

CHAPTER 7

SYSTEM IMPLEMENTATION AND CODING

7.0 Introduction

In order to translate the design into a machine-readable form, Microsoft Visual Basic 6.0 is used to develop the system. This programming language is not only an event-driven programming, but also is an object-oriented programming due to its supportiveness of object-oriented programming. The system that had been developed included the client-side application and server-side application. An introduction of object-oriented programming, Microsoft Visual Basic 6.0, COM Component as well as the algorithm and coding are discussed in this chapter.

7.1 Object-Oriented Programming

Object-oriented programming is a programming paradigm involving a collection of objects that interacted within each other. The objects are interacted by passing messages to transformed their state (Allen, 2002). The two most important concepts in object-oriented programming are the class and the object.

A class is a kind of mold or template that is used to create objects, which the data type is bounded together with the initializations and other operations (Thomas, 2001; Allen, 2002). An object is an instance of a class, an instance of exactly one class and an instance of a class belongs to the class (Thomas, 2001).

There are few criteria in object-oriented programming, abstraction, encapsulation, polymorphism and inheritance. Abstraction allowed management of the complexities of a

problem by allowing identifying a set of objects involved with that problem. Encapsulation enables the internal implementation of an abstraction being kept hidden within the particular object. Polymorphism provides multiple implementations of the same method on different objects. For example, different objects can have a same method name but each of the method performs a different processing. Inheritance allows the reused of interface and the implementation of a class (Microsoft Corporation, 2004).

7.1.1 Microsoft Visual Basic 6.0

In general, Visual Basic is a programming environment that is specifically designed to facilitate the creation of new programs. Visual Basic 6 is an event-driven programming. Besides, Visual Basic 6 also has an object orientation significance to support object-oriented programming.

As an event-driven programming, Visual Basic 6 allowed programmer designed the program starting from GUI, including creating the graphical objects such as buttons and menu. After that, programmer only needs to write code to the related events, such as button click event and menu click event (William, 2000).

To support an object-oriented programming, Visual Basic 6 provides the capability to construct objects within a program that combine data and instructions. Objects are software components that include data elements and behavior. Everything that works with Visual Basic were objects, such as forms, controls, printers and databases.

Visual Basic also allows programmer to create components ranging from code libraries to Automation-enabled applications. Besides, Visual Basic also allows programmer to merge with COM component and creation COM components (Microsoft Press, 1999).

7.1.2 COM Component

Component Object Model (COM) is a standard or a model for the interaction of objects. A COM component is a unit of executable code that provides specific functionality. COM components can either be an internal components or external components. Internal components are components that compile into a project and are available only to that project. External components are components that compile into an executable file (.exe) or dynamic-link library (.dll).

Visual Basic allowed programmer creates three types of COM components, such as ActiveX controls, DLLs, and ActiveX documents. COM code components, such as ActiveX DLLs and ActiveX EXEs are composed of one or more class modules in a Visual Basic project. ActiveX controls are COM components that provide a user interface.

COM components interacted with application or other components through a client-server relationship. The client used the features of a component where the server was the component and associated objects (Microsoft Press, 1999).

7.2 Code

Coding is a process of turning program logic into specific instructions that the computer system could execute. For doing so, programming languages is used to transform the program logic into code statements. In this chapter, some of the important algorithm, coding and the description are given. An algorithm is a finite instruction for performing a computation or for solving a problem. The given algorithm, coding and description are the implementation of functions discussed in chapter 5.

7.2.1 Login Page

Login page is the place where user registers to the system and the place for the system recognized the existing users. This page only has two functions, included log in user and register user.

7.2.1.1 Log in user

This function is used to log in a registered user.

Algorithm:

```
//Perform log in user.  
Procedure cmdLogin_Click ()  
  
If username field and password field were not empty then  
    Call lnkCmdLogin_Click function on ucMainWindow  
End if
```

Coding:

```
Private Sub cmdLogin_Click()  
    If clUsername.Text <> "" And tlPassword.Text <> "" Then  
        UserControl.Parent.lnkCmdLogin_Click clUsername.Text, tlPassword.Text  
    End If  
End Sub
```

Figure 7.1 Source code for log in user

7.2.1.2 Register user

This function is used to register a new user to the system.

Algorithm:

```
//Perform register user.  
Procedure cmdRegister_Click ()  
  
If username field and password field and confirm password field were empty then  
    Notice user to type again  
Else  
    If got blank space in username field then  
        Notice user to re-enter again  
    End if  
  
    If the password and the confirm password were not the same then  
        Notice user to re-enter again
```

```

End if

Check for exiting user
If user not found then
    Call lnkCmdRegisterUser on ucMainWindow
End if
End if

```

Coding:

```

Private Sub cmdRegister_Click()
    If trUsername.Text = "" Or trPassword.Text = "" Or trConfirmPassword.Text = "" Then
        MsgBox "Please enter all the required fields.", vbInformation, "LOOP"
    Else
        If InStr(1, trUsername.Text, " ", vbTextCompare) <> 0 Then
            MsgBox "No blank space(s) between username.", vbInformation, "LOOP"
            Exit Sub
        End If

        If trPassword.Text <> trConfirmPassword.Text Then
            MsgBox "'Password' and 'Confirm password' are not same.", vbInformation, "LOOP"
            Exit Sub
        End If

        Dim i As Integer
        For i = 0 To clUsername.ListCount - 1
            If LCase(clUsername.List(i)) = LCase(trUsername.Text) Then
                MsgBox "User name found.", vbInformation, "LOOP"
                Exit Sub
            End If
        Next i
        UserControl.Parent.lnkCmdRegisterClick trUsername.Text, trPassword.Text
    End If
End Sub

```

Figure 7.2 Source code for register user

7.2.2 Learning Contents Page

Learning contents page is the place for registered user to manage the personal learning contents. This page provides several functions that is used to manage the contents.

7.2.2.1 Send contents by email

This function is used to send currently viewed content and note to other user by email through client mailing system.

Algorithm:

```

//Perform send contents by email
Procedure SendEmail()

```

Write the current viewed content to LOOP language

Send it by email using client mailing system as attachment.

Coding:

```
Public Sub SendEmail()
    On Error GoTo errhandler
    If strUser = "" Then
        MsgBox "Please log in first.", vbInformation, "LOOP"
        Exit Sub
    End If
    If stMainWindow.Tab = 1 Then
        Dim strContents As String
        strContents = "{|LOOP|}{|LTITLE|}" & ucLearningControl.ScreenTitle & "{|/LTITLE|}"
        strContents = strContents & "{|LCONTENT|}" & ucLearningControl.ScreenHTMLText & "{|/LCONTENT|}"
        strContents = strContents & "{|LNOTE|}" & ucLearningControl.NoteText & "{|/LNOTE|}{|/LOOP|}"
        Dim CallclsFile As clsFile
        Set CallclsFile = New clsFile
        CallclsFile.WriteTextToFile strFileDir & "\ " & strUser & ".rLOOP", strContents
        Session.SignOn
        Messages.SessionID = Session.SessionID
        Messages.Compose
        Messages.MsgSubject = "LOOP content's file: " & ucLearningControl.ScreenTitle
        Messages.AttachmentPathName = strFileDir & "\ " & strUser & ".rLOOP"
        Messages.Send True
        Session.SignOff
        CallclsFile.DeleteFile strFileDir & "\ " & strUser & ".rLOOP"
        Set CallclsFile = Nothing
    End If
errhandler:
    If Err.Number <= 0 Then
        MsgBox "Cannot send email.", vbInformation, "LOOP"
    End If
End Sub
```

Figure 7.3 Source code for send contents by email

7.2.2.2 New page

This function is used to create a new content page.

Algorithm:

```
//Perform new page
Procedure NewContent()

Load ucNewContentControl
```

Coding:

```
Public Sub NewContent()
    If strUser = "" Then
        MsgBox "Please log in first.", vbInformation, "LOOP"
        Exit Sub
    End If
    ucNewContentControl.AddNode ucLearningControl.mtopic, mtopic
    stMainWindow.Tab = 1
    stMainWindow_GotFocus
    ucLearningControl.Visible = False
    ucNewContentControl.Visible = True
End Sub
```

Figure 7.4 Source code for new page

7.2.2.3 Open file

This function is used to open LOOOP file, web page, XML file, picture file or multimedia file.

Algorithm:

```
//Perform open file  
Procedure OpenPage()  
  
Get the file path  
Checked the file extension  
Call related page to load the file
```

Coding:

```
Public Sub OpenPage()  
    If strUser = "" Then  
        MsgBox "Please log in first.", vbInformation, "LOOOP"  
    Exit Sub  
End If  
cdl.Filter = "LOOOP file, XML page and Web page|*.htm;*.html;*.xml;*.rLOOOP"  
cdl.Filter = cdl.Filter & "|" & "Image, shockwave flash and multimedia file|*.jpg;*.bmp;*.gif;*.swf;*.avi;*.wav"  
cdl.ShowOpen  
If cdl.FileName <> "" Then  
    Dim clsTemp As clsFile  
    Set clsTemp = New clsFile  
    If clsTemp.FileExist(cdl.FileName) = True Then  
        Select Case LCase(clsTemp.GetFileExtension(cdl.FileName))  
            Case "html", "htm", "xml":  
                Dim strTemp As String  
                If Len(cdl.FileTitle) > 55 Then  
                    strTemp = Left(cdl.FileTitle, 24) & "~" & Mid(cdl.FileTitle, Len(cdl.FileTitle) - 31)  
                Else  
                    strTemp = cdl.FileTitle  
                End If  
                ucSearchControl.OpenFile strTemp, "<wb>" & cdl.FileName  
            Case "rloop":  
                ucSearchControl.OpenLOOOPPage cdl.FileName  
            Case "bmp", "jpg", "gif":  
                ucSearchControl.OpenPictureFile cdl.FileName  
            Case "avi", "wav", "swf":  
                ucSearchControl.OpenMultimediaFile cdl.FileName  
        End Select  
        cdl.FileName = ""  
        stMainWindow.Tab = 2  
        stMainWindow_GotFocus  
    End If  
    Set clsTemp = Nothing  
End If  
End Sub
```

Figure 7.5 Source code for open file

7.2.2.4 Save to database

This function is used to save the current viewed content to database.

Algorithm:

```
//Perform save to database  
Procedure SaveToDatabase()  
  
Call SaveContentToDatabase on ucLearningContent
```

Coding:

```
Public Sub SaveToDatabase()  
    If strUser = "" Then  
        MsgBox "Please log in first.", vbInformation, "LOOPP"  
        Exit Sub  
    End If  
    If ucLearningControl.Visible = True Then  
        ucLearningControl.SaveContentToDatabase  
    End If  
  
    If ucViewerControl.Visible = True Then  
        ucViewerControl.SaveFile  
    End If  
End Sub
```

Figure 7.6 Source code for save to database on learning contents page

7.2.2.5 Save as LOOPP file

This function is used to save the current viewed content as LOOPP file.

Algorithm:

```
//Perform save as LOOPP file  
Procedure SaveAsLOOPPFile()  
  
Call SaveAsLOOPPFile on ucLearningContent
```

Coding:

```
Public Sub SaveAsLOOPPFile()  
    If strUser = "" Then  
        MsgBox "Please log in first.", vbInformation, "LOOPP"  
        Exit Sub  
    End If  
    If ucViewerControl.Visible = True Then  
        MsgBox "Cannot save as LOOPP file.", vbInformation, "LOOPP"  
        Exit Sub  
    End If  
    If ucLearningControl.Visible = True Then  
        ucLearningControl.SaveAsLOOPPFile  
    End If  
End Sub
```

Figure 7.7 Source code for save as LOOPP file

7.2.2.6 Save as XML file

This function is used to save the current viewed content as XML file.

Algorithm:

```
//Perform save as XML file  
Procedure SaveAsXMLFile()  
  
Write content into XML language  
Save the file in XML format
```

Coding:

```
Public Sub SaveAsXMLFile()  
    If strUser = "" Then  
        MsgBox "Please log in first.", vbInformation, "LOOPP"  
        Exit Sub  
    End If  
    If ucViewerControl.Visible = True Then  
        MsgBox "Cannot save as XML file.", vbInformation, "LOOPP"  
        Exit Sub  
    End If  
    If ucLearningControl.Visible = True Then  
        cdl.Filter = "XML file|*.XML"  
        cdl.ShowSave  
        If cdl.FileName <> "" Then  
            Dim strContents As String  
            strContents = "<?xml version='1.0' encoding='utf-8'?><LOOPP><LTITLE>" & ucLearningControl.ScreenTitle & "</LTITLE>"  
            strContents = strContents & "<LCONTENT>" & ucLearningControl.ScreenText & "</LCONTENT>"  
            strContents = strContents & "<LNOTE>" & ucLearningControl.NoteText & "</LNOTE></LOOPP>"  
            Dim CallclsFile As clsFile  
            Set CallclsFile = New clsFile  
            CallclsFile.WriteTextToFile cdl.FileName, strContents  
            Set CallclsFile = Nothing  
            cdl.FileName = ""  
        End If  
    End If  
End Sub
```

Figure 7.8 Source code for save as XML file

7.2.2.7 Save as sharable learning contents

This function is used to save the current viewed content to sharable learning contents

Algorithm:

```
//Perform save as sharable learning contents  
Procedure SharableContent()  
  
Call SaveSharableContent on ucLearningContent
```

Coding:

```
Public Sub SharableContent()  
    If strUser = "" Then  
        MsgBox "Please log in first.", vbInformation, "LOOP"  
        Exit Sub  
    End If  
    If ucLearningControl.Visible = False Then  
        MsgBox "Only learning content(s) can be share within users.", vbInformation, "LOOP"  
        Exit Sub  
    End If  
    If ucLearningControl.Visible = True Then  
        If ucLearningControl.ScreenTitleIndex <> 0 Then  
            ucLearningControl.SaveSharableContent mtopic  
        End If  
    End If  
End Sub
```

Figure 7.9 Source code for save as sharable learning contents

7.2.2.8 Delete contents

This function is used to delete the current viewed content.

Algorithm:

```
//Perform Delete contents  
Procedure DeleteContent()  
  
Call DeleteContent on ucLearningContent
```

Coding:

```
Public Sub DeleteContent()  
    If strUser = "" Then  
        MsgBox "Please log in first.", vbInformation, "LOOP"  
        Exit Sub  
    End If  
    If stMainWindow.Tab = 1 Then  
        ucLearningControl.DeleteContent  
    End If  
End Sub
```

Figure 7.10 Source code for delete contents

7.2.2.9 Print content

This function is used to print the current viewed content and note.

Algorithm:

```
//Perform print content  
Procedure PrintContent()  
  
Integrated the content and note into HTML page  
Print out the HTML page
```

Coding:

```
Public Sub PrintContent()
    If strUser = "" Then
        MsgBox "Please log in first.", vbInformation, "LOOP"
        Exit Sub
    End If
    If stMainWindow.Tab = 1 Then
        Dim strhtml As String
        strhtml = "<HTML><TABLE><TR><TD><H2><U><B>Contents</B></U></H2></TD></TR><TR><TD>"
        strhtml = strhtml & ucLearningControl.ScreenHTMLText
        strhtml = strhtml & "</TD></TR></TABLE><BR><BR><TABLE><TR><TD><H2><U><B>Note</B></U></H2></TD></TR><TR><TD>"
        strhtml = strhtml & ucLearningControl.NoteText
        strhtml = strhtml & "</TD></TR></TABLE></HTML>"

        Dim aDoc As IHTMLDocument2
        Set aDoc = wbPrint.Document
        aDoc.body.innerHTML = strhtml
        wbPrint.ExecWB OLECMDID_PRINTPREVIEW, OLECMDEXEPT_DODEFAULT
        Set aDoc = Nothing
    ElseIf stMainWindow.Tab = 2 Then
        If ucSearchControl.PrintScreen = False Then
            MsgBox "Cannot print the page. Please try again later.", vbInformation, "LOOP"
        End If
    End If
End Sub
```

Figure 7.11 Source code for print content

7.2.2.10 Navigation buttons

These functions includes first page, previous page, next page function and last page function. These functions are used for navigating personal learning contents. For example, the algorithm and coding for previous page were showed.

Algorithm:

```
//Perform navigate previous page
Procedure GoPreviousChapter()

If there was a chapter before the currently viewed chapter then
    Go to previous content
End if
```

Coding:

```
Private Sub GoPreviousChapter()
    If tvContent.Nodes.Count > 0 Then
        If tvContent.SelectedItem.index <= 1 Then
            tvContent.SelectedItem = tvContent.Nodes(tvContent.SelectedItem.index - 1)
            tvContent_NodeClick tvContent.SelectedItem
        End If
    End If
    tvContent.SetFocus
End Sub
```

Figure 7.12 Source code for navigate previous page

7.2.2.11 Editing tools

These functions include cut, copy, paste and others. These functions are used for editing the currently viewed content. For example, the algorithm and coding for cut function are shown.

Algorithm:

```
//Perform cut function  
Procedure CutText()  
  
Call DHTML control built-in cut function
```

Coding:

```
Private Sub CutText()  
    DHTMLScreen.execCommand DECMD_CUT, OLECMDEXEPT_DODEFAULT  
End Sub
```

Figure 7.13 Source code for cut function

7.2.2.12 Internet search

This function is used to perform Internet searching for current selected text on currently viewed content.

Algorithm:

```
//Perform Internet searching  
Procedure InternetSearch()  
  
Call InternetSearching function on ucMainWindow
```

Coding:

```
Private Sub InternetSearch()  
    If DHTMLScreen.DOM.selection.Type = "Text" Then  
        UserControl.Parent.InternetSearching DHTMLScreen.DOM.selection.createRange.Text  
    End If  
End Sub
```

Figure 7.14 Source code for Internet search

7.2.2.13 Reader agent

This function is used to play text reading.

Algorithm:

```
//Perform reader agent read text
Function ActivateReader(bTrue: Boolean)

If there is a selected text then
    Read the selected text
Else
    Read the full content
End if
```

Coding:

```
Public Function ActivateReader(bTrue As Boolean) As Boolean
    On Error GoTo errhandler
    If bTrue = True Then
        If tts.IsSpeaking = 1 Then
            tts.StopSpeaking
        End If
        If DHTMLScreen.DOM.selection.createRange.Text <> "" Then
            tts.Speak DHTMLScreen.DOM.selection.createRange.Text
        Else
            If DHTMLScreen.DOM.body.innerText <> "" Then
                tts.Speak DHTMLScreen.DOM.body.innerText
            End If
        End If
        DHTMLToolBar2.Buttons(13).Enabled = True
    Else
        tts.StopSpeaking
        DHTMLToolBar2.Buttons(13).Enabled = False
    End If
errhandler:
    If Err.Number <> 0 Then
        If Err.Number <> 35600 Then
            Exit Function
        End If
    End If
End Function
```

Figure 7.15 Source code for reader agent

7.2.2.14 Highlight text and clear highlighted text

These functions are used to highlight the text and clear the highlighted text. For example, the highlight text function algorithm and coding are shown.

Algorithm:

```
//Perform highlight text
Function HighlightText(index: Integer)

If there is a selected text then
```

Highlight the text with specific color
End if

Coding:

```
Private Sub HighlightText(index As Integer)
    If DHTMLScreen.DOM.selection.Type = "Text" Then
        If DHTMLScreen.QueryStatus(DECMD_COPY) >= DECMDF_ENABLED Then
            DHTMLScreen.execCommand DECMD_SETBACKCOLOR, OLECMDEXECHOPT_DODEFAULT, FormatRGBString(Shape(index).FillColor)
            SetFontID index, FormatRGBString(Shape(index).FillColor)
        End If
    Else
        MsgBox "Please perform highlight on text only.", vbInformation, "Mix status"
    End If
End Sub
```

Figure 7.16 Source code for highlight text

7.2.2.15 Play board file

This function is used to play the associated file for current viewed content.

Algorithm:

```
//Perform play board file
Function ListView_NodeClick(Node: MSComctlLib.Node)

    Convert the file string to file
    Call BoardItemClick function on ucMainWindow to open the file
```

Coding:

```
Private Sub ListView_NodeClick(ByVal Node As MSComctlLib.Node)
    Dim clsTemp As clsFile
    Set clsTemp = New clsFile
    clsTemp.DeleteFile strFileDir & "LOOOPTemp\" & strUser & "." & mBoard(CInt(Mid(Node.Key, 5))).fType
    clsTemp.ConvertStringToFile strFileDir & "LOOOPTemp\" & strUser & "." & mBoard(CInt(Mid(Node.Key, 5))).fType, _
        mBoard(CInt(Mid(Node.Key, 5))).fData
    Set clsTemp = Nothing
    UserControl.Parent.BoardItemClick mBoard(CInt(Mid(Node.Key, 5))).fTitle, strFileDir & _
        "LOOOPTemp\" & strUser & "." & mBoard(CInt(Mid(Node.Key, 5))).fType, mBoard(CInt(Mid(Node.Key, 5))).fDescription, _
        CInt(mBoard(CInt(Mid(Node.Key, 5))).fID)
End Sub
```

Figure 7.17 Source code for play board file

7.2.3 Search Information Page

Search information page is the place for user searching the keyword from the Internet or sharable learning contents. The result page found can also a source for user to add it as learning contents.

7.2.3.1 Search from Internet or contents

This function is used to search the keyword from Internet or sharable learning contents.

Algorithm:

```
//Perform search from Internet or contents.  
Procedure cmdSearch_Click()  
  
If search from contents checkbox is selected then  
    Call SearchFromDBS on ucMainWindow  
End if  
If search from contents checkbox is selected then  
    Set web browser navigate to google.com search page  
End if
```

Coding:

```
Private Sub cmdSearch_Click()  
    If Trim(tSearch.Text) <> "" Then  
        tvResult.Nodes.Clear  
        If chkContents.Value = vbChecked Then  
            UserControl.Parent.SearchFromDBS tSearch.Text  
        End If  
  
        If chkInternet.Value = vbChecked Then  
            wb.Navigate2 "http://www.google.com.my/search?q=" & tSearch.Text & "&num=30&hl=en&lr=&as_qdr=all&start=150&sa=N"  
        End If  
    End If  
End Sub
```

Figure 7.18 Source code for search from Internet and contents

7.2.3.2 Save to contents

This function is used to save the current viewed page as content.

Algorithm:

```
//Perform save to contents.  
Procedure cmdOK_Click()  
  
Check for the image file included inside the page  
If any image file is found then  
    Get the file path from the list and download it  
    Convert the file to string  
    Delete the downloaded file  
End if  
Call SaveSearchPageToContent on ucMainWindow to save the file
```

Coding:

```
Private Sub cmdOK_Click()  
    If cChapter.Text = "" Then  
        MsgBox "Please create a chapter before saving a new content.", vbInformation, "LOOP"   
        fSaveFile.Visible = False  
        UserControl_Resize  
        Exit Sub  
    End If  
    If MsgBox("Are you sure want to save the page?", vbQuestion + vbYesNo, "Save page") = vbYes Then  
        Dim clsTemp As clsContentFileCollection  
        Set clsTemp = New clsContentFileCollection  
        Dim clstempfile As clsFile  
        Set clstempfile = New clsFile  
        If WebBrowser.Document.images.length > 0 Then  
            Dim i As Integer  
            Dim strData As String  
            Dim aDoc As IHTMLDocument2  
            Set aDoc = WebBrowser.Document  
            Dim img As IHTMLImageElement  
            For i = 0 To aDoc.images.length - 1  
                Set img = aDoc.images(i)  
                If clstempfile.DownloadFile(img.src, strFileDir & "LOOOPTemp\" & clstempfile.GetFileName(img.src)) = True Then  
                    strData = clstempfile.ConvertFileToString(strFileDir & "LOOOPTemp\" & clstempfile.GetFileName(img.src))  
                    clsTemp.Add clstempfile.GetFileName(img.src), strData, cChapter.Text & ">" & cAddress.Text  
                End If  
                img.src = clstempfile.GetFileName(img.src)  
                Set img = Nothing  
            Next i  
        End If  
        UserControl.Parent.SaveSearchPageToContent cChapter.Text, cAddress.Text, WebBrowser.Document.body.innerHTML, clsTemp  
        Set clsTemp = Nothing  
        If cAddress.SelectedItem.Image <> 1 Then  
            WebBrowser.Refresh  
        End If  
        Set clstempfile = Nothing  
        Set aDoc = Nothing  
    End If  
End Sub
```

Figure 7.19 Source code for save to contents

7.2.3.3 Save as board file

This function is used to save the multimedia file or picture file to selected content.

Algorithm:

```
//Perform save as board file.  
Procedure cmdFileOK_Click()  
  
    Check for the selected file from file list  
    If the file is selected then  
        Get the file path from the list and download it  
        Convert the file to string  
        Delete the downloaded file  
    End if  
    Call SaveToBoard on ucMainWindow to save the file
```


Coding:

```
Private Sub cmdFileOK_Click()  
    If cChapter2.Text = "" Then  
        MsgBox "Please create a content before saving a new file.", vbInformation, "LOOP"   
        fSaveFile.Visible = False  
        UserControl_Resize  
        Exit Sub  
    End If  
    If MsgBox("Are you sure want to save the file(s)?", vbQuestion + vbYesNo, "Save file(s)") = vbYes Then  
        Dim clsTemp As clsBoardFileCollection  
        Set clsTemp = New clsBoardFileCollection  
        Dim clstempfile As clsFile  
        Set clstempfile = New clsFile  
        Dim strData As String  
        Dim i As Integer  
        For i = 1 To tv.Nodes.Count  
            If tv.Nodes(i).Checked = True Then  
                If clstempfile.DownloadFile(Mid(tv.Nodes(i).Key, 12), strFileDir & "LOOPTemp\" & clstempfile.GetFileName(Mid(tv.Nodes(i).Key, 12))) = True Then  
                    strData = clstempfile.ConvertFileToString(strFileDir & "LOOPTemp\" & clstempfile.GetFileName(Mid(tv.Nodes(i).Key, 12)))  
                    clsTemp.Add tv.Nodes(i).Text, "-", Left(tv.Nodes(i).Key, 3), strData  
                    clstempfile.DeleteFile strFileDir & "LOOPTemp\" & clstempfile.GetFileName(Mid(tv.Nodes(i).Key, 12))  
                End If  
            End If  
        Next i  
        UserControl.Parent.SaveToBoard cChapter2.Text, clsTemp  
        Set clstempfile = Nothing  
        Set clsTemp = Nothing  
    End If  
    fSaveFile.Visible = False  
    UserControl_Resize  
End Sub
```

Figure 7.20 Source code for save as board file

7.2.4 View File Page

View file page is the place for user to view and manage the board file, which is the file that is associated with the personal learning contents.

7.2.4.1 Save to database

This function is used to save the modified file information of currently viewed file.

Algorithm:

//Perform save to database.

Procedure SaveFile()

Call InkModifyBoardFile function on ucMainWindow

Coding:

```

Public Sub SaveFile()
    UserControl.Parent.lnkModifyBoardFile txtTitle.Text, txt.Text, intFileOpenID
End Sub

```

Figure 7.21 Source code for save to database on view file page

7.2.4.2 Save as external file

This function is used to save the currently viewed file to original format file.

Algorithm:

```

//Perform save as external file.
Procedure SaveAsFile (strDestPath: String)

Include clsFile class
Call copy file function to save the file on selected file path

```

Coding:

```

Public Sub SaveAsFile(strDestPath As String)
    Dim clsTemp As clsFile
    Set clsTemp = New clsFile
    clsTemp.CopyFile strFileOpen, strDestPath
    Set clsTemp = Nothing
End Sub

```

Figure 7.22 Source code for save as external file

7.2.4.3 Delete file

This function is used to delete currently viewed file.

Algorithm:

```

//Perform delete file.
Function DeleteBoardFile ()

If file id is not empty then
    Call DeleteBoardFile on ucMainWindow
End if

```

Coding:

```

Public Function DeleteBoardFile()
    If intFileID <> 0 Then
        UserControl.Parent.DeleteBoardFile intFileID
    End If
End Function

```

Figure 7.23 Source code for delete file

7.2.5 Forum Window

Forum window is the place for registered user to communicate with another online LOOOP registered user.

7.2.5.1 Send file

This function is used to send the selected file to another online LOOOP registered user.

Algorithm:

```
//Perform send file.
Procedure cmdSendFile_Click ()

Find whether any online registered user selected as receiver
If got then
    Set strTemp //LOOOP language for selected user name
End if
Read the file path
Convert the file to string
Set the strTemp again with filename and file data
Sent to LOOOP Socket and sent to selected users
```

Coding:

```
Private Sub cmdSendFile_Click()
    If lstUser.ListCount <= 1 Then
        Exit Sub
    End If
    Dim i As Integer, intcount As Integer
    Dim strTemp As String
    intcount = 0
    For i = 0 To lstUser.ListCount - 1
        If lstUser.Selected(i) = True Then
            If lstUser.List(i) <> strUser Then
                intcount = intcount + 1
                strTemp = strTemp & "{|RECEIVER" & intcount & "|}" & lstUser.List(i) & _
                    "{|/RECEIVER" & intcount & "|}"
            End If
        End If
    Next i
    If intcount = 0 Then
        Exit Sub
    End If
    strTemp = "{|COUNT|}" & intcount & "{|/COUNT|}" & strTemp
    Dim clsTemp As clsFile
    Set clsTemp = New clsFile
    cdl.ShowOpen
    If cdl.FileName <> "" Then
        If clsTemp.FileExist(cdl.FileName) = True Then
            strFileStatement = strTemp & "{|FILENAME|}" & clsTemp.GetFileName(cdl.FileName) & _
                "{|/FILENAME|}{|DATA|}" & clsTemp.ConvertFileToString(cdl.FileName) & _
                "{|/DATA|}{|END|}"
            If wFile.State <> sckConnected Then
                Timer1_Timer
            Else
                wFile_Connect
            End If
        End If
    End If
    cdl.FileName = ""
    Set clsTemp = Nothing
End Sub
```

Figure 7.24 Source code for send file

7.2.5.2 Send message

This function is used to send the message that typed in the message input area on forum window.

Algorithm:

```
//Perform send message.  
Procedure cmdSend_Click ()  
  
Set strStatement //LOOOP language that include the user message typed  
If the socket is disconnect then  
    Call timer to connect it  
Else  
    Sent the message  
End if
```

Coding:

```
Private Sub cmdSend_Click()  
    strStatement = "[" & strUser & "]" & vbCrLf & txtMsg.Text & "{|END|}"  
    If wCommn.State <> sckConnected Then  
        Timer1_Timer  
    Else  
        wCommn_Connect  
    End If  
End Sub
```

Figure 7.25 Source code for send message

7.3 Summary

System implementation and coding is a step to turn the designed system process flow and data flow to a reality system. In order to change from design to reality, Microsoft Visual Basic 6.0 is selected as the programming language to develop the system. After the selection of programming language, algorithms are well defined to convert the data flow diagram into a text format. Based on the written algorithms, the codes are written and some tests are performed on the written codes in the next steps of software development life cycle.