Imports System.Data.OleDb

Public Class CountrySelector

Inherits System.Windows.Forms.Form

Public admin As CCAdminTool

Public QC As CustomerCenterQC

Public ingest As Data\_Ingestion\_Tool

Private Sub CountrySelector\_Load(sender As Object, e As EventArgs) Handles MyBase.Load

On Error GoTo ErrorHandler

Me.FormBorderStyle = FormBorderStyle.FixedDialog

Done:

If GlobalVariables.BOTHCOUNTRIES = False Then

Me.Hide()

End If

Exit Sub

ErrorHandler:

DisplayError(Err.Source, Err.Description, "CountrySelector.UserForm\_Initialize", Erl)

Resume Done

End Sub

Private Sub selectCountryBtn\_Click(sender As Object, e As EventArgs) Handles selectCountryBtn.Click

'On Error GoTo ErrorHandler

If auOptionButton.Checked = True Then

GlobalVariables.HISTINDTABLE = "Hist\_AU\_Ind\_ASW"

GlobalVariables.HISTENTTABLE = "Hist\_AU\_ENTITY\_ASW"

GlobalVariables.HIST\_CASE\_ID = "S\_id\_from\_au\_post"

GlobalVariables.COUNTRY = "Australia"

GlobalVariables.LINKEDCASESTABLE = "Au\_Linked\_Cases"

GlobalVariables.ESCALATIONSTABLE = "AU\_CC\_Escalations"

GlobalVariables.CUSTOMERTABLE = "AU\_CC\_Customer"

GlobalVariables.CONVERSATIONTABLE = "AU\_CC\_Conversation"

GlobalVariables.DBPATH = GlobalVariables.DBPATHAU

GlobalVariables.CC\_CUSTOMER = "AU\_CC\_Customer"

GlobalVariables.CC\_QC\_MAIN = "AU\_CC\_QC\_Main"

GlobalVariables.COUNTRYPRE = "AU"

GlobalVariables.CC\_CONVERSATION = "AU\_CC\_Conversation"

Me.Dispose()

If GlobalVariables.CCADMINAU = True Then

If admin Is Nothing Then

admin = New CCAdminTool

admin.CountrySel = Me

End If

CCAdminTool.ShowDialog()

ElseIf GlobalVariables.CCQCAU = True Then

If QC Is Nothing Then

QC = New CustomerCenterQC

QC.QCSel = Me

End If

CustomerCenterQC.ShowDialog()

ElseIf GlobalVariables.CCINGESTAU = True Then

If ingest Is Nothing Then

ingest = New Data\_Ingestion\_Tool

ingest.IngestSel = Me

End If

Data\_Ingestion\_Tool.ShowDialog()

End If

ElseIf nzOptionButton.Checked = True Then

GlobalVariables.HISTINDTABLE = "Hist\_NZ\_Ind\_ASW"

GlobalVariables.HISTENTTABLE = "Hist\_NZ\_ENTITY\_ASW"

GlobalVariables.HIST\_CASE\_ID = "S\_id\_from\_nz\_post"

GlobalVariables.COUNTRY = "New Zealand"

GlobalVariables.LINKEDCASESTABLE = "NZ\_Linked\_Cases"

GlobalVariables.ESCALATIONSTABLE = "NZ\_CC\_Escalations"

GlobalVariables.CUSTOMERTABLE = "NZ\_CC\_Customer"

GlobalVariables.CONVERSATIONTABLE = "NZ\_CC\_Conversation"

GlobalVariables.CC\_CUSTOMER = "NZ\_CC\_Customer"

GlobalVariables.CC\_QC\_MAIN = "NZ\_CC\_QC\_Main"

GlobalVariables.COUNTRYPRE = "NZ"

GlobalVariables.CC\_CONVERSATION = "NZ\_CC\_Conversation"

GlobalVariables.DBPATH = GlobalVariables.DBPATHNZ

Me.Dispose()

If GlobalVariables.CCADMINNZ = True Then

If admin Is Nothing Then

admin = New CCAdminTool

admin.CountrySel = Me

End If

CCAdminTool.ShowDialog()

ElseIf GlobalVariables.CCQCNZ = True Then

If QC Is Nothing Then

QC = New CustomerCenterQC

QC.QCSel = Me

End If

CustomerCenterQC.ShowDialog()

ElseIf GlobalVariables.CCINGESTNZ = True Then

If ingest Is Nothing Then

ingest = New Data\_Ingestion\_Tool

ingest.IngestSel = Me

End If

Data\_Ingestion\_Tool.ShowDialog()

End If

Else

MsgBox("Please select a country", vbInformation)

End If

Done:

Exit Sub

ErrorHandler:

DisplayError(Err.Source, Err.Description, "CountrySelector.selectCountryBtn\_Click", Erl)

Resume Done

End Sub

Private Sub cancelbtn\_Click(sender As Object, e As EventArgs) Handles cancelbtn.Click

On Error GoTo ErrorHandler

Me.Close()

Done:

Exit Sub

ErrorHandler:

DisplayError(Err.Source, Err.Description, "CountrySelector.cancelbtn\_Click", Erl)

'Resume Done

End Sub

End Class

Public Class myStartupClass

Public Shared Sub Main()

' On Error GoTo ErrorHandler

Application.EnableVisualStyles()

If readConfigfile() = False Then

'sql = "Select count(\*) From CustomerTable"

If authenticateusers() = False Then

MsgBox("ERROR: " & GlobalVariables.USERLANID & " does not have permission to use this system. Access Denied.")

Application.Exit()

'Else

' MsgBox("User " & GlobalVariables.USERFULLNAME & " Authenticated !")

Else

If GlobalVariables.FOUNDAU = True And GlobalVariables.FOUNDNZ = True Then

GlobalVariables.BOTHCOUNTRIES = True

End If

If GlobalVariables.BOTHCOUNTRIES = False Then

If GlobalVariables.CCADMINAU = True Or GlobalVariables.CCADMINNZ = True Then

CCAdminTool.ShowDialog()

ElseIf GlobalVariables.CCQCAU = True Or GlobalVariables.CCQCNZ = True Then

CustomerCenterQC.ShowDialog()

ElseIf GlobalVariables.CCINGESTAU = True Or GlobalVariables.CCINGESTNZ = True Then

Data\_Ingestion\_Tool.ShowDialog()

End If

Else

CountrySelector.ShowDialog()

End If

End If

End If

Done:

Exit Sub

ErrorHandler:

DisplayError(Err.Source, Err.Description, "myStartupClass.Main", Erl)

'Resume Done

End Sub

End Class

Public Class GlobalVariables

Public Shared DBPATHAU As String

Public Shared DBPATHNZ As String

Public Shared POSTDBPATH As String

Public Shared DBPATH As String

Public Shared CJDBpath As String

Public Shared CONN As OleDbConnection

Public Shared USERLANID As String

Public Shared USERFULLNAME As String

Public Shared AlreadyUsed As Boolean

Public Shared LINE\_NO\_TEXT As String

Public Shared HISTINDTABLE As String

Public Shared HISTENTTABLE As String

Public Shared LINKEDCASESTABLE As String

Public Shared ESCALATIONSTABLE As String

Public Shared CC\_QC\_MAIN As String

Public Shared COUNTRY As String

Public Shared BOTHCOUNTRIES As Boolean

Public Shared COUNTRYPRE As String

Public Shared COUNTRYCAP As String

Public Shared CC\_CONVERSATION As String

Public Shared CC\_CUSTOMER As String

Public Shared HIST\_CASE\_ID As String

Public Shared PROGVER As String

Public Shared FOUNDAU As Boolean

Public Shared FOUNDNZ As Boolean

Public Shared CUSTIDSEL As Boolean

Public Shared CCQCAU As Boolean

Public Shared CCADMINAU As Boolean

Public Shared CCINGESTAU As Boolean

Public Shared CCQCNZ As Boolean

Public Shared CCADMINNZ As Boolean

Public Shared CCINGESTNZ As Boolean

Public Shared CUSTOMERTABLE As String

Public Shared CONVERSATIONTABLE As String

Public Shared Reploc As String

Public Shared ESCReploc As String

Public Shared TROFAReploc As String

Public Shared NZFLAGMGR As Boolean

Public Shared NZFLAGING As Boolean

End Class

Imports System.Data.OleDb

Public Class CountrySelector

Inherits System.Windows.Forms.Form

Public admin As CCAdminTool

Public QC As CustomerCenterQC

Public ingest As Data\_Ingestion\_Tool

Private Sub CountrySelector\_Load(sender As Object, e As EventArgs) Handles MyBase.Load

On Error GoTo ErrorHandler

Me.FormBorderStyle = FormBorderStyle.FixedDialog

Done:

If GlobalVariables.BOTHCOUNTRIES = False Then

Me.Hide()

End If

Exit Sub

ErrorHandler:

DisplayError(Err.Source, Err.Description, "CountrySelector.UserForm\_Initialize", Erl)

Resume Done

End Sub

Private Sub selectCountryBtn\_Click(sender As Object, e As EventArgs) Handles selectCountryBtn.Click

'On Error GoTo ErrorHandler

If auOptionButton.Checked = True Then

GlobalVariables.HISTINDTABLE = "Hist\_AU\_Ind\_ASW"

GlobalVariables.HISTENTTABLE = "Hist\_AU\_ENTITY\_ASW"

GlobalVariables.HIST\_CASE\_ID = "S\_id\_from\_au\_post"

GlobalVariables.COUNTRY = "Australia"

GlobalVariables.LINKEDCASESTABLE = "Au\_Linked\_Cases"

GlobalVariables.ESCALATIONSTABLE = "AU\_CC\_Escalations"

GlobalVariables.CUSTOMERTABLE = "AU\_CC\_Customer"

GlobalVariables.CONVERSATIONTABLE = "AU\_CC\_Conversation"

GlobalVariables.DBPATH = GlobalVariables.DBPATHAU

GlobalVariables.CC\_CUSTOMER = "AU\_CC\_Customer"

GlobalVariables.CC\_QC\_MAIN = "AU\_CC\_QC\_Main"

GlobalVariables.COUNTRYPRE = "AU"

GlobalVariables.CC\_CONVERSATION = "AU\_CC\_Conversation"

Me.Dispose()

If GlobalVariables.CCADMINAU = True Then

If admin Is Nothing Then

admin = New CCAdminTool

admin.CountrySel = Me

End If

CCAdminTool.ShowDialog()

ElseIf GlobalVariables.CCQCAU = True Then

If QC Is Nothing Then

QC = New CustomerCenterQC

QC.QCSel = Me

End If

CustomerCenterQC.ShowDialog()

ElseIf GlobalVariables.CCINGESTAU = True Then

If ingest Is Nothing Then

ingest = New Data\_Ingestion\_Tool

ingest.IngestSel = Me

End If

Data\_Ingestion\_Tool.ShowDialog()

End If

ElseIf nzOptionButton.Checked = True Then

GlobalVariables.HISTINDTABLE = "Hist\_NZ\_Ind\_ASW"

GlobalVariables.HISTENTTABLE = "Hist\_NZ\_ENTITY\_ASW"

GlobalVariables.HIST\_CASE\_ID = "S\_id\_from\_nz\_post"

GlobalVariables.COUNTRY = "New Zealand"

GlobalVariables.LINKEDCASESTABLE = "NZ\_Linked\_Cases"

GlobalVariables.ESCALATIONSTABLE = "NZ\_CC\_Escalations"

GlobalVariables.CUSTOMERTABLE = "NZ\_CC\_Customer"

GlobalVariables.CONVERSATIONTABLE = "NZ\_CC\_Conversation"

GlobalVariables.CC\_CUSTOMER = "NZ\_CC\_Customer"

GlobalVariables.CC\_QC\_MAIN = "NZ\_CC\_QC\_Main"

GlobalVariables.COUNTRYPRE = "NZ"

GlobalVariables.CC\_CONVERSATION = "NZ\_CC\_Conversation"

GlobalVariables.DBPATH = GlobalVariables.DBPATHNZ

Me.Dispose()

If GlobalVariables.CCADMINNZ = True Then

If admin Is Nothing Then

admin = New CCAdminTool

admin.CountrySel = Me

End If

CCAdminTool.ShowDialog()

ElseIf GlobalVariables.CCQCNZ = True Then

If QC Is Nothing Then

QC = New CustomerCenterQC

QC.QCSel = Me

End If

CustomerCenterQC.ShowDialog()

ElseIf GlobalVariables.CCINGESTNZ = True Then

If ingest Is Nothing Then

ingest = New Data\_Ingestion\_Tool

ingest.IngestSel = Me

End If

Data\_Ingestion\_Tool.ShowDialog()

End If

Else

MsgBox("Please select a country", vbInformation)

End If

Done:

Exit Sub

ErrorHandler:

DisplayError(Err.Source, Err.Description, "CountrySelector.selectCountryBtn\_Click", Erl)

Resume Done

End Sub

Private Sub cancelbtn\_Click(sender As Object, e As EventArgs) Handles cancelbtn.Click

On Error GoTo ErrorHandler

Me.Close()

Done:

Exit Sub

ErrorHandler:

DisplayError(Err.Source, Err.Description, "CountrySelector.cancelbtn\_Click", Erl)

'Resume Done

End Sub

End Class

Public Class myStartupClass

Public Shared Sub Main()

' On Error GoTo ErrorHandler

Application.EnableVisualStyles()

If readConfigfile() = False Then

'sql = "Select count(\*) From CustomerTable"

If authenticateusers() = False Then

MsgBox("ERROR: " & GlobalVariables.USERLANID & " does not have permission to use this system. Access Denied.")

Application.Exit()

'Else

' MsgBox("User " & GlobalVariables.USERFULLNAME & " Authenticated !")

Else

If GlobalVariables.FOUNDAU = True And GlobalVariables.FOUNDNZ = True Then

GlobalVariables.BOTHCOUNTRIES = True

End If

If GlobalVariables.BOTHCOUNTRIES = False Then

If GlobalVariables.CCADMINAU = True Or GlobalVariables.CCADMINNZ = True Then

CCAdminTool.ShowDialog()

ElseIf GlobalVariables.CCQCAU = True Or GlobalVariables.CCQCNZ = True Then

CustomerCenterQC.ShowDialog()

ElseIf GlobalVariables.CCINGESTAU = True Or GlobalVariables.CCINGESTNZ = True Then

Data\_Ingestion\_Tool.ShowDialog()

End If

Else

CountrySelector.ShowDialog()

End If

End If

End If

Done:

Exit Sub

ErrorHandler:

DisplayError(Err.Source, Err.Description, "myStartupClass.Main", Erl)

'Resume Done

End Sub

End Class

Public Class GlobalVariables

Public Shared DBPATHAU As String

Public Shared DBPATHNZ As String

Public Shared POSTDBPATH As String

Public Shared DBPATH As String

Public Shared CJDBpath As String

Public Shared CONN As OleDbConnection

Public Shared USERLANID As String

Public Shared USERFULLNAME As String

Public Shared AlreadyUsed As Boolean

Public Shared LINE\_NO\_TEXT As String

Public Shared HISTINDTABLE As String

Public Shared HISTENTTABLE As String

Public Shared LINKEDCASESTABLE As String

Public Shared ESCALATIONSTABLE As String

Public Shared CC\_QC\_MAIN As String

Public Shared COUNTRY As String

Public Shared BOTHCOUNTRIES As Boolean

Public Shared COUNTRYPRE As String

Public Shared COUNTRYCAP As String

Public Shared CC\_CONVERSATION As String

Public Shared CC\_CUSTOMER As String

Public Shared HIST\_CASE\_ID As String

Public Shared PROGVER As String

Public Shared FOUNDAU As Boolean

Public Shared FOUNDNZ As Boolean

Public Shared CUSTIDSEL As Boolean

Public Shared CCQCAU As Boolean

Public Shared CCADMINAU As Boolean

Public Shared CCINGESTAU As Boolean

Public Shared CCQCNZ As Boolean

Public Shared CCADMINNZ As Boolean

Public Shared CCINGESTNZ As Boolean

Public Shared CUSTOMERTABLE As String

Public Shared CONVERSATIONTABLE As String

Public Shared Reploc As String

Public Shared ESCReploc As String

Public Shared TROFAReploc As String

Public Shared NZFLAGMGR As Boolean

Public Shared NZFLAGING As Boolean

End Class

Imports outlook = Microsoft.Office.Interop.Outlook

Imports System.Data.OleDb

Module Mailer

Public managermail As String

Public managername As String

Public ccmanagermail As String

Public agentmail As String

Public Sub Mail\_workbook\_Outlook()

'On Error GoTo ErrorBlock

Dim cmd As OleDbCommand

Dim mailBody As String, dateStr As String, timeStr As String

Dim sSqlQry As String

Dim reader As OleDbDataReader

Dim location As String

'Public managermail As String

'Dim managername As String

'Dim ccmanagermail As String

connectToDB(GlobalVariables.CONN, GlobalVariables.DBPATH, "MSAccess")

Dim key As String = DirectCast(CustomerCenterQC.AgentNameComboBox.SelectedItem, KeyValuePair(Of String, String)).Key

Dim value As String = DirectCast(CustomerCenterQC.AgentNameComboBox.SelectedItem, KeyValuePair(Of String, String)).Value

sSqlQry = "Select Email\_Address, Location From Authenticatedusers WHERE LanID = '" & key & "'"

cmd = New OleDbCommand(sSqlQry, GlobalVariables.CONN)

reader = cmd.ExecuteReader

Try

If reader.HasRows Then

While reader.Read

agentmail = reader(0)

location = reader(1)

End While

End If

Catch

MsgBox("No Email/Location assigned to the user")

GoTo Done

End Try

'MsgBox(agentmail)

closeRecordSet(cmd, reader)

If location = "On-Shore" Then

sSqlQry = "Select PARAM\_VALUE, FILT1 From CUSTOM\_PARAMETERS WHERE PARAM\_TYPE = 'CC\_MANAGER\_MAIL\_ID\_ONSHORE'"

ElseIf location = "Off-Shore" Then

sSqlQry = "Select PARAM\_VALUE, FILT1 From CUSTOM\_PARAMETERS WHERE PARAM\_TYPE = 'CC\_MANAGER\_MAIL\_ID\_OFFSHORE'"

End If

cmd = New OleDbCommand(sSqlQry, GlobalVariables.CONN)

Dim oledbReader As OleDbDataReader = cmd.ExecuteReader()

If oledbReader.HasRows Then

While oledbReader.Read

managermail = oledbReader.Item(0)

managername = oledbReader.Item(1)

End While

Else

disconnectFromDB(GlobalVariables.CONN)

MsgBox("Missing Parameter for Manager Mail. Please check with DB admin")

GoTo Done

End If

closeRecordSet(cmd, oledbReader)

'If location = "On-Shore" Then

' sSqlQry = "Select FILT1 From CUSTOM\_PARAMETERS WHERE PARAM\_TYPE = 'CC\_MANAGER\_MAIL\_ID\_ONSHORE'"

'ElseIf location = "Off-Shore" Then

' sSqlQry = "Select FILT1 From CUSTOM\_PARAMETERS WHERE PARAM\_TYPE = 'CC\_MANAGER\_MAIL\_ID\_OFFSHORE'"

'End If

'cmd = New OleDbCommand(sSqlQry, GlobalVariables.CONN)

'oledbReader = cmd.ExecuteReader()

'If oledbReader.HasRows Then

' While oledbReader.Read

' managername = oledbReader.Item(1)

' End While

'Else

' disconnectFromDB(GlobalVariables.CONN)

' MsgBox("Missing Parameter for Manager Name. Please check with DB admin")

' GoTo Done

'End If

'closeRecordSet(cmd, oledbReader)

sSqlQry = "Select \* From CUSTOM\_PARAMETERS WHERE PARAM\_TYPE = 'CC\_MANAGER\_MAIL\_ID\_CC'"

cmd = New OleDbCommand(sSqlQry, GlobalVariables.CONN)

oledbReader = cmd.ExecuteReader()

If oledbReader.HasRows Then

While oledbReader.Read

ccmanagermail = oledbReader.Item(1)

End While

Else

disconnectFromDB(GlobalVariables.CONN)

MsgBox("Missing Parameter for Manager Name. Please check with DB admin")

GoTo Done

End If

closeRecordSet(cmd, oledbReader)

Dim OutlookMessage As outlook.MailItem

Dim AppOutlook As New outlook.Application

Dim objNS As outlook.\_NameSpace = AppOutlook.Session

Dim objFolder As outlook.MAPIFolder

objFolder = objNS.GetDefaultFolder(outlook.OlDefaultFolders.olFolderDrafts)

disconnectFromDB(GlobalVariables.CONN)

OutlookMessage = AppOutlook.CreateItem(outlook.OlItemType.olMailItem)

Dim Recipents As outlook.Recipients = OutlookMessage.Recipients

Dim recipientTo As outlook.Recipient = Nothing

Dim recipientCC As outlook.Recipient = Nothing

Dim recipientBCC As outlook.Recipient = Nothing

dateStr = Format(CustomerCenterQC.CallDateTextBox.Value, "dd/MMM/yyyy")

timeStr = Format(CustomerCenterQC.CallDateTextBox.Value, "hh:mm:ss tt")

If CustomerCenterQC.OAComboBox.Text = "Pass" Then

Try

mailBody = "<HTML><BODY>Hi " & value & ",<br><br>" &

Chr(13) & Chr(13) &

"A Quality Assessment has been completed for a call taken on <b>" & dateStr & "</b> at <b>" & timeStr & ".</b><br><br>" &

Chr(13) & Chr(13) &

"<b>Policy and Compliance:</b> <br>"

For iCnt = 0 To (CustomerCenterQC.PandCListBoxMet.Items.Count - 1)

If CustomerCenterQC.PandCListBoxMet.Items(iCnt) <> "" Then

mailBody = mailBody & CustomerCenterQC.PandCListBoxMet.Items(iCnt) & ": <font color = green>Met</font><br>"

End If

Next iCnt

For iCnt = 0 To (CustomerCenterQC.PandCListBoxNotMet.Items.Count - 1)

If CustomerCenterQC.PandCListBoxNotMet.Items(iCnt) <> "" Then

mailBody = mailBody & CustomerCenterQC.PandCListBoxNotMet.Items(iCnt) & ": <font color = red>Not Met</font><br>"

End If

Next iCnt

mailBody = mailBody & "<br><b>Review:</b> " & CustomerCenterQC.PACTextBox.Text & "<br><br>"

mailBody = mailBody & "<b>Business Accuracy:</b><br>"

For iCnt = 0 To (CustomerCenterQC.BAListBoxMet.Items.Count - 1)

If CustomerCenterQC.BAListBoxMet.Items(iCnt) <> "" Then

mailBody = mailBody & CustomerCenterQC.BAListBoxMet.Items(iCnt) & ": <font color = green>Met</font><br>"

End If

Next iCnt

For iCnt = 0 To (CustomerCenterQC.BAListBoxNotMet.Items.Count - 1)

If CustomerCenterQC.BAListBoxNotMet.Items(iCnt) <> "" Then

mailBody = mailBody & CustomerCenterQC.BAListBoxNotMet.Items(iCnt) & ": <font color = red>Not Met</font><br>"

End If

Next iCnt

mailBody = mailBody & "<br><b>Review:</b> " & CustomerCenterQC.BATextBox.Text & "<br><br>"

mailBody = mailBody & "<b>Customer Enquiry Resolution:</b><br>"

For iCnt = 0 To (CustomerCenterQC.CERListBoxMet.Items.Count - 1)

If CustomerCenterQC.CERListBoxMet.Items(iCnt) <> "" Then

mailBody = mailBody & CustomerCenterQC.CERListBoxMet.Items(iCnt) & ": <font color = green>Met</font><br>"

End If

Next iCnt

For iCnt = 0 To (CustomerCenterQC.CERListBoxNotMet.Items.Count - 1)

If CustomerCenterQC.CERListBoxNotMet.Items(iCnt) <> "" Then

mailBody = mailBody & CustomerCenterQC.CERListBoxNotMet.Items(iCnt) & ": <font color = red>Not Met</font><br>"

End If

Next iCnt

mailBody = mailBody & "<br><b>Review:</b> " & CustomerCenterQC.CERTextBox.Text & "<br><br>"

mailBody = mailBody & "<br><b>Overall Assessment:</b><br>" &

"<b>Policy and Compliance:</b> " & IIf(CustomerCenterQC.PACComboBox.Text = "N/A", "N/A", IIf(CustomerCenterQC.PACComboBox.Text = "Pass", "<font color = green>Pass</font>", "<font color = red>Fail</font>")) & ".<br>" &

"<b>Business Accuracy:</b> " & IIf(CustomerCenterQC.BAComboBox.Text = "N/A", "N/A", IIf(CustomerCenterQC.BAComboBox.Text = "Pass", "<font color = green>Pass</font>", "<font color = red>Fail</font>")) & ".<br>" &

"<b>Customer Enquiry:</b> " & IIf(CustomerCenterQC.CEComboBox.Text = "N/A", "N/A", IIf(CustomerCenterQC.CEComboBox.Text = "Pass", "<font color = green>Pass</font>", "<font color = red>Fail</font>")) & ".<br>" &

"<b>Overall:</b> " & IIf(CustomerCenterQC.OAComboBox.Text = "Pass", "<font color = green>Pass</font>", "<font color = red>Fail</font>") & ".<br><br>" &

"<b>Final Comments:</b> " & CustomerCenterQC.OverallAssessmentTextBox.Text & "<br><br>" &

"<b>NOTE: This is automatically generated email which is unable to be replied to. Please contact your Team Leader/Quality Assessor if you have any concerns.</b><br><br>" &

"Regards,<br>Contact Center Manager.</BODY></HTML>"

recipientTo = Recipents.Add(agentmail)

recipientTo.Type = outlook.OlMailRecipientType.olTo

recipientCC = Recipents.Add(managermail)

recipientCC.Type = outlook.OlMailRecipientType.olCC

OutlookMessage.Subject = "A New Quality Assessment Is Available"

OutlookMessage.Display(OutlookMessage)

OutlookMessage.HTMLBody = mailBody '+ OutlookMessage.HTMLBody

OutlookMessage.BodyFormat = outlook.OlBodyFormat.olFormatHTML

OutlookMessage.Send()

Catch ex As Exception

MsgBox("Mail Not Sent")

Finally

OutlookMessage = Nothing

AppOutlook = Nothing

End Try

'OutlookMessage.Move(objFolder)

Else

Try

mailBody = "<HTML><BODY>Hi " & managername & ",<br><br>" &

"I have completed a Quality Assessment for " & value & " for a call taken on <b>" & dateStr & "</b> at <b>" & timeStr & ".</b><br><br>" &

"This assessment needs to be flagged for your attention for the reasons provided on the Quality sections: <br><br>" &

"<b>Policy and Compliance:</b> <br>"

For iCnt = 0 To (CustomerCenterQC.PandCListBoxMet.Items.Count - 1)

If CustomerCenterQC.PandCListBoxMet.Items(iCnt) <> "" Then

mailBody = mailBody & CustomerCenterQC.PandCListBoxMet.Items(iCnt) & ": <font color = green>Met</font><br>"

End If

Next iCnt

For iCnt = 0 To (CustomerCenterQC.PandCListBoxNotMet.Items.Count - 1)

If CustomerCenterQC.PandCListBoxNotMet.Items(iCnt) <> "" Then

mailBody = mailBody & CustomerCenterQC.PandCListBoxNotMet.Items(iCnt) & ": <font color = red>Not Met</font><br>"

End If

Next iCnt

mailBody = mailBody & "<br><b>Review:</b> " & CustomerCenterQC.PACTextBox.Text & "<br><br>"

mailBody = mailBody & "<b>Business Accuracy:</b><br>"

For iCnt = 0 To (CustomerCenterQC.BAListBoxMet.Items.Count - 1)

If CustomerCenterQC.BAListBoxMet.Items(iCnt) <> "" Then

mailBody = mailBody & CustomerCenterQC.BAListBoxMet.Items(iCnt) & ": <font color = green>Met</font><br>"

End If

Next iCnt

For iCnt = 0 To (CustomerCenterQC.BAListBoxNotMet.Items.Count - 1)

If CustomerCenterQC.BAListBoxNotMet.Items(iCnt) <> "" Then

mailBody = mailBody & CustomerCenterQC.BAListBoxNotMet.Items(iCnt) & ": <font color = red>Not Met</font><br>"

End If

Next iCnt

mailBody = mailBody & "<br><b>Review:</b> " & CustomerCenterQC.BATextBox.Text & "<br><br>"

mailBody = mailBody & "<b>Customer Enquiry Resolution:</b><br>"

'MsgBox(CustomerCenterQC.CERListBoxNotMet.Items.Count)

For iCnt = 0 To (CustomerCenterQC.CERListBoxMet.Items.Count - 1)

If CustomerCenterQC.CERListBoxMet.Items(iCnt) <> "" Then

mailBody = mailBody & CustomerCenterQC.CERListBoxMet.Items(iCnt) & ": <font color = green>Met</font><br>"

End If

Next iCnt

For iCnt = 0 To (CustomerCenterQC.CERListBoxNotMet.Items.Count - 1)

If CustomerCenterQC.CERListBoxNotMet.Items(iCnt) <> "" Then

mailBody = mailBody & CustomerCenterQC.CERListBoxNotMet.Items(iCnt) & ": <font color = red>Not Met</font><br>"

End If

Next iCnt

mailBody = mailBody & "<br><b>Review:</b> " & CustomerCenterQC.CERTextBox.Text & "<br><br>"

mailBody = mailBody & "<br><b>Overall Assessment:</b><br>" &

"<b>Policy and Compliance:</b> " & IIf(CustomerCenterQC.PACComboBox.Text = "N/A", "N/A", IIf(CustomerCenterQC.PACComboBox.Text = "Pass", "<font color = green>Pass</font>", "<font color = red>Fail</font>")) & ".<br>" &

"<b>Business Accuracy:</b> " & IIf(CustomerCenterQC.BAComboBox.Text = "N/A", "N/A", IIf(CustomerCenterQC.BAComboBox.Text = "Pass", "<font color = green>Pass</font>", "<font color = red>Fail</font>")) & ".<br>" &

"<b>Customer Enquiry:</b> " & IIf(CustomerCenterQC.CEComboBox.Text = "N/A", "N/A", IIf(CustomerCenterQC.CEComboBox.Text = "Pass", "<font color = green>Pass</font>", "<font color = red>Fail</font>")) & ".<br>" &

"<b>Overall: </b>" & IIf(CustomerCenterQC.OAComboBox.Text = "Pass", "<font color = green>Pass</font>", "<font color = red>Fail</font>") & ".<br><br>" &

"<b>Final Comments:</b> " & CustomerCenterQC.OverallAssessmentTextBox.Text & "<br><br>" &

"Please let me know if you have any questions.<br><br>" &

"Regards,<br>Contact Center Manager.</BODY></HTML>"

recipientTo = Recipents.Add(managermail)

recipientTo.Type = outlook.OlMailRecipientType.olTo

recipientCC = Recipents.Add(ccmanagermail)

recipientCC.Type = outlook.OlMailRecipientType.olCC

OutlookMessage.Subject = "Quality Assessment Notification"

OutlookMessage.Display(OutlookMessage)

OutlookMessage.HTMLBody = mailBody '+ OutlookMessage.HTMLBody

OutlookMessage.BodyFormat = outlook.OlBodyFormat.olFormatHTML

OutlookMessage.Send()

'On Error GoTo 0

Catch ex As Exception

MsgBox("Mail Not Sent")

Finally

OutlookMessage = Nothing

AppOutlook = Nothing

End Try

End If

'Catch ex As Exception

'MessageBox.Show("Mail could not be sent") 'if you dont want this message, simply delete this line

'Finally

'End Try

Done:

'Exit Sub

'ErrorBlock:

' DisplayError(Err.Source, Err.Description, "PhoneSelector.LinkedCustIDs\_Load", Erl)

' Resume Done

End Sub

End Module