

BAB IV

IMPLEMENTASI DAN PENGUJIAN

Pada bab IV ini akan dibahas mengenai implementasi hasil dari analisis dan perancangan perangkat lunak yang telah dilakukan pada bab-bab sebelumnya pada tugas akhir ini dan juga akan dijelaskan dari hasil yang didapatkan dari pengujian perangkat lunak yang telah dibangun.

4.1 Lingkungan Implementasi

Implementasi yang dilakukan menggunakan sebuah perangkat komputer untuk membangun perangkat lunak dan uji coba perangkat lunak yang telah dibangun.

Perangkat keras yang digunakan untuk melakukan implementasi memiliki spesifikasi sebagai berikut:

1. Laptop Compaq 510
2. Processor Intel Core 2 Duo CPU T5870 @2.00 GHz
3. RAM 2 GB
4. Hard Disk 250GB

Adapun Perangkat lunak yang digunakan dalam melakukan implementasi adalah sebagai berikut:

1. Sistem Operasi Windows XP *Service Pack 3*
2. Winrar 3.90
3. Microsoft Visual Studio 2005
4. MySQL 5.0

Perangkat lunak dibangun menggunakan bahasa pemrograman VB.NET. Hasil implementasi diujikan pada Laptop Compaq 510.

4.2 Batasan Implementasi

Pada tugas akhir ini, perangkat lunak yang dibangun memiliki batasan sebagai berikut:

1. Perangkat lunak membutuhkan koneksi internet untuk proses konsolidasi data/*upload file* via FTP.
2. Perangkat lunak yang dibangun hanya dapat dijalankan pada hardware yang memiliki monitor *touch screen*.

4.3 Implementasi Kelas

Kelas-kelas yang telah dirancang diimplementasikan dengan menggunakan bahasa pemrograman *Visual Basic* (VB).

4.3.1 Deskripsi Kelas

Pada tabel 4.1 dapat dilihat daftar implementasi kelas-kelas yang ada pada perangkat lunak beserta keterangannya.

Tabel 4.1 Implementasi Kelas

Nama Kelas	Nama File	Keterangan
Symetric	Crypto.vb	
DataSync	DataSync.vb	Menangani proses sinkronisasi/ konsolidasi data
TableName	TableName.vb	Editable Class tabel master
TableNameInfo	TableNameInfo.vb	Read Only Class tabel master
TableNameInfoList	TableNameInfoList.vb	Read Only Collection Class tabel master
Journal	Journal.vb	Editable Class Transaction
JournalInfo	JournalInfo.vb	Read Only Class Transaction

JournalInfoList	JournalInfoList.vb	Read Only Collection Class Transaction
JournalItem	JournalItem	Editable Class Transaction Item
JournalInfoItem	JournalInfoItem	Read Only Class Transaction Item
JournalInfoItemList	JournalInfoItemList	Read Only Collection Class Transaction Item

4.3.2 Operasi dan Atribut

1. Kelas Symetric

Daftar hasil implementasi operasi dari kelas symetric dapat dilihat pada Tabel 4.2 dan atribut kelas pada tabel 4.3.

Tabel 4.2 Dasar Implementasi Operasi Kelas Symetric

Nama Operasi	Visibility <i>(private, public)</i>	Implementasi
Init	<i>Public</i>	<code>Public Sub Init(ByVal type As CipherType, ByVal key As String)</code>
Encrypt	<i>Public</i>	<code>Public Function Encrypt(ByVal data As Byte()) As Byte() Public Function Encrypt(ByVal data As String) As String</code>
Decrypt	<i>public</i>	<code>Public Function Decrypt(ByVal data As Byte()) As Byte() Public Function Decrypt(ByVal data As String)</code>
CallCipher	<i>public</i>	<code>Private Function CallCipher(ByVal data As Byte())</code>

Tabel 4.3 Dasar Implementasi Atribut Kelas Symetric

Nama Atribut	Visibility <i>(private, public)</i>	Implementasi
pdb	<i>private</i>	<code>Private pdb As PasswordDeriveBytes</code>
cipher	<i>private</i>	<code>Private cipher As BufferedBlockCipher</code>
key	<i>private</i>	<code>Private key As KeyParameter</code>

2. Kelas DataSync

Daftar implementasi operasi dari kelas DataSynch dapat dilihat pada tabel 4.4 dan daftar implementasi atribut pada tabel 4.5.

Tabel 4.4 Dasar Implementasi Operasi Kelas DataSync

Nama Operasi	Visibility <i>(private, public)</i>	Implementasi
BackupToXML	<i>Public shared</i>	Public Shared Function BackupToXML(ByVal keys As String, ByVal log As LoginMember, ByVal journalid As String, ByVal transDate As Date) As Boolean
GetFileContents	<i>Private shared</i>	Private Shared Function GetFileContents(ByVal fullPath As String, Optional ByRef ErrInfo As String = "") As String
SaveTextToFile	<i>Private shared</i>	Private Shared Function SaveTextToFile(ByVal strData As String, ByVal FullPath As String, Optional ByVal ErrInfo As String = "") As Boolean

Tabel 4.5 Dasar Implementasi Atribut Kelas DataSynch

Nama Atribut	Visibility <i>(private, public)</i>	Implementasi
Login	<i>private</i>	Private _login As LoginMember = Nothing
FETCH_DR	<i>Private</i>	Private Const FETCH_DR As String
FETCH_DR_ITEM	<i>Private</i>	Private Const FETCH_DR_ITEM As String
FETCH_BYDATE	<i>Private</i>	Private Const FETCH_BYDATE As String
FETCH_TRX_ITEM	<i>private</i>	Private Const FETCH_TRX_ITEM As String

4.4 Implementasi Antar Muka

Sub bab ini berisi gambar-gambar hasil implementasi antar muka dari perangkat lunak yang telah dibangun. Ada 2 jenis aplikasi yang dibangun yaitu *Front End* dan *Data Exchange*.

4.4.1 *Front End*

Aplikasi *front end* merupakan aplikasi utama yang digunakan langsung oleh *user*. Terdiri dari 2 form yaitu form login dan form transaksi. Untuk masuk ke form transaksi, *user* harus memasukkan *username* dan *password* kemudian klik button “Login”. Jika login berhasil maka tampil form transaksi.



Gambar 4.1 Form *Login*

	Nama Item	Stok Awal	Stok In	Trans In	Trans Out	Waste	Stok Sisa	Terjual
Check	<input type="checkbox"/> Es Cinciau							?
Check	<input type="checkbox"/> Sop Buah+Yogurt							?
Uncheck	<input checked="" type="checkbox"/> Teh Botol Sosro							?
Check	<input type="checkbox"/> Joytea							?
Check	<input type="checkbox"/> Tebs							?
Check	<input type="checkbox"/> Fruit Tea							?
Check	<input type="checkbox"/> Aqua 600 ml							?
Check	<input type="checkbox"/> Sprite Botol							?
Check	<input type="checkbox"/> Fanta Botol							?
Check	<input type="checkbox"/> Cocacola Botol							?
Check	<input type="checkbox"/> Isotonik							?
Check	<input type="checkbox"/> Pulpy Orange							?

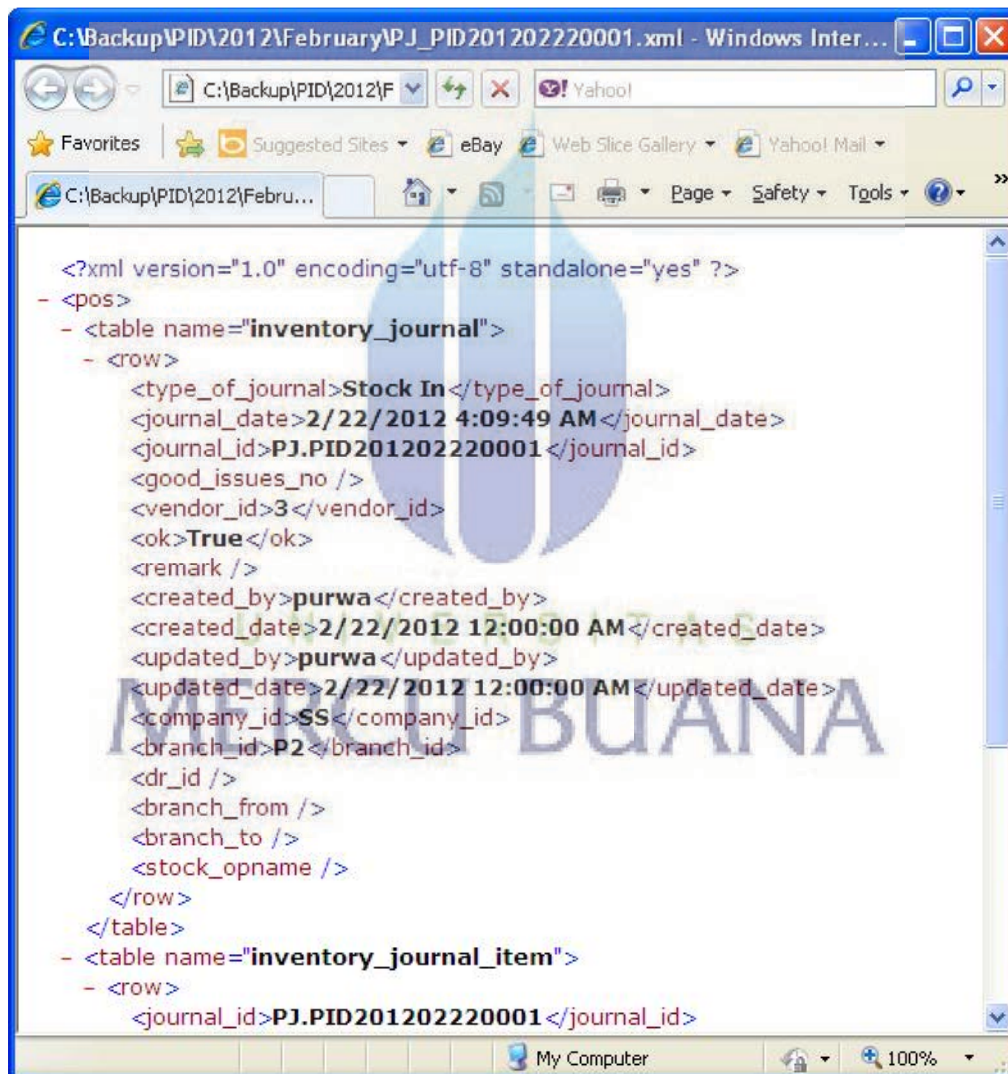
Teh Botol Sosro				
30		BOTOL		
Back Space	7	8	9	Enter
	4	6	6	
Back	1	2	3	Enter
	.	00	0	

Gambar 4.2 Form Transaksi

Pada gambar 4.2 diatas, terlihat form transaksi yang digunakan user untuk input data inventory. Untuk memasukkan jumlah item dilakukan dengan cara menekan tombol “check”. Sebelum tombol “Check” ditekan, maka item barang dalam inventory tidak aktif dan dianggap tidak ada penginputan untuk barang tersebut. Selanjutnya klik kolom yang ada tanda tanya “?” untuk memasukkan jumlah item barang masuk, transfer, waste, sisa stock maupun terjual. Kolom yang berwarna abu-abu menunjukkan bahwa jumlah item tersebut dihitung otomatis oleh program atau memang tidak memenuhi kriteria untuk diisi. Ada 5 tombol pada form transaksi, yaitu top, up, down, bottom, pending dan OK. Tombol Top digunakan untuk kembali atau pindah fokus ke baris pertama, tombol up untuk pindah fokus ke baris diatasnya, tombol down untuk pindah fokus ke baris di bawahnya dan tombol bottom untuk pindah fokus ke baris paling bawah/ terakhir. Sedangkan tombol *Pending* digunakan untuk menyimpan transaksi sementara dan tombol OK untuk menyimpan *final* transaksi. Pada saat klik tombol OK inilah *backup data* dan proses enkripsi 2 tingkat dilakukan.

Backup data berupa *text file* XML yang di enkripsi 2 kali menggunakan metode *Rivest Code 6 (RC5)* dan *Triple DES (3DES)*. Kata kunci (*key*) yang digunakan dari setiap metode tersebut berbeda. Berikut ini tahap-tahapan proses *backup data* :

1. User melakukan transaksi inventory
2. Data inventory disimpan ke dalam *local database* dan di *export* ke dalam text file berupa XML



```

<?xml version="1.0" encoding="utf-8" standalone="yes" ?>
- <pos>
- <table name="inventory_journal">
- <row>
  <type_of_journal>Stock In</type_of_journal>
  <journal_date>2/22/2012 4:09:49 AM</journal_date>
  <journal_id>PJ.PID201202220001</journal_id>
  <good_issues_no />
  <vendor_id>3</vendor_id>
  <ok>True</ok>
  <remark />
  <created_by>purwa</created_by>
  <created_date>2/22/2012 12:00:00 AM</created_date>
  <updated_by>purwa</updated_by>
  <updated_date>2/22/2012 12:00:00 AM</updated_date>
  <company_id>SS</company_id>
  <branch_id>P2</branch_id>
  <dr_id />
  <branch_from />
  <branch_to />
  <stock_opname />
</row>
</table>
- <table name="inventory_journal_item">
- <row>
  <journal_id>PJ.PID201202220001</journal_id>

```

Gambar 4.3 Plain Text

3. *Text* dalam format XML di enkripsi menggunakan algoritma RC6



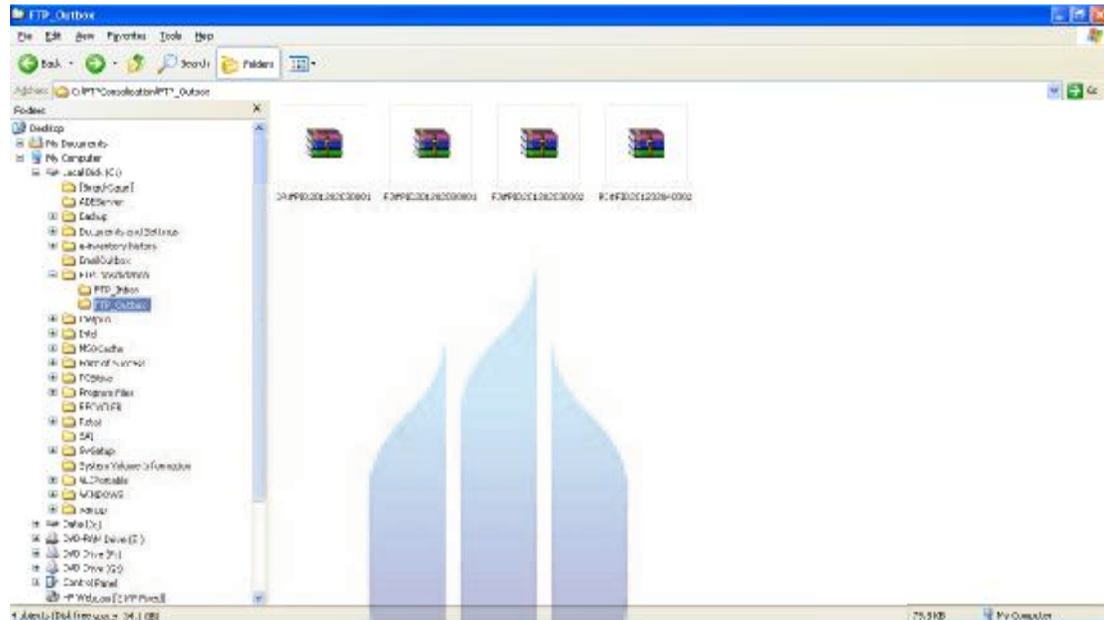
Gambar 4.4 Cipher Text RC6

4. Cipher Text dari enkripsi RC6 dienkrripsi lagi menggunakan algoritma 3DES



Gambar 4.5 Cipher Text 3DES

5. *Data Cipher Text 3DES* akan di kompres menggunakan *Winrar* untuk mengurangi ukuran *file*. *File* yang sudah di kompres selanjutnya disimpan di dalam *folder C:\FTPConsolidation\FTP_Outbox*.



Gambar 4.6 Output File

Kode Program 4.1 Backup Data

```

Friend Shared Function BackupToXML(ByVal crit As Criteria) As Boolean
    Dim result As Boolean = False
    Dim log As LoginMember = crit.Login
    Dim enc As Encoding = Nothing
    Dim fName As String = crit.JournalID.Replace(".", "_")
    Dim bln As String = GetMonth(crit.TransDate)
    Dim _strPathFile As String = ""
    _strPathFile = "C:\Backup\" & log.ObjectBranch.StrukID & "\" &
        crit.TransDate.Year & "\" & bln & "\" & fName

    If Not Directory.Exists(_strPathFile) Then
        Directory.CreateDirectory(_strPathFile)
    End If

    Dim strFile As String = _strPathFile & "\" & fName & ".xml"

    Dim dt As System.Data.DataTable = Nothing
    Dim ws As New XmlWriterSettings()
    With ws
        .CheckCharacters = True
        .CloseOutput = True
    End With

```

```

End With

Dim objXMLWriter As XmlWriter = XmlWriter.Create(strFile, ws)
Try
    If ws.ConformanceLevel <> ConformanceLevel.Fragment Then
        objXMLWriter.WriteStartDocument(True)
    End If
    Dim tblName As String = ""
    Dim tblItemName As String = ""
    Dim ds As New DataSet("Transaction")
    Dim da As MySqlDataAdapter = Nothing
    Using cn As New MySqlConnection(ConnecString.POSDB_ADO)
        cn.Open()

        Using cm As MySqlCommand = cn.CreateCommand
            With cm
                .CommandType = CommandType.Text
                If crit.Key.ToLower = "dr" Then
                    .CommandText = FETCH_DR
                    tblName = "delivery_request"
                Else
                    .CommandText = FETCH_BYDATE
                    tblName = "inventory_journal"
                End If

                .Parameters.AddWithValue("?a", crit.JournalID)
                .Parameters.AddWithValue("?b", log.PubCompanyID)
                .Parameters.AddWithValue("?c", log.PubBranchID)
                da = New MySqlDataAdapter(cm)
                da.Fill(ds, tblName)
            End With
        End Using

        Using cm As MySqlCommand = cn.CreateCommand
            With cm
                .CommandType = CommandType.Text
                If crit.Key.ToLower = "dr" Then
                    .CommandText = FETCH_DR_ITEM
                    tblItemName = "delivery_request_item"
                Else
                    .CommandText = FETCH_TRX_ITEM
                    tblItemName = "inventory_journal_item"
                End If

                .Parameters.AddWithValue("?a", crit.JournalID)
                .Parameters.AddWithValue("?b", log.PubCompanyID)
                .Parameters.AddWithValue("?c", log.PubBranchID)
                da = New MySqlDataAdapter(cm)
                da.Fill(ds, tblItemName)
            End With
        End Using
    End Using

    dt = ds.Tables(tblName)

```

```

If dt.Rows.Count > 0 Then
    objXMLWriter.WriteStartElement("pos")
    objXMLWriter.WriteStartElement("table")
    objXMLWriter.WriteAttributeString("name", tblName)
    For Each dr As DataRow In dt.Rows
        objXMLWriter.WriteStartElement("row")
        For Each dc As DataColumn In dt.Columns
            With objXMLWriter
                .WriteElementString(dc.ColumnName,
                    dr(dc.ColumnName).ToString)
            End With
        Next
    objXMLWriter.WriteEndElement()
Next
objXMLWriter.WriteEndElement()
End If

dt = ds.Tables(tblItemName)
If dt.Rows.Count > 0 Then
    objXMLWriter.WriteStartElement("table")
    objXMLWriter.WriteAttributeString("name", tblItemName)
    For Each dr As DataRow In dt.Rows
        objXMLWriter.WriteStartElement("row")
        For Each dc As DataColumn In dt.Columns
            With objXMLWriter
                .WriteElementString(dc.ColumnName,
                    dr(dc.ColumnName).ToString)
            End With
        Next
    objXMLWriter.WriteEndElement()
Next
objXMLWriter.WriteEndElement()
End If

objXMLWriter.WriteEndElement()
If ws.ConformanceLevel <> ConformanceLevel.Fragment Then
    objXMLWriter.WriteEndDocument()
End If

objXMLWriter.Flush()
objXMLWriter.Close()

'open file
Dim xmlContents As String = GetFileContents(strFile).Trim
'RC6
Dim cipherRC6 As Crypto.Symmetric = New Crypto.Symmetric("sour",
Crypto.CipherType.RC6)
Dim cipherText As String = cipherRC6.Encrypt(xmlContents)

'3DES
Dim cipher3DES As Crypto.Symmetric = New
Crypto.Symmetric("sally", Crypto.CipherType.TRIPLE_DES)
Dim cipherText2 As String = cipher3DES.Encrypt(cipherText.Trim)

```

```

        SaveTextToFile(ciperText2, strFile)
        ZipData(_strPathFile, fName)

    Catch ex As Exception
        objXMLWriter.Flush()
        objXMLWriter.Close()
        File.Delete(strFile)
        MsgBox(ex.Message, MsgBoxStyle.Information, "Generate")
        Return False
    End Try
    Return True
End Function

```

4.4.2 Data Exchange

Data exchange merupakan *background application* yang berfungsi untuk konsolidasi/ pengiriman data ke *FTP Server*.

4.4.2.1 Client Side

a. Unggah Data

File Backup yang berupa *winrar* kemudian akan dikirim/ diunggah ke *FTP Server*, tentunya setelah dilakukan pengecekan ada tidaknya koneksi internet. Periode waktu untuk melakukan pengiriman *file* bisa diatur lewat menu *Setting*. Jadi selama *file* belum terkirim, maka secara periodik sistem akan mencoba mengirim kembali ke *FTP Server*. Pada Kode Program 4.2 adalah proses yang terjadi saat unggah data ke *FTP Server*.

Kode Program 4.2 Unggah Data ke FTP Server

```

Private Sub DoUpload()
    UpdateLogProgress("-----")
    -----")
    UpdateLogProgress(Date.Now.ToString("dd-MM-yyyy HH:mm:ss") & " :
Check Connection ...")
    Dim fileNames1 As String() =
Directory.GetFiles("C:\FTPConsolidation\FTP_Outbox\")
    Dim fileNames As New ArrayList(fileNames1)

    Dim br As Branch = Nothing
    Try
        br = Branch.GetBranch()
        UpdateLogProgress(Date.Now.ToString("dd-MM-yyyy HH:mm:ss") &
": Database connected ...")

```

```

    Catch ex As Exception
        UpdateLogProgress(Date.Now.ToString("dd-MM-yyyy HH:mm:ss") &
": " & ex.Message)
    End Try

    Dim conn As Boolean = isConnectionAvailable()

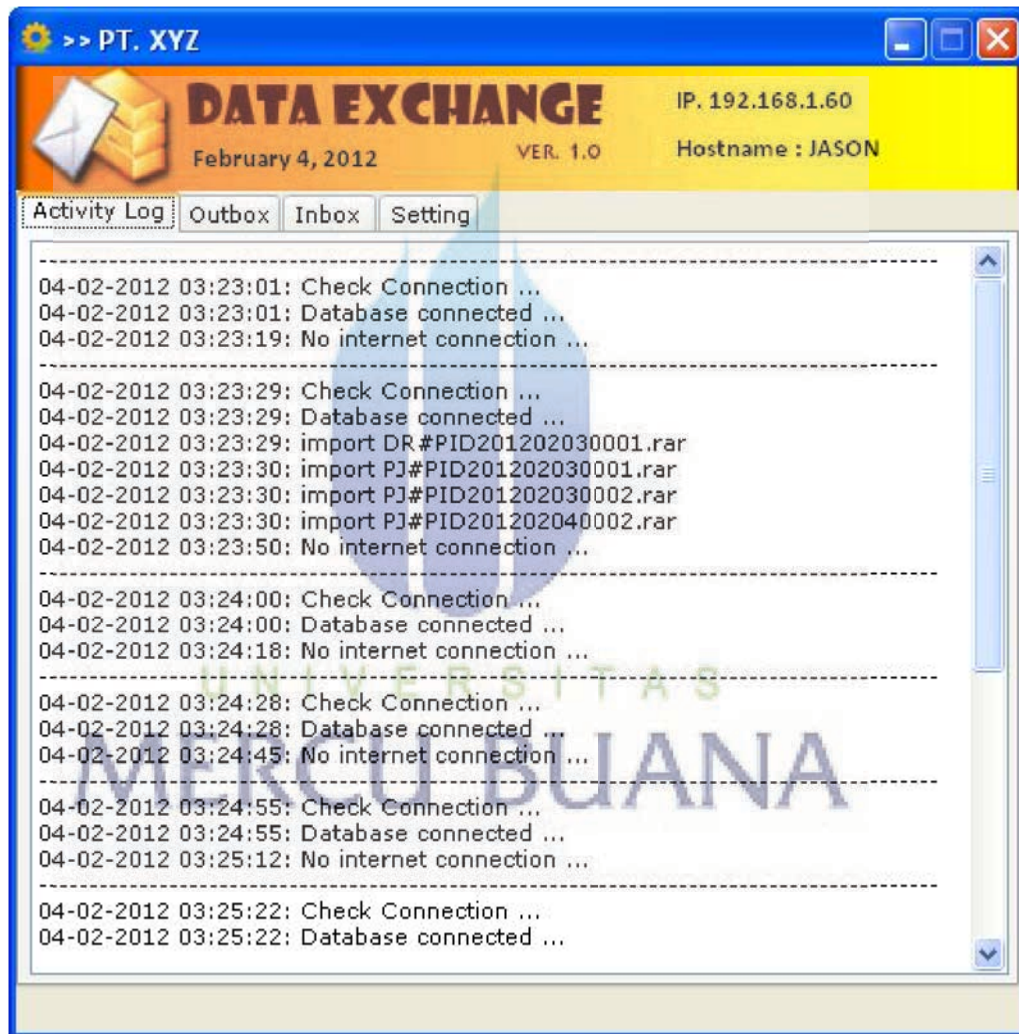
    If conn Then
        UpdateLogProgress(Date.Now.ToString("dd-MM-yyyy HH:mm:ss") &
": Internet connected ...")
    Else
        UpdateLogProgress(Date.Now.ToString("dd-MM-yyyy HH:mm:ss") &
": No internet connection ...")
    End If
    If Not br Is Nothing And conn Then
        Dim delFiles As ArrayList = New ArrayList
        Dim curval As Integer = 0
        Dim rowTotal As Integer = fileNames.Count
        If rowTotal > 0 Then
            Timer1.Enabled = False
            Dim ftp As New FTPClient(_objConfig.HostName,
_objConfig.UserName, _objConfig.Password)
            For Each fs As String In fileNames 'krm via FTP
                Dim fi As New FileInfo(fs)
                UpdateLogProgress(Date.Now.ToString("dd-MM-yyyy
HH:mm:ss") & ": sending " & fs)
                If ftp.Upload(fs, "/data/consolidation/HO/" & fi.Name)
Then
                    UpdateLogProgress(Date.Now.ToString("dd-MM-yyyy
HH:mm:ss") & ": sent successfully ...")
                    'Hapus file
                    File.Delete(fs)
                Else
                    UpdateLogProgress("The following problem occurred
when attempting to send your data")
                    Exit For
                End If
            Next
        Else
            UpdateLogProgress(Date.Now.ToString("dd-MM-yyyy HH:mm:ss")
& ": No file sending...")
        End If
    End If
End Sub

```


4.4.2.2 Server Side

a. Activity Log

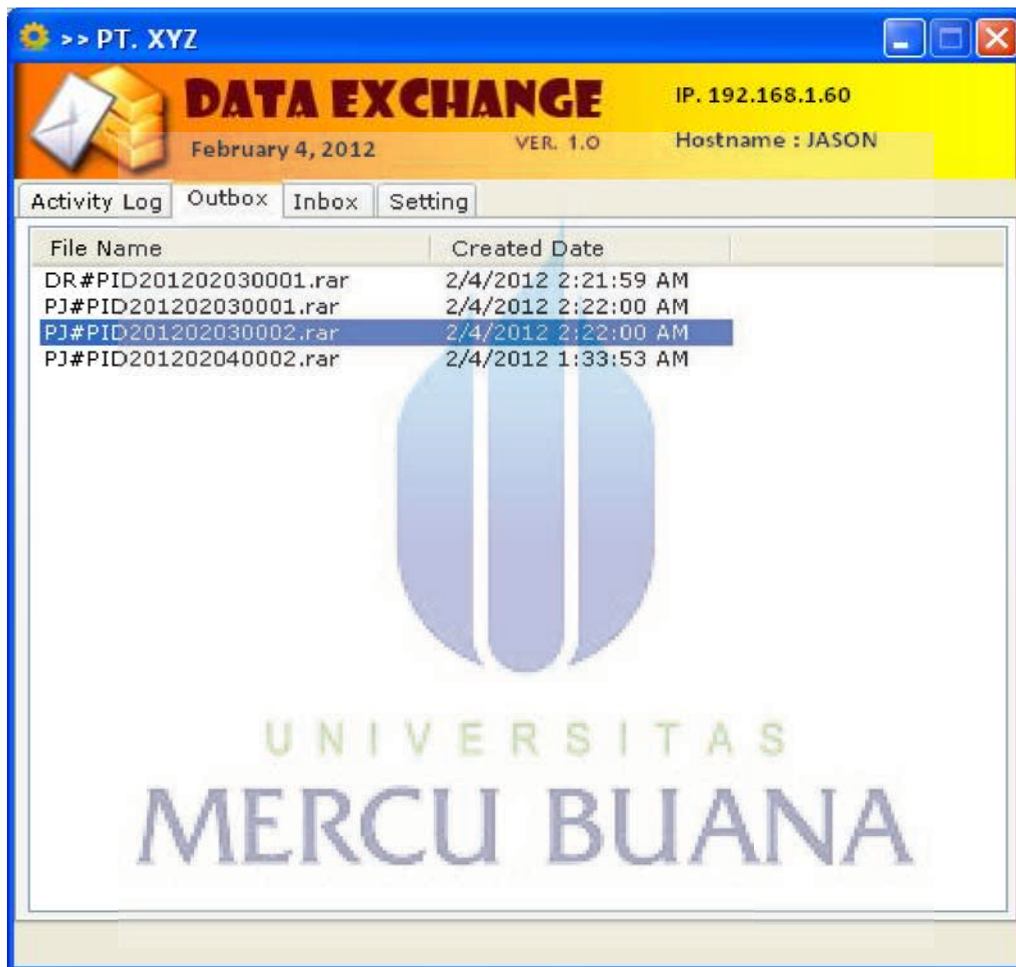
Activity log berisi ringkasan aktivitas yang dilakukan oleh program. Aktivitas disini tentunya yang berhubungan dengan proses *upload*, *download* serta *import* file ke dalam *database*.



Gambar 4.7 Activity Log

b. *Inbox dan Outbox*

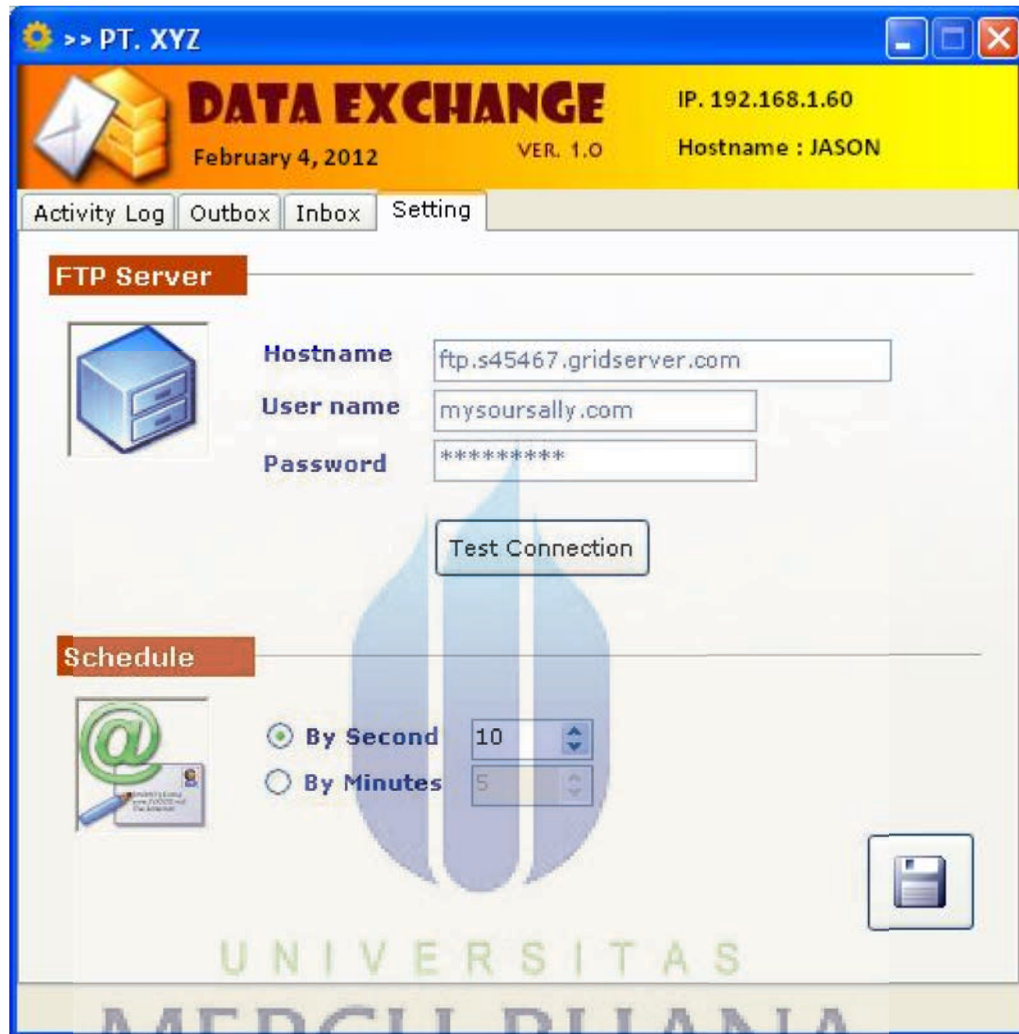
Tab Inbox berisi tentang file hasil mengunduh dari *FTP Server* namun belum di import ke dalam *database*. Sedangkan *outbox* sebaliknya, yaitu berisi file yang masih *pending* atau gagal terkirim ke *FTP server* dikarenakan tidak adanya koneksi *internet*.



Gambar 4.8 File *Outbox*

c. *Setting*

Tab setting digunakan untuk mengatur alamat *FTP Server* beserta interval waktu untuk mengunggah atau mengunduh *file*. Alamat *FTP server* yang diatur meliputi nama *server (hostname)*, *user name* dan *password*.



Gambar 4.9 Setting FTP Server

Berikut ini 2 tahap yang dilakukan *server* untuk proses konsolidasi data dari *outlet* ke kantor pusat (HO):

1. Unduh Data

Setiap periode tertentu akan melakukan pengecekan ada tidaknya *file* yang masuk ke FTP Server. Jika ada akan diunduh dan disimpan ke dalam *folder* C:\FTPConsolidation\FTP_Inbox untuk selanjutnya di-*import* ke dalam *database*. Kode Program 4.3 menunjukkan proses unduh data dari FTP Server.

Kode Program 4.3 Unduh Data dari FTP Server

```

Private Sub DoDownload() ' download
    UpdateLogProgress("-----
-----")
    UpdateLogProgress(Date.Now.ToString("dd-MM-yyyy HH:mm:ss") & ": Check
Connection ...")

    Dim br As Branch = Nothing
    Try
        br = Branch.GetBranch()
        UpdateLogProgress(Date.Now.ToString("dd-MM-yyyy HH:mm:ss") & ":
Database connected ...")
    Catch ex As Exception
        UpdateLogProgress(Date.Now.ToString("dd-MM-yyyy HH:mm:ss") & ": " &
ex.Message)
    End Try

    Dim conn As Boolean = isConnectionAvailable()
    If conn Then
        UpdateLogProgress(Date.Now.ToString("dd-MM-yyyy HH:mm:ss") & ":
Internet connected ...")
    Else
        UpdateLogProgress(Date.Now.ToString("dd-MM-yyyy HH:mm:ss") & ": No
internet connection ...")
    End If
    'download
    If Not br Is Nothing And conn Then
        Dim ftp As New FTPclient(_objConfig.HostName, _objConfig.UserName,
_objConfig.Password)
        'Get the detailed directory listing of /pub/
        Dim dirList As FTPdirectory =
ftp.ListDirectoryDetail("/data/consolidation/" & br.StrukID & "/")
        'filter out only the files in the list
        Dim filesOnly As FTPdirectory = dirList.GetFiles()
        If filesOnly.Count > 0 Then
            For Each file As FTPfileInfo In filesOnly
                If ftp.Download(file, "C:\FTPConsolidation\FTP_Inbox\" &
file.Filename, True) Then
                    'delete file di FTP Server
                    UpdateLogProgress(Date.Now.ToString("dd-MM-yyyy
HH:mm:ss") & ": download successfully ...")
                    ftp.FtpDelete(file.FullName)
                End If
            Next
        Else
            UpdateLogProgress(Date.Now.ToString("dd-MM-yyyy HH:mm:ss") & ":
No file download...")
        End If
    End If

    Dim fileNames1 As String() =

```

```

Directory.GetFiles("C:\FTPConsolidation\FTP_Inbox\")
Dim fileNames As New ArrayList(fileNames1)
'Import Data Jika ada file yg pending
If fileNames.Count > 0 Then
    For Each fs As String In fileNames 'krm via FTP
        Dim fi As New FileInfo(fs)
        UpdateLogProgress(Date.Now.ToString("dd-MM-yyyy HH:mm:ss") & ":
import " & fi.Name)
        'unzip file
        Dim unzipFileName As String = UnzipData(fi)
        'open file
        Dim cipherText As String = GetFileContents(unzipFileName).Trim
        '3DES
        Dim cipher3DES As Crypto.Symetric = New
Crypto.Symetric("sally", Crypto.CipherType.TRIPLE_DES)
        Dim cipherText2 As String = cipher3DES.Decrypt(cipherText)
        'RC6
        Dim cipherRC6 As Crypto.Symetric = New Crypto.Symetric("sour",
Crypto.CipherType.RC6)
        Dim xmlContents As String = cipherRC6.Decrypt(cipherText2.Trim)
        SaveTextToFile(xmlContents, unzipFileName)
        'Imports Data
        Try
            Dim m_xmld As XmlDocument = Nothing
            Dim m_nodelist As XmlNodeList = Nothing

            'Create the XML Document
            m_xmld = New XmlDocument()
            'Load the Xml file
            m_xmld.Load(unzipFileName)
            'Get the list of name nodes
            m_nodelist = m_xmld.SelectNodes("/pos/table")
            For Each m_node As XmlNode In m_nodelist
                If Not ImportToDatabase(m_node) Then
                    MsgBox("Import data not completed, please try again
!", MsgBoxStyle.Critical, "Error")
                    Exit Sub
                End If
            Next
            File.Delete(unzipFileName)
            File.Delete(fs)
        Catch ex As Exception
            MsgBox(ex.Message, MsgBoxStyle.Critical +
MsgBoxStyle.OkOnly, "Import Data")
        End Try
    Next
End If

End Sub

```


2. Import File ke Database

Tiap file hasil unduhan akan di dekripsi 2 kali menggunakan kata kunci yang sama (*simetris*) dengan kunci yang digunakan pada waktu proses enkripsi. Proses deskripsi menghasilkan *file text* yang berformat XML untuk selanjutnya di-*import* ke dalam *database*. Dengan demikian pada sisi *back end* dapat membaca data yang dibuat oleh *user* di *outlet*. Pada kode Program 4.4 merupakan kode program untuk proses import data ke dalam *database*.

Kode Program 4.4 Import File ke database

```
Private Function ImportToDatabase(ByVal xnd As XmlNode) As Boolean
    Dim result As Boolean = False
    Dim attributeName As String = xnd.Attributes.GetNamedItem("name").Value

    Dim ndList As XmlNodeList = xnd.SelectNodes("row")
    'Dim changeDelegate As New ChangeStatusLabel(AddressOf ChangeLabel)
    'Dim barDelegate As New Updatebar(AddressOf UpdateProgress)
    'Dim logDelegate As New UpdateLog(AddressOf UpdateLogProgress)

    Select Case attributeName
        Case "delivery_request"
            For Each nd As XmlNode In ndList
                Dim journalDate As Date =
CType(nd.ChildNodes.Item(0).InnerText, Date)
                Dim journalid As String = nd.ChildNodes.Item(1).InnerText
                Dim requestedBy As String = nd.ChildNodes.Item(2).InnerText
                Dim isOK As Boolean =
CType(nd.ChildNodes.Item(3).InnerText, Boolean)
                Dim createdBy As String = nd.ChildNodes.Item(4).InnerText
                Dim createDate As Date =
CType(nd.ChildNodes.Item(5).InnerText, Date)
                Dim updatedBy As String = nd.ChildNodes.Item(6).InnerText
                Dim updatedDate As Date =
CType(nd.ChildNodes.Item(7).InnerText, Date)
                Dim companyID As String = nd.ChildNodes.Item(8).InnerText
                Dim branchID As String = nd.ChildNodes.Item(9).InnerText
                Dim remark As String = nd.ChildNodes.Item(10).InnerText
                Dim vendorID As Integer =
IIf(nd.ChildNodes.Item(11).InnerText = "", 0, nd.ChildNodes.Item(11).InnerText)
                Dim deliveryDate As Date =
CType(nd.ChildNodes.Item(12).InnerText, Date)

                Dim obj As DRExim = Nothing

                obj = DRExim.GetDeliveryRequest(companyID, branchID,
journalid)
                If String.IsNullOrEmpty(obj.DrId) Then
```

```

obj = DRExim.NewDeliveryRequest()
With obj
    .DrDate = journalDate
    .DrId = journalid
    .RequestBy = requestedBy
    .Ok = isOK
    .CreatedBy = createdBy
    .CreatedDate = createdDate
    .UpdatedBy = updatedBy
    .UpdatedDate = updatedDate
    .CompanyID = companyID
    .BranchID = branchID
    .Remark = remark
    .VendorID = vendorID
    .DeliveryDate = deliveryDate
End With
Else
    With obj
        .DrDate = journalDate
        .DrId = journalid
        .RequestBy = requestedBy
        .Ok = isOK
        .CreatedBy = createdBy
        .CreatedDate = createdDate
        .UpdatedBy = updatedBy
        .UpdatedDate = updatedDate
        .CompanyID = companyID
        .BranchID = branchID
        .Remark = remark
        .VendorID = vendorID
        .DeliveryDate = deliveryDate
    End With
End If
If Not SaveDR(obj) Then
    Return False
End If
Next
result = True

Case "delivery_request_item"
    For Each nd As XmlNode In ndList
        Dim journalID As String = nd.ChildNodes.Item(0).InnerText
        Dim productID As String = nd.ChildNodes.Item(1).InnerText
        Dim reqQty As Double = IIf(nd.ChildNodes.Item(2).InnerText
= "", 0, nd.ChildNodes.Item(2).InnerText)
        CType(nd.ChildNodes.Item(3).InnerText, Double)
        Dim unit As String = nd.ChildNodes.Item(3).InnerText
        Dim price As Double = IIf(nd.ChildNodes.Item(4).InnerText
= "", 0, nd.ChildNodes.Item(4).InnerText)
        CType(nd.ChildNodes.Item(3).InnerText, Double)
        Dim createdBy As String = nd.ChildNodes.Item(4).InnerText

```

```

        Dim createdDate As Date =
CType(nd.ChildNodes.Item(5).InnerText, Date)
        Dim updatedBy As String = nd.ChildNodes.Item(6).InnerText
        Dim updatedDate As Date =
CType(nd.ChildNodes.Item(7).InnerText, Date)
        Dim companyID As String = nd.ChildNodes.Item(8).InnerText
        Dim branchID As String = nd.ChildNodes.Item(9).InnerText

        Dim obj As DRItemExim = Nothing

        obj = DRItemExim.GetDeliveryOrderItem(journalID, productID,
unit, companyID, branchID)
        If String.IsNullOrEmpty(obj.DrId) Then
            obj = DRItemExim.NewDeliveryOrderItem()
            With obj
                .DrId = journalID
                .productId = productID
                .Unit = unit
                .RequestedQuantity = reqQty
                .CreatedBy = createdBy
                .CreatedDate = createdDate
                .UpdatedBy = updatedBy
                .UpdatedDate = updatedDate
                .CompanyID = companyID
                .BranchID = branchID
            End With
        Else
            With obj
                .DrId = journalID
                .productId = productID
                .RequestedQuantity = reqQty
                .Unit = unit
                .CreatedBy = createdBy
                .CreatedDate = createdDate
                .UpdatedBy = updatedBy
                .UpdatedDate = updatedDate
                .CompanyID = companyID
                .BranchID = branchID
            End With
        End If
        If Not SaveDRItem(obj) Then
            Return False
        End If
    Next
    result = True
    Case "inventory_journal"
        For Each nd As XmlNode In ndList
            Dim tipeJournal As String = nd.ChildNodes.Item(0).InnerText
            Dim journalDate As Date =
CType(nd.ChildNodes.Item(1).InnerText, Date)
            Dim journalid As String = nd.ChildNodes.Item(2).InnerText
            Dim goodIssuesNo As String =

```

```

nd.ChildNodes.Item(3).InnerText
    Dim vendorID As String = nd.ChildNodes.Item(4).InnerText
    Dim isOK As Boolean =
CType(nd.ChildNodes.Item(5).InnerText, Boolean)
    Dim remark As String = nd.ChildNodes.Item(6).InnerText
    Dim createdBy As String = nd.ChildNodes.Item(7).InnerText
    Dim createDate As Date =
CType(nd.ChildNodes.Item(8).InnerText, Date)
    Dim updatedBy As String = nd.ChildNodes.Item(9).InnerText
    Dim updatedAt As Date =
CType(nd.ChildNodes.Item(10).InnerText, Date)
    Dim companyID As String = nd.ChildNodes.Item(11).InnerText
    Dim branchID As String = nd.ChildNodes.Item(12).InnerText
    Dim drID As String = nd.ChildNodes.Item(13).InnerText
    Dim branchFrom As String = nd.ChildNodes.Item(14).InnerText
    Dim branchTo As String = nd.ChildNodes.Item(15).InnerText
    Dim stockOpname As String =
nd.ChildNodes.Item(16).InnerText

    Dim obj As JournalExim = Nothing

journalid)
    obj = JournalExim.GetJournalExim(companyID, branchID,

    If String.IsNullOrEmpty(obj.JournalID) Then
        obj = JournalExim.NewJournalExim
        With obj
            .TypeOfJournal = tipeJournal
            .JournalDate = journalDate
            .JournalID = journalid
            .GoodIssuesNo = goodIssuesNo
            .VendorID = vendorID
            .Ok = isOK
            .Remark = remark
            .CreatedBy = createdBy
            .CreateDate = createDate
            .UpdatedBy = updatedBy
            .UpdatedAt = updatedAt
            .CompanyID = companyID
            .BranchID = branchID
            .DrID = drID
            .BranchFrom = branchFrom
            .BranchTo = branchTo
            .StockOpname = stockOpname
        End With
    Else
        With obj
            .TypeOfJournal = tipeJournal
            .JournalDate = journalDate
            .JournalID = journalid
            .GoodIssuesNo = goodIssuesNo
            .VendorID = vendorID
            .Ok = isOK

```

```

        .Remark = remark
        .CreatedBy = createdBy
        .CreateDate = createDate
        .UpdatedBy = updatedBy
        .UpdatedDate = updatedDate
        .CompanyID = companyID
        .BranchID = branchID
        .DrId = drId
        .BranchFrom = branchFrom
        .BranchTo = branchTo
        .StockOpname = stockOpname

        End With
    End If
    If Not SaveProductJournal(obj) Then
        Return False
    End If
Next
result = True

Case "inventory_journal_item"
    For Each nd As XmlNode In ndList
        Dim journalID As String = nd.ChildNodes.Item(0).InnerText
        Dim productID As String = nd.ChildNodes.Item(1).InnerText
        Dim unit As String = nd.ChildNodes.Item(2).InnerText
        Dim qty As Double = IIf(nd.ChildNodes.Item(3).InnerText =
"", 0, nd.ChildNodes.Item(3).InnerText) * CType(nd.ChildNodes.Item(3).InnerText,
Double)

        Dim price As Double = IIf(nd.ChildNodes.Item(4).InnerText =
"", 0, nd.ChildNodes.Item(4).InnerText) * CType(nd.ChildNodes.Item(3).InnerText,
Double)

        Dim createdBy As String = nd.ChildNodes.Item(5).InnerText
        Dim createDate As Date =
CType(nd.ChildNodes.Item(6).InnerText, Date)
        Dim updatedBy As String = nd.ChildNodes.Item(7).InnerText
        Dim updatedDate As Date =
CType(nd.ChildNodes.Item(8).InnerText, Date)
        Dim companyID As String = nd.ChildNodes.Item(9).InnerText
        Dim branchID As String = nd.ChildNodes.Item(10).InnerText

        Dim obj As JournalItemExim = Nothing

        obj = JournalItemExim.GetJournalItemExim(journalID,
productID, unit, companyID, branchID)
        If String.IsNullOrEmpty(obj.JournalID) Then
            obj = JournalItemExim.NewJournalItemExim()
            With obj
                .JournalID = journalID
                .ProductID = productID
                .Unit = unit
                .Price = price
                .CreatedBy = createdBy

```



```

        .CreateDate = createdDate
        .UpdatedBy = updatedBy
        .UpdatedDate = updatedDate
        .CompanyID = companyID
        .BranchID = branchID
    End With
Else
    With obj
        .JournalID = journalID
        .ProductID = productID
        .Unit = unit
        .CreatedBy = createdBy
        .CreateDate = createdDate
        .UpdatedBy = updatedBy
        .UpdatedDate = updatedDate
        .CompanyID = companyID
        .BranchID = branchID
    End With
    End If
    If Not SaveProductJournalItem(obj) Then
        Return False
    End If
Next
result = True

End Select
Return result
End Function

```

4.5 Pengujian

Pengujian yang dilakukan dibagi menjadi tiga bagian yaitu skema pengujian program, pengujian ukuran *backup file* dan pengujian performansi enkripsi - dekripsi.

4.5.1 Skema Pengujian Program

Skenario pengujian diperlukan untuk mengetahui apakah menu-menu yang ada dapat berfungsi dengan baik atau tidak. Skenario pengujian dilakukan dengan memilih menu yang ada pada halaman utama. Skenario ini dilakukan setiap menu dalam kondisi yang mungkin terjadi.

Tabel 4.6 Skenario pengujian untuk *client*

Tombol	Skenario Pengujian	Skenario yang diharapkan	Hasil Pengujian
Login	Memasukkan <i>user id</i> dan <i>password</i> kemudian klik <i>login</i>	Masuk ke <i>form</i> utama	OK
	Membuat transaksi inventory kemudian simpan	Data tersimpan	OK
Top	Klik tombol Top untuk pindah fokus ke baris pertama/ paling atas	Fokus pindah ke baris pertama	OK
Up	Klik tombol Up untuk pindah fokus ke baris di atasnya	Fokus pindah ke baris ke atas	OK
Down	Klik tombol Top untuk pindah fokus ke baris dibawahnya	Fokus pindah ke baris dibawahnya	OK
Bottom	Klik tombol Bottom untuk pindah fokus ke baris terakhir/ paling bawah	Fokus pindah ke baris terakhir/ paling bawah	OK
Pending	Klik tombol pending untuk menyimpan sementara transaksi	Transaksi tersimpan ke database	OK
OK	Klik tombol OK untuk menyimpan transaksi <i>final</i>	Transaksi tersimpan ke database dan backup data	OK

Tabel 4.7 Skenario pengujian untuk *server*

Tombol	Skenario Pengujian	Skenario yang diharapkan	Hasil Pengujian
<i>Save</i>	Merubah <i>hostname</i> , <i>username</i> , <i>password</i> dan <i>interval</i>	Data berhasil disimpan	OK
<i>Test Connection</i>	Setting FTP Server	Tampil pesan “ <i>connect</i> ” jika pengaturan benar dan “ <i>not connect</i> ” jika salah	OK

4.5.2 Pengujian Ukuran *Backup File*

Pengujian ini bertujuan untuk mengetahui besar *backup file* asli, setelah proses enkripsi dan setelah di – *compress*.

Tabel 4.8 Hasil pengujian ukuran file

Plain Text	Kunci	Enkripsi RC6	Enkripsi 3DES	Compress (*.rar)
17 kb	50ur5474	33 kb	65 kb	38 kb
17 kb	sour	33 kb	65 kb	38 kb
10 kb	50ur5474	20 kb	39 kb	23 kb
10 kb	sour	20 kb	39 kb	23 kb

4.5.3 Pengujian Performansi Deskripsi dan Enkripsi

Pengujian deskripsi dan enkripsi ini bertujuan untuk mengetahui lama proses enkripsi dan enkripsi. Tata cara pengujian ini dalah dengan menjalankan proses enkripsi dan deskripsi data dengan beberapa kunci dan mencatat waktunya. Kunci yang dipakai untuk pengujian ini yaitu “sour” untuk enkripsi RC6 dan “sally” untuk enkripsi 3DES.

Tabel 4.9 Hasil pengujian enkripsi

Plain Text	RC6		3DES	
	Cipher Text	Wkt	Cipher Text	Wkt
<pre><table name="delivery_request"> <row> <dr_date>1/27/2012 12:37:48 AM</dr_date> <dr_id>DR.SNC201201270001</dr_ id> <requested_by>Julian Mulya</requested_by> <ok>0</ok> <created_by>purwa</created_by> <created_date>1/27/2012 12:37:53 AM</created_date> <updated_by>purwa</updated_by> <updated_date>1/27/2012 9:32:26 AM</updated_date> <company_id>SS</company_id> <branch_id>SS.SC</branch_id> <remark>Urgent</remark> <vendor_id>1</vendor_id> <delivery_date>1/27/2012 9:32:11 AM</delivery_date> </row> </table></pre>	<pre>1778419b1f904f53fc8531b84ecbf41c3 346dc9bc88e085c479be15fd1d2af384c 126704e0d88d1e3e801efd314d3540ed5 df919f74d84aacba96b7cd17de7eebf0 1512aa188ed9fe4ef23756f011d6ae7e8 9aafc039f58edcc0b4c35449483e22e3e fa1de912f82bd9aaca731c8c1a65afcb bedd8bf078af1b10565be41b42f955b1c 30e209b77e233c7074dadd21df6554359 fc32fb32abcebe3855a8795b4079ee22b 00ec4e1c40c77104db55f1f5fe6ee05c3 336c9aa60d1ce0baf3ace81f34cd47cee b68f8cdcd53e6d40db397c180462b2672 bcbeaaf6bdd677070d96864aa049d8e1c f6d419308649255c2a9e42b15b558bdd4 9d7f02013563a129f8b77119f7065af5 d76e65d9250e67c58f4fb6a2696a6b596 4800fb0f7e8b102f4733534e5f807ef3 d10e80ad588d43b2748b4a492941ee3c3 fe1fe782515d7111f1c77a11ca8015e2c 00a28d0a422aeaa60c13fcc09d101076c 9cb5a0efd8f1255125f5814633f060cbf 5ea50ffc3c8f33130cb25101e05b4aa2 8e255a2641d0d73b12b5d1103419a5cfa 508a127005d30e488c4527785c258fdd0 0611df85bd59dbf50284897daca0d0ef7 da8be152cc536fa426a8e212ab3f80efe 2d2e6255418debc6ffce3387418609c36 b1fc54a420a0c2cd8a2d88750dc301652 4b16db85296de6a6ec437df8f98bf4857 396539bc9eaeac5f35964a9d5ec9e723 eae63dca5d26e8c019a85a07c1a6c56d7</pre>	31 ms	<pre>a770706e70e06aa1e865c2d9610a38f8c0eaf9d66cb2d68fdf3312dfae8 602b30470c21e5718cca72ce7e1e48faa52a94f4c0e0fc65b0cde82e4cf 0ec54924f601c33e221028e0596d390966c6762f383276642781129c813 87be624d013559a5a81b6848e45f66dad4c6074914e34d9a3a004ffd133 e2f90df5aae947649782effff37418e10adef202465dc6b496276c82086 d87bd54afdabf0f38d2657938028b1ea03baeda4d72ddb287987c450012 7c922d6eab52adc7c7d88bf3db852ab1fe292c078ed4cd96748e6b08a77 89f0b40808a2a48e6a466367116290eee74a6f6c34e29cff7ff567f85f2 9427122a7d7b2139f431c9bad421dfb439de6b660a9fb601d4068dac2f1 bbae332bbcd2667aa311bb38ca82ff4fd522a427935589a51c91b807008 2c571cf2f561a401b5c295e807e414dac84ca70c6868a5abada2cf4a3ac 561e0b47b344e0fbfc002d426ed1600916b75602debcece1137f27d0b7 a12328cb6d15f861c888dc2e4d5bb543e8e88f38b80572da956c0ed1cf0 5db4e53de8aa4f8e370e81ffa5200cc54128b3e980c460872af130b22b9 fa06e5290fb0badc5ee21f45fd419c37c20e4baf15613ee694a81ae63f5 f76bd2b465ae4ad4ba033b80197f4ba0425640f6436c0d87766d050b7cc f86be305c92794006139248beff1efd46b6e3069a092e473c91a1fcf3be 01379a2e44d6ae8b61fcd1df624d55654bc7e181d2b3510c86526b028c96 2e15f3bd4230fffe096437dc11b2ee21e5d8f293202db655ff0cfc148c 6c908189793a830d25abb63d2c9f0b949c8044ef7dadecb90c5d9d9d1f5 22e961d79fecc38f0173461af756e6fe344b843e0914582398ea08c2632 d9137724210805044cac47ace136d43a41d90bc288791aab2dcd9f50cd1 f549dc769692df066e6fb467b75d5fc07d079b26c09c421558b2d542180 6d88074b0308d4e14bb8b93e2f3f5844f3e51a09ded8aff78c978f73c11 b7f24279593a1395af63619cfb6fa62d4fa5885a38dc870524c11fb05ab 1aac51bb00a96850e787abb1a036be30bdaf318c8ea002b4dd91ee27d44 90f20f7a1a3bc5037641d04475b09d3eba53b745f8cc4b54431926fef1 e5601dfe44a80b7b80c03558ec463c7d80cd7cfa416b596b0b8cda3a167 51f2317510853e0bd42c1dfbfd13b07e9666975ee9006f21574092fe8da fa063d9971c757affe70ac2d253166b51060091b523968ee41530882ab a7f14d5f0f59743d0fe86f312922b2d69e8c9243cfef8a3e00feac9cb28 3fd8a1a7ba2613c602dfa5c7c154c795c7113311a0912702ed6b30ebae b5671569a58d883be73a01e646ee55ad3148b5fe584f7cbe88a15299fb6 3f1eb786ea7ad22369e8f32147bf97445f628f6c4706d98667eaeacd571d 1044cc118d7bb8d3615f80f531cf0aa4f740c9c35c02d72292e64ce5797 221293dd761ac4c78e36fec40096b824fef32490d28b654ce9d8ecca6b5 8e07</pre>	46 ms

Tabel 4.10 Hasil pengujian deskripsi

3DES		RC6		Plain Text
Cipher Text	Wkt	Cipher Text	Wkt	
a770706e70e06aale865c2d9610a38f8c0eaf9d66cb2d68fdf3312dfae8602 b30470c21e5718cca72ce7e1e48faa52a94f4c0e0fc65b0cde82e4cf0ec549 24f601c33e221028e0596d390966c6762f383276642781129c81387be624d0 13559a5a81b6848e45f66dad4c6074914e34d9a3a004ffd133e2f90df5aae9 47649782effff37418e10adef202465dc6b496276c82086d87bd54afdabf0f 38d2657938028b1ea03baeda4d72ddb287987c4500127c922d6eab52adc7c7 d88bf3db852ab1fe292c078ed4cd96748e6b08a7789f0b40808a2a48e6a466 367116290eee74a6f6c34e29cff7f567f85f29427122a7d7b2139f431c9ba d421dfb439de6b660a9fb601d4068dac2f1bbae332bbcd2667aa311bb38ca8 2ff4fd522a427935589a51c91b8070082c571cf2f561a401b5c295e807e414 dac84ca70c6868a5abada2cf4a3ac561e0b47b344e0fbfc002d426ed160091 6b75602debceee1137f27d0b7a12328cb6d15f861c888dc2e4d5bb543e8e8 8f38b80572da956c0ed1cf05db4e53de8aa4f8e370e81ffa5200cc54128b3e 980c460872af130b22b9fa06e5290fb0badc5ee21f45fd419c37c20e4baf15 613ee694a81ae63f5f76bd2b465ae4ad4ba033b80197f4ba0425640f4636c0 d87766d050b7ccf86be305c92794006139248bef1ef46b6e3069a092e473 c91a1fcf3be01379a2e44d6ae8b61fcd1df624d56554bc7e181d2b3510c8652 6b028c962e15f3b1d4230fffe096437dc11b2ee21e5d8f293202db655ff0cf c148c6c908189793a830d25abb63d2c9f0b949c8044ef7dadecb90c5d9d9d1 f522e961d79f6cc38f0173461af756e6fe344b843e0914582398ea08c2632d 9137724210805044cac47ace136d43a41d90bc288791aab2dcd9f50cd1f549 dc769692df066e6fb467b75d5fc07d079b26c09c421558b2d5421806d88074 b0308d4e14bb8b93e2f3f5844f3e51a09ded8aff78c978f73c11b7f2427959 3a1395af63619cfb6fa62d4fa5885a38dc870524c11fb05ab1aac51bb00a96 850e787abb1a036be30bdaf318c8ea002b4dd91ee27d4490f20f7a1a3bc503 7641d04475b09d3eba53b745ff8cc4b54431926fef1e5601dfe44a80b7b80c 03558ec463c7d80cd7cfa416b596b0b8cda3a16751f2317510853e0bd42c1d fbfd13b07e9666975ee9006f21574092fe8dafa063d9e971c757affe70acd2d 253166b51060091b523968ee41530882aba7f14d5fcf59743d0fe86f312922 b2d69e8c9243cfef8a3e00feac9cb283fd8a1a7ba2613c602dfa5c7c154c79 5c7113311a0912702ed6b30ebaebb5671569a58d883be73a01e646ee55ad31 48b5fe584f7cbe88a15299fb63f1eb786ea7ad22369e8f32147bf97445f628 f6c4706d98667eaecd571d1044cc118d7bb8d3615f80f531cf0aa4f740c9c3 5c02d72292e64ce5797221293dd761ac4c78e36fec40096b824fef32490d28 b654ce9d8ecca6b58e07	15 ms	1778419b1f904f53fc8531b84ecbf41 c3346dc9bc88e085c479be15fd1d2af 384c126704e0d88d1e3e801efd314d3 540ed5df919f74d84aacba96b7cd17d e7eebf01512aa188ed9fe4ef23756f 011d6ae7e89aafc039f58edcc0b4c35 449483e22e3efalde912f82bd9aaca7 31c8c1a65afcbbedd8bf078af1b105 65be41b42f955b1c30e209b77e233c7 074dadd21df6554359fc32fb32abceb e3855a8795b4079ee22b00ec4e1c40c 77104db55f1f5f5e6ee05c3336c9aa60 d1ce0baf3ace81f34cd47ceeb68f8cd cd53e6d40db397c180462b2672bcbea af6bdd677070d96864aa049d8e1cf6d 419308649255c2a9e42b15b558bdd49 d7f092013563a129f8b77119f7065af 5d76e65d9250e67c58f4fb6a2696a6b 5964800fbf0f7e8b102f4733534e5f8 07ef3d10e80ad588d43b2748b4a4929 41ee3c3fe1fe782515d7111f1c77a11 ca8015e2c00a28d0a422aeaa60c13fc c09d101076c9cb5a0efd8f1255125f5 814633f060cbf5ea50ffc3c8f33130c b825101e05b4aa28e255a2641d0d73b 12b5d1103419a5cfa508a127005d30e 488c4527785c258fdd00611df85bd59 dbf50284897daca0d0ef7da8be152cc 536fa426a8e212ab3f80efe2d2e6255 418debc6ffce3387418609c36b1fc54 a420a0c2cd8a2d88750dc3016524b16 db85296de6a6ec437df8f98bf485739 6539bc9eaeac535964a9d5ec9e723 eae63dca5d26e8c019a85a07c1a6c56 d7	15 ms	<table name="delivery_request"> <row> <dr_date>1/27/2012 12:37:48 AM</dr_date> <dr_id>DR.SNC201201270001</dr_id> <requested_by>Julian Mulya</requested_by> <ok>0</ok> <created_by>purwa</created_by> <created_date>1/27/2012 12:37:53 AM</created_date> <updated_by>purwa</updated_by> <updated_date>1/27/2012 9:32:26 AM</updated_date> <company_id>SS</company_id> <branch_id>SS.SC</branch_id> <remark>Urgent</remark> <vendor_id>1</vendor_id> <delivery_date>1/27/2012 9:32:11 AM</delivery_date> </row> </table>