus').Value :=
SendBankSvcRq.NewBankSvcRq.NewSAddRq.NewStuInfo.StStatus;
Parameters.ParamByName('@Amt').Value :=
SendBankSvcRq.NewBankSvcRq.NewSAddRq.NewCurAmt.Amt;
Parameters.ParamByName('@WriteAmt').Value :=
SendBankSvcRq.NewBankSvcRq.NewSAddRq.NewCurAmt.WriteAmt;
Parameters.ParamByName('@CurCodeNo').Value :=
SendBankSvcRq.NewBankSvcRq.NewSAddRq.NewCurAmt.CurCode;
Parameters.ParamByName('@RemAmt').Value :=
SendBankSvcRq.NewBankSvcRq.NewSAddRq.NewCurAmt.RemAmt;
Parameters.ParamByName('@PayAmt').Value :=
SendBankSvcRq.NewBankSvcRq.NewSAddRq.NewCurAmt.PayAmt;
Parameters.ParamByName('@Tax').Value :=
SendBankSvcRq.NewBankSvcRq.NewSAddRq.NewCurAmt.tax;
Parameters.ParamByName('@PayMethodId').Value:=
SendBankSvcRq.NewBankSvcRq.NewSAddRq.NewCurAmt.;
ExecProc;
end;
ImpBankSvcRs.NewStatus := Status.Create;
ImpBankSvcRs.NewStatus.StatusCode:= 'Error';
ImpBankSvcRs.NewStatus.SeverityId:= 1;
ImpBankSvcRs.NewStatus.StatusDesc:= ";

Result := ImpBankSvcRs;
end;
//*****//Msg3
function TMyAppServerSoapDataModule.MsgSvcRqM3(SendBankSvcRqM3 : AllBankSvcRqM3) :
BankSvcRsM4; stdcall;
var
f: textfile;
ImpBankSvcRsM4 :BankSvcRsM4;
begin
GetDir(0,Path);
Assignfile(f,path+'\ConStrDB.txt');
Reset(f);
readln(f,ConnStr);
ImpBankSvcRsM4 := BankSvcRsM4.Create;
ImpBankSvcRsM4.NewAllBankSvcRqM3 := AllBankSvcRqM3.Create;

```

```

ImpBankSvcRsM4.NewAllBankSvcRqM3.NewSignonRq           := SignonRq.Create;
ImpBankSvcRsM4.NewAllBankSvcRqM3.NewSignonRq.NewSignonPswd := SignonPswd.Create;
ImpBankSvcRsM4.NewAllBankSvcRqM3.NewSignonRq.NewSignonPswd.NewCustPswd :=
  CustPswd.Create;
ImpBankSvcRsM4.NewAllBankSvcRqM3.NewSignonRq.NewSignonPswd.NewCustId := CustId.Create;
ImpBankSvcRsM4.NewAllBankSvcRqM3.NewBankSvcRqM3         := BankSvcRqM3.Create;
ImpBankSvcRsM4.NewAllBankSvcRqM3.NewBankSvcRqM3.NewTmBank := TerminalBank.Create;
ImpBankSvcRsM4.NewAllBankSvcRqM3.NewBankSvcRqM3.NewTmUniv := TerminalUniv.Create;
ImpBankSvcRsM4.NewAllBankSvcRqM3.NewBankSvcRqM3.NewPmtAddRq := PmtAddRq.Create;
ImpBankSvcRsM4.NewAllBankSvcRqM3.NewBankSvcRqM3.NewPmtAddRq.NewPmtInfo:=PmtInfo.Create;
ImpBankSvcRsM4.NewAllBankSvcRqM3.NewBankSvcRqM3.NewPmtAddRq.NewPmtInfo.NewStuInfo
:= StuInfo.Create;

ImpBankSvcRsM4.NewAllBankSvcRqM3.NewBankSvcRqM3.NewPmtAddRq.NewPmtInfo.NewCurAmt
:= CurAmt.Create;
ImpBankSvcRsM4.NewAllBankSvcRqM3.NewBankSvcRqM3.NewPmtAddRq.NewPmtInfo.NewPayMethod
:=PayMethod.Create;
ImpBankSvcRsM4.NewAllBankSvcRqM3.NewBankSvcRqM3.NewPmtAddRq.NewPmtInfo.NewPayMethod.N
ewDepAccIdFrom := DepAccIdFrom.Create;
ImpBankSvcRsM4.NewAllBankSvcRqM3.NewBankSvcRqM3.NewPmtAddRq.NewPmtInfo.NewPayMethod.N
ewDepAcctIdTo := DepAcctIdTo.Create;
  with SP_UpdateStInfo do
    begin
      Close;
      ConnectionString:=ConnStr;
      Parameters.Refresh;
Parameters.ParamByName('@StNo').Value:=SendBankSvcRqM3.NewBankSvcRqM3.NewPmtAddRq.NewPmt
Info.NewStuInfo.StNo;
Parameters.ParamByName('@RemAmt').Value:=SendBankSvcRqM3.NewBankSvcRqM3.NewPmtAddRq.New
PmtInfo.NewCurAmt.RemAmt;
Parameters.ParamByName('@PayAmt').Value:=SendBankSvcRqM3.NewBankSvcRqM3.NewPmtAddRq.NewP
mtInfo.NewCurAmt.PayAmt;
      ExecProc;
    end;
ImpBankSvcRsM4.NewStatus                               := Status.Create;
ImpBankSvcRsM4.NewStatus.StatusCode                   := 'Error';
ImpBankSvcRsM4.NewStatus.SeverityId                   := 2;
ImpBankSvcRsM4.NewStatus.StatusDesc                   := ' ';
Result := ImpBankSvcRsM4;
end;
//***** Msg5
function TMyAppServerSoapDataModule.MsgSvcRqM5(SendBankSvcRqM5 : AllBankSvcRqM5) :
BankSvcRsM6; stdcall;
var
  ImpBankSvcRsM6 :BankSvcRsM6;
begin
  ImpBankSvcRsM6                               := BankSvcRsM6.Create;
  ImpBankSvcRsM6.NewBankSvcRqM5RS              := BankSvcRqM5RS.Create;
  ImpBankSvcRsM6.NewBankSvcRqM5RS.NewAllBankSvcRqM5:= AllBankSvcRqM5.Create;
  ImpBankSvcRsM6.NewBankSvcRqM5RS.NewAllBankSvcRqM5.NewSignonRq5:= SignonRq.Create;
  ImpBankSvcRsM6.NewBankSvcRqM5RS.NewAllBankSvcRqM5.NewSignonRq5.NewSignonPswd
:= SignonPswd.Create;
  ImpBankSvcRsM6.NewBankSvcRqM5RS.NewAllBankSvcRqM5.NewSignonRq5.NewSignonPswd.NewC
ustPswd := CustPswd.Create;
  ImpBankSvcRsM6.NewBankSvcRqM5RS.NewAllBankSvcRqM5.NewSignonRq5.NewSignonPswd.NewC
ustId := CustId.Create;
  ImpBankSvcRsM6.NewBankSvcRqM5RS.NewAllBankSvcRqM5.NewBankSvcRqM5:=
  BankSvcRqM5.Create;

```



```

ImpBankSvcRsM6.NewBankSvcRqM5RS.NewAllBankSvcRqM5.NewBankSvcRqM5.NewAccAddRq
:= AccAddRq.Create;

ImpBankSvcRsM6.NewBankSvcRqM5RS.NewAllBankSvcRqM5.NewBankSvcRqM5.NewAccAddRq.Ne
wAccInfo := AccInfo.Create;
ImpBankSvcRsM6.NewBankSvcRqM5RS.NewCurAmtDay                := CurAmtDay.Create;
ImpBankSvcRsM6.NewBankSvcRqM5RS.NewCurAmtDay.Amt           := 99999;
ImpBankSvcRsM6.NewBankSvcRqM5RS.NewCurAmtDay.WriteAmt      := ' ' ;
ImpBankSvcRsM6.NewBankSvcRqM5RS.NewCurAmtDay.CurCode       := 'Dinar';
ImpBankSvcRsM6.NewStatus                                     := Status.Create;
ImpBankSvcRsM6.NewStatus.StatusCode                        := 'Good ' ;
ImpBankSvcRsM6.NewStatus.SeverityId                        := 3;
ImpBankSvcRsM6.NewStatus.StatusDesc                        := ' ' ;
Result := ImpBankSvcRsM6;
end;

```

```

procedure TMyAppServerSoapDataModuleCreateInstance(out obj: TObject);
begin
obj := TMyAppServerSoapDataModule.Create(nil);
end;

```

initialization

```

InvRegistry.RegisterInvokableClass(TMyAppServerSoapDataModule,
TMyAppServerSoapDataModuleCreateInstance);
InvRegistry.RegisterInterface(TypeInfo(IMyAppServerSoapDataModule));
end.

```

MySOAPClientForm

```

Unit MySOAPClientFormU;
interface
uses
Windows, Messages, SysUtils, Variants, Classes, Graphics, Controls, Forms,
Dialogs, InvokeRegistry, Rio, SOAPHTTPClient,
StdCtrls, DB, ADODB, ExtCtrls;
type
TMySOAPClientForm = class(TForm)
Button1: TButton;
Label1: TLabel;
HTTPRIO1: THHTPRIO;
ADOQuery2: TADOQuery;
SP_GetIFxElement: TADOStoredProc;
SP_GetLogin: TADOStoredProc;
SP_GetStPaymentInfo: TADOStoredProc;
SP_GetAccInfo: TADOStoredProc;
Insert_BalanceDate: TADOStoredProc;
TimerMsgSvcRq: TTimer;
Image1: TImage;
Button2: TButton;
procedure Button1Click(Sender: TObject);
procedure TimerMsgSvcRqTimer(Sender: TObject);
procedure Button2Click(Sender: TObject);
private
{ Private declarations }
public
Path:string;
ConnStr:string;

```

```

{ Public declarations }
end;
var
  MySOAPClientForm: TMySOAPClientForm;
implementation
uses IMySOAPWebService1;
{$R *.dfm}

procedure TMySOAPClientForm.Button1Click(Sender: TObject);
var  URL : string;
    f1: textfile;
    NAllBankSvcRq :AllBankSvcRq;
    NBankSvcRs   : BankSvcRs;
    NAllBankSvcRqM3:AllBankSvcRqM3;
    NBankSvcRsM4  :BankSvcRsM4;
    NAllBankSvcRqM5: AllBankSvcRqM5;
    NBankSvcRsM6  : BankSvcRsM6;
begin
  GetDir(0,Path);
  Assignfile(f1,path+'\ConStrDB.txt');
  Reset(f1);
  readln(f1,ConnStr);
  /******* Msg3
  NAllBankSvcRqM3                := AllBankSvcRqM3.Create;
  NAllBankSvcRqM3.NewSignonRq    := SignonRq.Create;
  NAllBankSvcRqM3.NewSignonRq.NewSignonPswd    := SignonPswd.Create;
  NAllBankSvcRqM3.NewSignonRq.NewSignonPswd.NewCustPswd    := CustPswd.Create;
  NAllBankSvcRqM3.NewSignonRq.NewSignonPswd.NewCustId    := CustId.Create;
  NAllBankSvcRqM3.NewBankSvcRqM3    := BankSvcRqM3.Create;
  NAllBankSvcRqM3.NewBankSvcRqM3.NewTmBank    := TerminalBank.Create;
  NAllBankSvcRqM3.NewBankSvcRqM3.NewTmUniv    := TerminalUniv.Create;
  NAllBankSvcRqM3.NewBankSvcRqM3.NewPmtAddRq    := PmtAddRq.Create;
  NAllBankSvcRqM3.NewBankSvcRqM3.NewPmtAddRq.NewPmtInfo    := PmtInfo.Create;
  NAllBankSvcRqM3.NewBankSvcRqM3.NewPmtAddRq.NewPmtInfo.NewStuInfo := StuInfo.Create;
  NAllBankSvcRqM3.NewBankSvcRqM3.NewPmtAddRq.NewPmtInfo.NewCurAmt := CurAmt.Create;
  NAllBankSvcRqM3.NewBankSvcRqM3.NewPmtAddRq.NewPmtInfo.NewPayMethod :=
  PayMethod.Create;
  AllBankSvcRqM3.NewBankSvcRqM3.NewPmtAddRq.NewPmtInfo.NewPayMethod.NewDepAccIdFrom:=
  DepAccIdFrom.Create;
  NAllBankSvcRqM3.NewBankSvcRqM3.NewPmtAddRq.NewPmtInfo.NewPayMethod.NewDepAcctIdTo :=
  DepAcctIdTo.Create;
  with SP_GetLogin do
  begin
    close;
    ConnectionString:=ConnStr;
    Parameters.Refresh;
    open;
    NAllBankSvcRqM3.NewSignonRq.NewSignonPswd.NewCustId.LoginId :=
    SP_GetLogin.Fields[1].AsString;
    NAllBankSvcRqM3.NewSignonRq.NewSignonPswd.NewCustPswd.password :=
    SP_GetLogin.Fields[2].AsString;
  end;
  with SP_GetStPaymentInfo do
  begin
    close;
    ConnectionString:=ConnStr;
    Parameters.Refresh;
    Parameters.ParamByName('@StNo').Value:=1;
  open;

```

```

NAllBankSvcRqM3.NewBankSvcRqM3.NewTmBank.BankNo := SP_GetStPaymentInfo.Fields[0].AsInteger;
NAllBankSvcRqM3.NewBankSvcRqM3.NewTmBank.BName := SP_GetStPaymentInfo.Fields[1].AsString;
NAllBankSvcRqM3.NewBankSvcRqM3.NewTmBank.BCountry := SP_GetStPaymentInfo.Fields[2].AsString;
NAllBankSvcRqM3.NewBankSvcRqM3.NewTmBank.BCity := SP_GetStPaymentInfo.Fields[4].AsString;
NAllBankSvcRqM3.NewBankSvcRqM3.NewTmBank.BTel := SP_GetStPaymentInfo.Fields[3].AsString;
NAllBankSvcRqM3.NewBankSvcRqM3.NewTmUniv.UnivNo := SP_GetStPaymentInfo.Fields[5].AsInteger;
NAllBankSvcRqM3.NewBankSvcRqM3.NewTmUniv.UName := SP_GetStPaymentInfo.Fields[6].AsString;
NAllBankSvcRqM3.NewBankSvcRqM3.NewTmUniv.UCountry := SP_GetStPaymentInfo.Fields[7].AsString;
NAllBankSvcRqM3.NewBankSvcRqM3.NewTmUniv.UCity := SP_GetStPaymentInfo.Fields[9].AsString;
NAllBankSvcRqM3.NewBankSvcRqM3.NewTmUniv.UTel := SP_GetStPaymentInfo.Fields[8].AsString;
NAllBankSvcRqM3.NewBankSvcRqM3.NewPmtAddRq.NewPmtInfo.NewStuInfo.StNo :=
SP_GetStPaymentInfo.Fields[12].AsInteger;
NAllBankSvcRqM3.NewBankSvcRqM3.NewPmtAddRq.NewPmtInfo.NewStuInfo.StName :=
SP_GetStPaymentInfo.Fields[13].AsString;
NAllBankSvcRqM3.NewBankSvcRqM3.NewPmtAddRq.NewPmtInfo.NewStuInfo.CollNo :=
SP_GetStPaymentInfo.Fields[10].AsInteger;
NAllBankSvcRqM3.NewBankSvcRqM3.NewPmtAddRq.NewPmtInfo.NewStuInfo.StName :=
SP_GetStPaymentInfo.Fields[11].AsString;
NAllBankSvcRqM3.NewBankSvcRqM3.NewPmtAddRq.NewPmtInfo.NewStuInfo.StStatus :=
SP_GetStPaymentInfo.Fields[14].AsString;
NAllBankSvcRqM3.NewBankSvcRqM3.NewPmtAddRq.NewPmtInfo.NewCurAmt.Amt :=
SP_GetStPaymentInfo.Fields[15].AsInteger;
NAllBankSvcRqM3.NewBankSvcRqM3.NewPmtAddRq.NewPmtInfo.NewCurAmt.WriteAmt :=
SP_GetStPaymentInfo.Fields[16].AsString;
NAllBankSvcRqM3.NewBankSvcRqM3.NewPmtAddRq.NewPmtInfo.NewCurAmt.CurCode :=
SP_GetStPaymentInfo.Fields[23].AsString;
NAllBankSvcRqM3.NewBankSvcRqM3.NewPmtAddRq.NewPmtInfo.NewCurAmt.RemAmt :=
SP_GetStPaymentInfo.Fields[17].AsInteger;
NAllBankSvcRqM3.NewBankSvcRqM3.NewPmtAddRq.NewPmtInfo.NewCurAmt.PayAmt :=
SP_GetStPaymentInfo.Fields[18].AsInteger;
NAllBankSvcRqM3.NewBankSvcRqM3.NewPmtAddRq.NewPmtInfo.NewCurAmt.tax :=
SP_GetStPaymentInfo.Fields[19].AsInteger;
NAllBankSvcRqM3.NewBankSvcRqM3.NewPmtAddRq.NewPmtInfo.NewPayMethod.NewDepAccIdFrom.Ac
cNo := SP_GetStPaymentInfo.Fields[26].AsInteger;
NAllBankSvcRqM3.NewBankSvcRqM3.NewPmtAddRq.NewPmtInfo.NewPayMethod.NewDepAccIdFrom.Ac
cTypName := SP_GetStPaymentInfo.Fields[25].AsString;
NAllBankSvcRqM3.NewBankSvcRqM3.NewPmtAddRq.NewPmtInfo.NewPayMethod.NewDepAccIdFrom.Ba
nkName := SP_GetStPaymentInfo.Fields[1].AsString;
end;
NBankSvcRsM4 := (HTTPRIO1 as IMYSOAPWebService).MsgSvcRqM3(NAllBankSvcRqM3);
end;

procedure TMySOAPClientForm.TimerMsgSvcRqTimer(Sender: TObject);
var
    URL : string;
    f1: textfile;
    NAllBankSvcRq :AllBankSvcRq;
    NBankSvcRs : BankSvcRs;
begin
    GetDir(0,Path);
    Assignfile(f1,path+'\ConStrDB.txt');
    Reset(f1);
    readln(f1,ConnStr);
    NAllBankSvcRq :=AllBankSvcRq.Create;
    NAllBankSvcRq.NewSignonRq :=SignonRq.Create;
    NAllBankSvcRq.NewSignonRq.NewSignonPswd :=SignonPswd.Create;
    NAllBankSvcRq.NewSignonRq.NewSignonPswd.NewCustPswd :=CustPswd.Create;
    NAllBankSvcRq.NewSignonRq.NewSignonPswd.NewCustId :=CustId.Create;
    NAllBankSvcRq.NewBankSvcRq :=BankSvcRq.Create;

```

```

NAllBankSvcRq.NewBankSvcRq.NewTmBank           :=TerminalBank.Create;
NAllBankSvcRq.NewBankSvcRq.NewTmUniv          :=TerminalUniv.Create;
NAllBankSvcRq.NewBankSvcRq.NewSAddRq          :=StuAddRq.Create;
NAllBankSvcRq.NewBankSvcRq.NewSAddRq.NewStuInfo :=StuInfo.Create;
NAllBankSvcRq.NewBankSvcRq.NewSAddRq.NewCurAmt :=CurAmt.Create;
with SP_GetLogin do
begin
close;
ConnectionString:=ConnStr;
Parameters.Refresh;
open;
NAllBankSvcRq.NewSignonRq.NewSignonPswd.NewCustId.LoginId := SP_GetLogin.Fields[1].AsString;
NAllBankSvcRq.NewSignonRq.NewSignonPswd.NewCustPswd.password := SP_GetLogin.Fields[2].AsString;
end;
with SP_GetIFxElement do
begin
close;
ConnectionString:=ConnStr;
Parameters.Refresh;
Parameters.ParamByName('@StStatus').Value:='new';
open;

NAllBankSvcRq.NewBankSvcRq.NewTmBank.BName      := SP_GetIFxElement.Fields[4].AsString;
NAllBankSvcRq.NewBankSvcRq.NewTmBank.BCity      := SP_GetIFxElement.Fields[7].AsString;
NAllBankSvcRq.NewBankSvcRq.NewTmBank.BCountry   := SP_GetIFxElement.Fields[5].AsString;
NAllBankSvcRq.NewBankSvcRq.NewTmBank.BTel       := SP_GetIFxElement.Fields[6].AsString;
NAllBankSvcRq.NewBankSvcRq.NewTmUniv.UName      := SP_GetIFxElement.Fields[0].AsString;
NAllBankSvcRq.NewBankSvcRq.NewTmUniv.UCountry   := SP_GetIFxElement.Fields[1].AsString;
NAllBankSvcRq.NewBankSvcRq.NewTmUniv.UCity      := SP_GetIFxElement.Fields[46].AsString;
NAllBankSvcRq.NewBankSvcRq.NewTmUniv.UTel       := SP_GetIFxElement.Fields[2].AsString;
NAllBankSvcRq.NewBankSvcRq.NewSAddRq.NewStuInfo.StNo:= SP_GetIFxElement.Fields[8].AsInteger;
NAllBankSvcRq.NewBankSvcRq.NewSAddRq.NewStuInfo.StName := SP_GetIFxElement.Fields[9].AsString;
NAllBankSvcRq.NewBankSvcRq.NewSAddRq.NewStuInfo.CollNo :=SP_GetIFxElement.Fields[10].AsInteger;
NAllBankSvcRq.NewBankSvcRq.NewSAddRq.NewStuInfo.CollName:=SP_GetIFxElement.Fields[19].AsString;
NAllBankSvcRq.NewBankSvcRq.NewSAddRq.NewCurAmt.Amt := SP_GetIFxElement.Fields[12].AsInteger;
NAllBankSvcRq.NewBankSvcRq.NewSAddRq.NewCurAmt.WriteAmt :=
SP_GetIFxElement.Fields[13].AsString;
NAllBankSvcRq.NewBankSvcRq.NewSAddRq.NewCurAmt.CurCode :=
SP_GetIFxElement.Fields[35].AsString;
NAllBankSvcRq.NewBankSvcRq.NewSAddRq.NewCurAmt.RemAmt :=
SP_GetIFxElement.Fields[15].AsInteger;
NAllBankSvcRq.NewBankSvcRq.NewSAddRq.NewCurAmt.PayAmt:=
SP_GetIFxElement.Fields[16].AsInteger;
NAllBankSvcRq.NewBankSvcRq.NewSAddRq.NewCurAmt.tax := SP_GetIFxElement.Fields[17].AsInteger;
end;
NBankSvcRs := (HTTPRIO1 as IMYSOAPWebService).MsgSvcRq(NAllBankSvcRq);
end;

procedure TMySOAPClientForm.Button2Click(Sender: TObject);
var
f1: textfile;
NAllBankSvcRqM5: AllBankSvcRqM5;
NBankSvcRsM6 : BankSvcRsM6;
begin
GetDir(0,Path);
Assignfile(f1,path+'\ConStrDB.txt');
Reset(f1);
readln(f1,ConnStr);
//*****Msg5

```

```

NAllBankSvcRqM5                := AllBankSvcRqM5.Create;
NAllBankSvcRqM5.NewSignonRq5    := SignonRq.Create;
NAllBankSvcRqM5.NewSignonRq5.NewSignonPswd    := SignonPswd.Create;
NAllBankSvcRqM5.NewSignonRq5.NewSignonPswd.NewCustPswd    := CustPswd.Create;
NAllBankSvcRqM5.NewSignonRq5.NewSignonPswd.NewCustId    := CustId.Create;
NAllBankSvcRqM5.NewBankSvcRqM5    := BankSvcRqM5.Create;
NAllBankSvcRqM5.NewBankSvcRqM5.NewAccAddRq    := AccAddRq.Create;
NAllBankSvcRqM5.NewBankSvcRqM5.NewAccAddRq.NewAccInfo    := AccInfo.Create;
with SP_GetLogin do
  begin
  close;
  ConnectionString:=ConnStr;
  Parameters.Refresh;
  open;
NAllBankSvcRqM5.NewSignonRq5.NewSignonPswd.NewCustId.LoginId := SP_GetLogin.Fields[1].AsString;
NAllBankSvcRqM5.NewSignonRq5.NewSignonPswd.NewCustPswd.password :=
  SP_GetLogin.Fields[2].AsString;
end;
with SP_GetAccInfo do
  begin
  close;
  ConnectionString:=ConnStr;
  Parameters.Refresh;
  Parameters.ParamByName('@AccNO').Value:=1;
  open;
NAllBankSvcRqM5.NewBankSvcRqM5.NewAccAddRq.NewAccInfo.AccNo    :=
SP_GetAccInfo.Fields[0].AsInteger;
NAllBankSvcRqM5.NewBankSvcRqM5.NewAccAddRq.NewAccInfo.AccTypNm :=
SP_GetAccInfo.Fields[1].AsString;
NAllBankSvcRqM5.NewBankSvcRqM5.NewAccAddRq.NewAccInfo.AccAvailDt :=
SP_GetAccInfo.Fields[2].AsString;
end;
NBankSvcRsM6 := (HTTTPRIO1 as IMYSOAPWebService).MsgSvcRqM5(NAllBankSvcRqM5);
with Insert_BalanceDate do
  begin
  close;
  ConnectionString:=ConnStr;
  Parameters.Refresh;
  Parameters.ParamByName('@BankNo').Value    := 1;
  Parameters.ParamByName('@UniNo').Value    := 1;
Parameters.ParamByName('@AmtDay').Value:=NBankSvcRsM6.NewBankSvcRqM5RS.NewCurAmtDay.Amt
Parameters.ParamByName('@WriteAmtDay').Value :=
NBankSvcRsM6.NewBankSvcRqM5RS.NewCurAmtDay.WriteAmt;
Parameters.ParamByName('@CurCode').Value    :=
NBankSvcRsM6.NewBankSvcRqM5RS.NewCurAmtDay.CurCode;
end;
end;
end.

```