

## Table of Contents

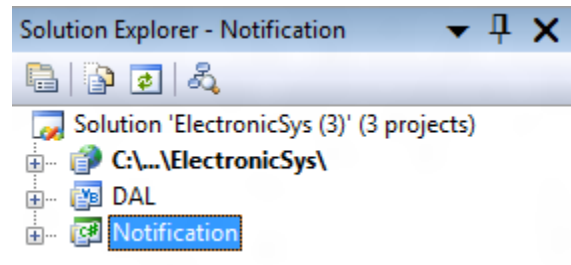
Introduction.....	1
Master Page.....	2
Security.....	4
Presentation Layer.....	7
Application.....	7
Administration.....	10
Training Offer Operation .....	10
Courses Attendance Result.....	12
Training Programs Operations.....	12
Sending Filtration Results Via Mobile.....	13
Differentiation .....	14
Main Reference Code .....	17
Update Application.....	20
Filtration Operations .....	20
Trainee’s nomination.....	21
Reports .....	24
App_Code Classes.....	26
Data Access Layer .....	27
Database.....	28

## Introduction

ERS Solution is consisting of 3 Projects ([Visual Studio.Net 2008 – FW: 3.5 SP1](#))

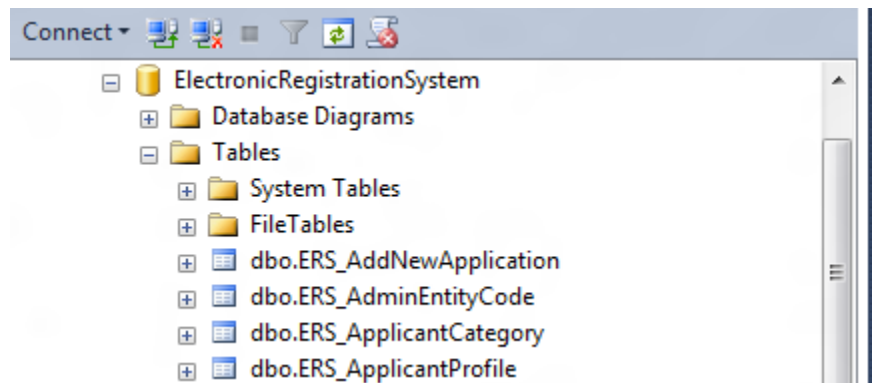
- **Electronic Sys ASP.NET website “front end web site + backend Administration”**
- **DAL Class Library “Data Access Layer”**
- **Notification Class Library “SMS Service”**

Using [SQL Server Database 2005](#) all functionality



## Resident in Stored Procedures.

Integration between Application and Database are going into Connected Model: Direct SQL Stored Procedures Calling from Dot NET.



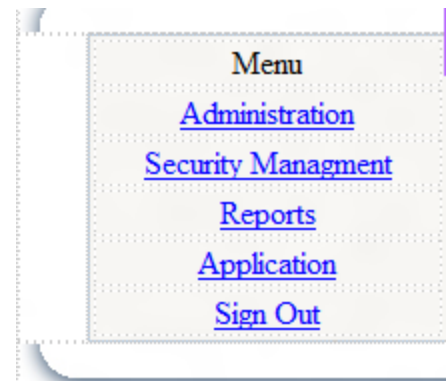
## Master Page

Using 2 Master pages with CSS File

and layout images resident in IMG Folder are composing website layout

- **Main.master** : for login & Registration screens
- **Intro.Master**: for Application Parts; Same as Main.Master but Adds Menu Functionality, a collapsible menu is a custom HTML + CSS Menu + Javascript, using cookies to save the last user location.

```
<tr>
- <td class="rowValue">
- <asp:HyperLink ID="HyperLink1" runat="server"
  CssClass="NormaiLink"
  NavigateUrl="~/Administator/Default.aspx"
- Text="<%%$ Resources:Menu, Administrator
  %>"></asp:HyperLink>
- </td>
- </tr>
- <tr>
- <td class="rowValue">
- <asp:HyperLink ID="HyperLink79" runat="server"
  CssClass="NormaiLink"
- NavigateUrl="~/SAM/Default.aspx"
- Text="<%%$ Resources:Menu, SAM %>"></asp:HyperLink>
- </td>
- </tr>
```



Each one of those links are redirecting to a specific part of the application which found in a folder containing its own master page and web forms.

- Administrator

Admin.Master

```
<tr>
```

```

<td class="rowValue" onclick="javascript:showLayer('Menu1','ImgMenu1')"
style="cursor:hand">
</td>
<td class="rowValue">
<asp:Literal ID="Literal1" runat="server" Text="<%%$ Resources:Menu,
TrainingOfferOperation%>" /></td>
</tr>
<tr><td colspan="2"><div id="Menu1" style="visibility: hidden; position: absolute">
<table>
<tr>
<td class="rowValue">
</td>
<td class="rowValue">
<asp:HyperLink ID="HyperLink11" runat="server" CssClass="NormaiLink"
NavigateUrl="~/Administator/FrmAddNewOffer.aspx"
Text="<%%$ Resources:Menu, Addnewofferoperation %>"></asp:HyperLink>
</td>
</tr>
<tr>
<td class="rowValue">
</td>
<td class="rowValue">
<asp:HyperLink ID="HyperLink12" runat="server" CssClass="NormaiLink"
NavigateUrl="~/Administator/FrmAddCourse.aspx"
Text="<%%$ Resources:Menu, Courses %>"></asp:HyperLink>
</td>
</tr>

```

```

function showLayer(layerName,imgCtl) {

if (layerName == '' && imgCtl == '') {

if (getCookieByName("Layer")) {
layerName = getCookieByName("Layer");
imgCtl = "Img" + layerName;
}

}

if (layerName != '') {
if (eval("document.getElementById('" + layerName + "').style.visibility") ==
'visible') {
eval("document.getElementById('" + layerName + "').style.visibility = 'hidden'");
eval("document.getElementById('" + layerName + "').style.position = 'absolute'");
eval("document.getElementById('" + imgCtl + "').src = '../img/folder.gif'");
document.cookie = "Layer=";
} else {

eval("document.getElementById('" + layerName + "').style.visibility = 'visible'");
eval("document.getElementById('" + layerName + "').style.position = 'static'");
eval("document.getElementById('" + imgCtl + "').src = '../img/folder_open.gif'");
var oldLayer = getCookieByName("Layer")
if (oldLayer && oldLayer != layerName) {
eval("document.getElementById('" + oldLayer + "').style.visibility = 'hidden'");

```

```

eval("document.getElementById('" + oldLayer + "').style.position = 'absolute'");
eval("document.getElementById('Img' + oldLayer + "').src = '../img/folder.gif'");
}
document.cookie = "Layer=" + layerName;

}
}
}
function getCookieByName(name) {
var results = document.cookie.match('(' + name + '=([^;]*)' + ';$)');
if (results)
return (unescape(results[2]));
else
return null;
}

```

## Security

Security Is using ASP.NET Forms authentication as the following

- Web.Config: we disallow any non-logged in user to the system

```

<authentication mode="Forms">
  <forms loginUrl="Default.aspx"/>
</authentication>
<authorization>
  <deny users="?" />
</authorization>

```
- We Excluding Registration Folder from the previous Rule

```

<location path="Registration">
  <system.web>
    <authorization>
      <allow users="*" />
    </authorization>
  </system.web>
</location>

```
- Registration Process
  - a- For Non KSU Users we use the form RegisterNew.aspx to add New account to our Database through Applicant Calss in DAL Project
  - b- For KSU Users we are using Register.aspx to copy user profile from SSO Service and register the new user.
- LogIn Process
  - a- Default.aspx: if user found we check his groupNo if 1 then he is administrator so we redirect him to admin page else he is

[Literal]	
	<input type="button" value="Add New"/> <input type="button" value="Update"/> <input type="button" value="Delete"/> <input type="button" value="Save"/> <input type="button" value="Cancel"/> <input type="button" value="Search"/>
Three Name	<input type="text"/> [bId]
Login	<input type="text"/> *
Password	<input type="password"/>
Group Permission	Unbound ▾
Email	<input type="text"/> **
Mobile	<input type="text"/> *
Office No	<input type="text"/>
Fax No	<input type="text"/>
<a href="#">Home Page</a> <a href="#">Back</a>	

applicant so we redirect him to applicant page

- SAM Folder contains pages to admin users and groups as well as operation

a- FrmAddUpdateUser.aspx :

```
Protected Sub BtnSearch_Click(ByVal sender As Object, ByVal e As
System.EventArgs) Handles BtnSearch.Click
```

```
Dim found As Boolean = False
```

```
Try
```

```
ObjDs = AllCodes.GetAllTableDataById("ERS_UserRegistration",
"UserName", txtLogin.Text)
If ObjDs Is Nothing Then Throw New Exception
If ObjDs.Tables(0).Rows.Count > 0 Then
    txtEmail.Text = ObjDs.Tables(0).Rows(0) ("Email").ToString()
    txtFaxNo.Text = ObjDs.Tables(0).Rows(0) ("FaxNo").ToString()
    txtLogin.Text = ObjDs.Tables(0).Rows(0) ("UserName").ToString()
    txtPassword.Text = ObjDs.Tables(0).Rows(0) ("Password").ToString()
    txtMobile.Text = ObjDs.Tables(0).Rows(0) ("MobileNo").ToString()
    txtName.Text = ObjDs.Tables(0).Rows(0) ("FullName").ToString()
    txtOfficeNo.Text = ObjDs.Tables(0).Rows(0) ("OfficeNo").ToString()
    lblId.Text = ObjDs.Tables(0).Rows(0) ("ID").ToString()
    ddGroups.SelectedValue =
ObjDs.Tables(0).Rows(0) ("GroupNo").ToString()
    found = True
Else
    ScriptManager.RegisterStartupScript(Me, Me.GetType(), "msg",
>alert("'" & CType(GetGlobalResourceObject("Messages", "NoMatchFound"), String) &
"')", True)
End If
Catch ex As Exception
    CommonFunction.AddErrorLog(Request.CurrentExecutionFilePath & "
BtnSearch_Click Event" & ": " & ex.Message.ToString())
End Try
```

b- Form FrmGroupManagement.aspx : Used to Add/Edit Groups

c- Form FrmUserManagement.aspx : Used to view group Users / Add Existing User to group.

d- Form FrmOperationManagement.aspx : Used to add/edit Operations to system to link system jobs with pages links

e- Form FrmGroupRightManagement.aspx : Assigning Operations to each group with permissions ( View Only – Read/Write.

f- We use Enum to expose Account Access Security : Found in DAL project, Class LoginInfo.vb

```
Public Enum ACL
    None = 0
    ReadOnlyMode = 1
    ReadWriteMode = 2
End Enum
```

g- To retrieve User Permissions we Create a base Class and implement security functions in page\_Load and inherit all application pages from it so on Every Page load the Public EntryMode As ACL variable will be accessed from each page and decide permissions  
PageClass.vb

```
Imports Microsoft.VisualBasic
Imports System.Globalization
Imports DAL
Public Class PageClass
    Inherits System.Web.UI.Page
    Public EntryMode As ACL

    Protected Overrides Sub InitializeCulture()
        Dim selectedLanguage As String = Session("LangSelection").ToString()

        If Not String.IsNullOrEmpty(selectedLanguage) Then
            selectedLanguage = selectedLanguage
        Else
            selectedLanguage = "en-US"
        End If

        Threading.Thread.CurrentThread.CurrentCulture = _
            New GenericCultureUmAlQuraCalendar(selectedLanguage)
        Threading.Thread.CurrentThread.CurrentUICulture = _
            CultureInfo.CreateSpecificCulture(selectedLanguage)
        MyBase.InitializeCulture()
    End Sub

    Protected Sub Page_Load(ByVal sender As Object, ByVal e As System.EventArgs)
Handles Me.Load
        EntryMode = GetUserACR()
        If User.Identity.IsAuthenticated And Not
Page.AppRelativeVirtualPath.EndsWith("Main1.aspx") Then
            If Not SessionClass.IsAdmin And CheckPageList() Then
                If EntryMode = ACL.None Then
                    Response.Redirect("~/main1.aspx")
                End If
            Else
                EntryMode = ACL.ReadWriteMode
            End If
        End If
    End Sub

    Private Function GetUserACR() As DAL.ACL
        Dim LgnInfo As New LoginInfo()
        Return LgnInfo.GetUserACL(SessionClass.UserID,
Me.Page.AppRelativeVirtualPath.Replace("\", "/"))
    End Function

    Private Function CheckPageList() As Boolean
        Dim securepage As Boolean = False
        Dim sourceDirectory As String
        sourceDirectory = Me.Page.AppRelativeTemplateSourceDirectory
        If Not SessionClass.IsApplicant And
(Me.Page.AppRelativeVirtualPath.EndsWith("Default.aspx") Or
Page.AppRelativeVirtualPath.EndsWith("/")) Then
            Return False
        End If
        If sourceDirectory = "~//" Or sourceDirectory = "~/Application/" Or
sourceDirectory = "~/Registration/" Then
            securepage = False
        Else
            securepage = True
        End If
    End Function
```

```

End If
Return securepage
End Function
End Class

```

## Presentation Layer

### Application

#### - AvailableOffers.aspx : Main Functions

```
Dim ClsOffer As New OfferInfo
```

```
Dim objds As DataSet
```

On Page Load we call the following Function

```

1- Private Sub GetAvailableOffer()
    objds = ClsOffer.GetAllAvailableOffer()
    If objds Is Nothing Then Throw New Exception
    If objds.Tables(0).Rows.Count > 0 Then
        GvAvailableOffer.DataSource =
objds.Tables(0)
        GvAvailableOffer.DataBind()
        GvAvailableOffer.Visible = True
        lblErrorMsg.Visible = False
        BtnAddNewRequest.Enabled = True
    Else
        GvAvailableOffer.Visible = False
        BtnAddNewRequest.Enabled = False
        lblErrorMsg.Visible = True
        ScriptManager.RegisterStartupScript(Me,
Me.GetType(), "msg", "alert('" & CType(GetGlobalResourceObject("Messages",
"NoMatchFound"), String) & "')" , True)

    End If

```

```

2- Protected Sub GvAvailableOffer_SelectedIndexChanged(ByVal sender As
Object, ByVal e As System.EventArgs)

```

Get Selected offer to populate its information on screen

#### - UpdateRequest.aspx Main Functions

```
Dim ClsApp As New ApplicationInfo
```

```
Dim objds As DataSet
```

```
Dim ClsCom As New CommonControlFunction
```

```
Dim ClsCourse As New CourseInfo
```

#### 1- On Page load we call

```

GetApplicationByUser() \ Get Applied applications for the current User
TraineeClass() \ Bind Applicant Category to Drop Down List

```

```

2- Protected Sub GvApplication_SelectedIndexChanged(ByVal sender As Object,
ByVal e As System.EventArgs) Handles GvApplication.SelectedIndexChanged
    \ Select Application to bind Applied courses to courses grid

```



```

- DeleteRequest.aspx Main Functions
1- Protected Sub Page_Load(ByVal sender As Object, ByVal e As
    System.EventArgs) Handles Me.Load ' Get All Applications for the current
    user
2- Protected Sub GvApplication_RowCommand(ByVal sender As Object, ByVal e As
    System.Web.UI.WebControls.GridViewCommandEventArgs) ' If user click on
    Delete we Delete the current request if updatable or deleteable

- UpdateCourse.aspx Main Functions
1- Protected Sub Page_Load(ByVal sender As Object, ByVal e As
    System.EventArgs) ' Get Application For the current user
2- Protected Sub GvApplication_SelectedIndexChanged(ByVal sender As Object,
    ByVal e As System.EventArgs) ' Publish Courses for the selected
    Application
3- Protected Sub GvAvailableCourse_RowCommand(ByVal sender As Object, ByVal e
    As System.Web.UI.WebControls.GridViewCommandEventArgs) ' Update Selected
    Course

- DeleteCourse.aspx Main Functions
1- Protected Sub Page_Load(ByVal sender As Object, ByVal e As
    System.EventArgs) ' Get Application For the current user
2- Protected Sub GvApplication_SelectedIndexChanged(ByVal sender As Object,
    ByVal e As System.EventArgs) ' Publish Courses for the selected
    Application
3- Protected Sub GvAvailableCourse_RowCommand(ByVal sender As Object, ByVal e
    As System.Web.UI.WebControls.GridViewCommandEventArgs) ' Delete Selected
    Course

- ListofPardonCourse.aspx Main Functions
1- Protected Sub Page_Load(ByVal sender As Object, ByVal e As
    System.EventArgs) ' Get Registered Courses Availible to pardon
2- Protected Sub GvApplication_RowCommand(ByVal sender As Object, ByVal e As
    System.Web.UI.WebControls.GridViewCommandEventArgs) ' Update Applicant
    Course Status to be pardoned

- FrmUpdateProfile.aspx main Functions
1- Protected Sub Page_Load(ByVal sender As Object, ByVal e As System.EventArgs)
    Handles Me.Load
    If Not IsPostBack Then
        Try
            ClsCom.BindCodestoDropDown("ERS_MainEntityCode", ddAdvViewHome)
            ClsCom.BindToDropDown("ERS_IdentityType", ddIdentityType)
            ClsCom.BindCodestoDropDown("ERS_NationalityCode", ddNationality)
            ClsCom.SortList(ddNationality)
            ClsCom.SortList(ddAdvViewHome)
            ClsCom.SortList(ddAdvViewHome)
            objds = AllCodes.GetAllTableDataById("ERS_UserRegistration", "id",
SessionClass.UserID)
            If Not objds Is Nothing Then
                If objds.Tables.Count > 0 Then
                    If objds.Tables(0).Rows.Count > 0 Then

```



```

txtEmail.Text =
objds.Tables(0).Rows(0) ("Email").ToString()
txtFirstName.Text =
objds.Tables(0).Rows(0) ("firstName").ToString().Trim()
txtMidName.Text =
objds.Tables(0).Rows(0) ("MiddleName").ToString().Trim()
txtLastName.Text =
objds.Tables(0).Rows(0) ("FamilyName").ToString().Trim()
txtMobile.Text =
objds.Tables(0).Rows(0) ("MobileNo").ToString()
txttel.Text =
objds.Tables(0).Rows(0) ("TelephoneNo").ToString()
txtIDNo.Text =
objds.Tables(0).Rows(0) ("IdentityNo").ToString()
txtUsername.Text =
objds.Tables(0).Rows(0) ("Username").ToString()
ddAdvViewHome.SelectedValue =
objds.Tables(0).Rows(0) ("AdvancedviewHome").ToString()
ddIdentityType.SelectedValue =
objds.Tables(0).Rows(0) ("IdentityType").ToString()
ddNationality.SelectedValue =
objds.Tables(0).Rows(0) ("Nationality").ToString()
RbutProfession.SelectedValue =
objds.Tables(0).Rows(0) ("Profession").ToString()
End If
End If
End If
Catch ex As Exception
End Try
End If
End Sub

```

```
2- Protected Sub BtnSave_Click(ByVal sender As Object, ByVal e As
    System.EventArgs) Handles BtnSave.Click
```

```

Try
    Dim IsInserted As Boolean = False
    Dim SourceRegistration As String = "apl"
    Dim password As String = AllCodes.GetColumnData("Password",
"ERS_UserRegistration", "id", SessionClass.UserID)
    Dim Gender As String = AllCodes.GetColumnData("Gender",
"ERS_UserRegistration", "id", SessionClass.UserID)
    Dim Status As String = AllCodes.GetColumnData("Status",
"ERS_UserRegistration", "id", SessionClass.UserID)
    IsInserted = clsApplicant.AddNewApplicant(txtFirstName.Text & " ",
txtMidName.Text & " ", txtLastName.Text, _
txtMobile.Text, txttel.Text,
_
txtUsername.Text.ToLower(),
password, ddIdentityType.SelectedValue, _
txtIDNo.Text, Gender,
ddAdvViewHome.SelectedValue, _
Status,
RbuthProfession.SelectedValue, SourceRegistration, "", "", txtEmail.Text.ToLower(),
ddNationality.SelectedValue)

```

```

        If IsInserted = True Then

            ScriptManager.RegisterStartupScript(Me, Me.GetType(), "msg",
"alert('" & CType(GetGlobalResourceObject("Messages", "SaveSuccess"), String) &
"')", True)

        Else

            ScriptManager.RegisterStartupScript(Me, Me.GetType(), "msg",
"alert('" & CType(GetGlobalResourceObject("Messages", "AlreadyExist"), String) &
"')", True)

        End If
    Catch ex As Exception

    End Try
End Sub

```

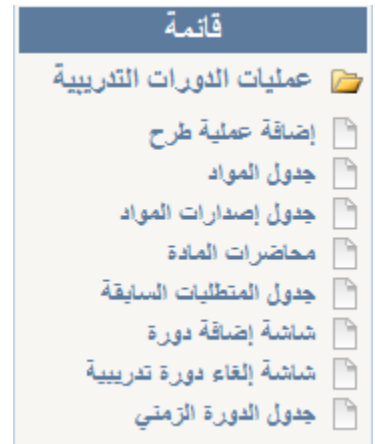
#### - CourseCompleted.aspx Main Functions

- 1- Protected Sub Page\_Load(ByVal sender As Object, ByVal e As System.EventArgs) \ Get Finished Courses For Applicant To Evaluate
- 2- Protected Sub GvApplication\_RowCommand(ByVal sender As Object, ByVal e As System.Web.UI.WebControls.GridViewCommandEventArgs) \ Show Course Information
- 3- Protected Sub BtnCourseEvaluation\_Click(ByVal sender As Object, ByVal e As System.EventArgs) \ Start Course Evaluation Process

## Administration

### Training Offer Operation

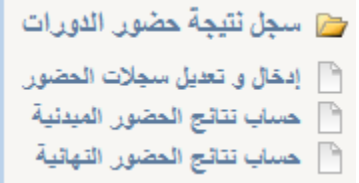
- FrmAddNewOffer.aspx
  - 1- Protected Sub Page\_Load(ByVal sender As Object, ByVal e As System.EventArgs) \ Prepare search lists and controls
  - 2- Protected Sub BtnSearch\_Click(ByVal sender As Object, ByVal e As System.EventArgs) \ Search Offers
  - 3- Protected Sub BtnSaveOffer\_Click(ByVal sender As Object, ByVal e As System.EventArgs) \ Save New Offer as well as updating existing/Selected offer
- FrmAddCourse.aspx main Functions
  - 1- On Load Publish All Subjects
  - 2- Protected Sub BtnSearch\_Click(ByVal sender As Object, ByVal e As System.EventArgs) \ Search Subject
  - 3- Protected Sub BtnSave\_Click(ByVal sender As Object, ByVal e As System.EventArgs) \ Save / Update Subject
- FrmSubjectVersion.aspx
  - 1- Protected Sub Page\_Load(ByVal sender As Object, ByVal e As System.EventArgs) \ publish versions



- 2- `Protected Sub BtnSearch_Click(ByVal sender As Object, ByVal e As System.EventArgs) \ Search versions`
  - 3- `Protected Sub GvSubVersion_RowCommand(ByVal sender As Object, ByVal e As System.Web.UI.WebControls.GridViewCommandEventArgs) \ Delete Versions`
  - 4- `Protected Sub BtnSave_Click(ByVal sender As Object, ByVal e As System.EventArgs) \ Save New Version / Update Selected version`
- **FrmSubjectLecture.aspx**
- 1- `Protected Sub Page_Load(ByVal sender As Object, ByVal e As System.EventArgs) \ publish lectures`
  - 2- `Protected Sub BtnSearch_Click(ByVal sender As Object, ByVal e As System.EventArgs) \ Search lectures`
  - 3- `Protected Sub GvSubLecture_RowCommand(ByVal sender As Object, ByVal e As System.Web.UI.WebControls.GridViewCommandEventArgs) \ Delete lectures`
  - 4- `Protected Sub BtnSave_Click(ByVal sender As Object, ByVal e As System.EventArgs) \ Save New lectures/ Update Selected lecture`
- **FrmSubjectPrerequisite.aspx**
- 1- `Protected Sub Page_Load(ByVal sender As Object, ByVal e As System.EventArgs) \ publish Prerequisites`
  - 2- `Protected Sub BtnSearch_Click(ByVal sender As Object, ByVal e As System.EventArgs) \ Search Prerequisites`
  - 3- `Protected Sub GvSubPrerequisite_RowCommand(ByVal sender As Object, ByVal e As System.Web.UI.WebControls.GridViewCommandEventArgs) \ Delete Prerequisites`
  - 4- `Protected Sub BtnSave_Click(ByVal sender As Object, ByVal e As System.EventArgs) \ Save New Prerequisite/ Update Selected Prerequisite`
- **FrmNewTrainingCourse.aspx**
- 1- `Protected Sub Page_Load(ByVal sender As Object, ByVal e As System.EventArgs) \ Bind Course List and search lists`
  - 2- `Protected Sub BtnSearch_Click(ByVal sender As Object, ByVal e As System.EventArgs) \ Search Courses`
  - 3- `Protected Sub BtnSave_Click(ByVal sender As Object, ByVal e As System.EventArgs) \ Add/Edit training Course`
- **FrmCancelTrainingCourse.aspx**
- 1- Just to change course status to be canceled , Enter Cancellation Date
- **FrmLectureTimeSchedule.aspx**
- 1- `Protected Sub Page_Load(ByVal sender As Object, ByVal e As System.EventArgs) \ Bind Time schedules to grid`
  - 2- `Protected Sub BtnSearch_Click(ByVal sender As Object, ByVal e As System.EventArgs) \ Search time schedules by course`
  - 3- `Protected Sub BtnSave_Click(ByVal sender As Object, ByVal e As System.EventArgs) \ Save / Update lectures time schedules.`

## Courses Attendance Result

- Course attendance record - FrmCourseAttendanceRecord.aspx
  1. `Protected Sub Page_Load(ByVal sender As Object, ByVal e As System.EventArgs) 'publish the courses list from the database.`
  2. `Protected Sub ddCourseNo_SelectedIndexChanged(ByVal sender As Object, ByVal e As System.EventArgs) 'get the details of the selected course`
  3. `Protected Sub BtnSave_Click(ByVal sender As Object, ByVal e As System.EventArgs) ' save the information of the attendance and the absence`
  4. `Protected Sub BtnOpenAttRec_Click(ByVal sender As Object, ByVal e As System.EventArgs)' show the infromation of the attendees`
- Calculation of preliminary course attendance - FrmCalcPreCourseAttendance.aspx
  1. `Protected Sub Page_Load(ByVal sender As Object, ByVal e As System.EventArgs)' publish the dropdown list with the courses codes.`
  2. `Protected Sub ddCourseNo_SelectedIndexChanged(ByVal sender As Object, ByVal e As System.EventArgs) ' get the details of the selected course`
  3. `Protected Sub BtnOpenAttRec_Click(ByVal sender As Object, ByVal e As System.EventArgs) ' shows the results of the attendance.`
- Calculation of final course attendance records - FrmCalcFinalAttendance.aspx
  1. `Protected Sub Page_Load(ByVal sender As Object, ByVal e As System.EventArgs) 'publish the dropdown list with the courses codes.`
  2. `Protected Sub ddCourseNo_SelectedIndexChanged(ByVal sender As Object, ByVal e As System.EventArgs) ' get the details of the selected course`
  3. `Protected Sub BtnFinalAttendance_Click(ByVal sender As Object, ByVal e As System.EventArgs) ' shows the results of the attendance.`



## Training Programs Operations

- Add a training program - FrmAddTrainingProgram.aspx
  - 1- `Protected Sub Page_Load(ByVal sender As Object, ByVal e As System.EventArgs) ' Bind Offers`
  - 2- `Protected Sub GvAllOffer_SelectedIndexChanged(ByVal sender As Object, ByVal e As System.EventArgs) ' Publish Offer Courses`
  - 3- `Protected Sub BtnSave_Click(ByVal sender As Object, ByVal e As System.EventArgs) ' Add/Edit Training program based on selected courses`
- Update course exam results - FrmUpdateCourseExamresult.aspx
  - 1- `Protected Sub Page_Load(ByVal sender As Object, ByVal e As System.EventArgs) ' bind courses`
  - 2- `Protected Sub BtnSearch_Click(ByVal sender As Object, ByVal e As System.EventArgs) ' get Course Attendees`



- 3- `Protected Sub GvAttendees_RowCommand(ByVal sender As Object, ByVal e As System.Web.UI.WebControls.GridViewCommandEventArgs) \ Update selected attendee exam result`
- Update records of applicants to courses - FrmUpdaterecordsCourses.aspx
  - 1- `Protected Sub Page_Load(ByVal sender As Object, ByVal e As System.EventArgs) \ get closed courses list`
  - 2- `Protected Sub ddCourseCode_SelectedIndexChanged(ByVal sender As Object, ByVal e As System.EventArgs) \ publish course info, get registered applicants`
  - 3- `Protected Sub BtnSave_Click(ByVal sender As Object, ByVal e As System.EventArgs) \ update exemption status`
- Manual registration form - FrmManualRegistration.aspx
  - 1- `Protected Sub BtnSearch_Click(ByVal sender As Object, ByVal e As System.EventArgs) \ Search course and publish its data`
  - 2- `Protected Sub BtnSave_Click(ByVal sender As Object, ByVal e As System.EventArgs) \ Register Entered Applicant to searched course`
- Training rooms profile - FrmTrainingRoomsProfile.aspx
  - 1- `Protected Sub BtnSave_Click(ByVal sender As Object, ByVal e As System.EventArgs) \ Save/Update Training room information`
- Certificates printing - FrmPrintCertificate1.aspx
  - 1- `Protected Sub BtnPrint_Click(ByVal sender As Object, ByVal e As System.EventArgs) \ Prepare predesigned report for certificate, pass parameters then Print.`
- Issuing certificates - FrmIssueCertificate.aspx
  - 1- `Protected Sub Page_Load(ByVal sender As Object, ByVal e As System.EventArgs) \ Publish course info based on dropdownlist selection`
  - 2- `Protected Sub BtnPrint_Click(ByVal sender As Object, ByVal e As System.EventArgs) \ Issue certificate by date`

## Sending Filtration Results Via Mobile

- Sending filtration results via mobile text message - FrmFilterResultViaMobile.aspx
  - 1- `Protected Sub Page_Load(ByVal sender As Object, ByVal e As System.EventArgs) \ Publish Filter Operations`
  - 2- `Protected Sub ddFilterNo_SelectedIndexChanged(ByVal sender As Object, ByVal e As System.EventArgs) \ Publish Filter info`



- 3- `Protected Sub BtnSend_Click(ByVal sender As Object, ByVal e As System.EventArgs) \ Send Message to applicants included in filter to inform them about result.`
- Send a mobile text message to a user - `FrmSendMobileTextUser.aspx`
    - 1- `Protected Sub BtnSearch_Click(ByVal sender As Object, ByVal e As System.EventArgs) \ Search Applicant by Id to send message`
    - 2- `Protected Sub BtnSend_Click(ByVal sender As Object, ByVal e As System.EventArgs) \ Send mobile sms to selected user and entered message body.`
  - Send a mobile text message to a group of applicants - `FrmSendMessageToGroup.aspx`
    - 1- `Protected Sub Page_Load(ByVal sender As Object, ByVal e As System.EventArgs) \ Bind Applicants List`
    - 2- `Protected Sub BtnAddApplicant_Click(ByVal sender As Object, ByVal e As System.EventArgs) \ Add selected applicant to message recipient list.`
    - 3- `Protected Sub BtnSend_Click(ByVal sender As Object, ByVal e As System.EventArgs) \ Send Message to selected applicants.`
  - Send email message to registered users in course - `FrmSendEmailToRegisteredUser.aspx`
    - 1- `Protected Sub ddFilterNo_SelectedIndexChanged(ByVal sender As Object, ByVal e As System.EventArgs) \ select filter operation to get courses and registered Applicants`
    - 2- `Protected Sub BtnSend_Click(ByVal sender As Object, ByVal e As System.EventArgs) \ Send Message to registered applicants`

## Differentiation

- Types of differentiations - `FrmTypeofDifferentiation.aspx`
  - 1- `Protected Sub Page_Load(ByVal sender As Object, ByVal e As System.EventArgs) \ Bind differentiation types to grid`
  - 2- `Protected Sub GvTypeOfDiff_RowCommand(ByVal sender As Object, ByVal e As System.Web.UI.WebControls.GridViewCommandEventArgs) \ Delete selected row`
  - 3- `Protected Sub GvTypeOfDiff_SelectedIndexChanged(ByVal sender As Object, ByVal e As System.EventArgs) \ publish selected diff type fields to update`
  - 4- `Protected Sub BtnSave_Click(ByVal sender As Object, ByVal e As System.EventArgs) \ Update \ Add differentiation type`



- Details of the differentiation - FrmDetailDifferentiation.aspx
  - 1- `Protected Sub Page_Load(ByVal sender As Object, ByVal e As System.EventArgs) \ Bind diffrentiaition type list`
  - 2- `Protected Sub ddDiffType_SelectedIndexChanged(ByVal sender As Object, ByVal e As System.EventArgs) \ Bind diffrentiation types details to grid`
  - 3- `Protected Sub GvdiffDetail_RowCommand(ByVal sender As Object, ByVal e As System.Web.UI.WebControls.GridViewCommandEventArgs) \ Delete selected detail`
  - 4- `Protected Sub GvdiffDetail_SelectedIndexChanged(ByVal sender As Object, ByVal e As System.EventArgs) \ publish selected detail fields to update`
  - 5- `Protected Sub BtnSave_Click(ByVal sender As Object, ByVal e As System.EventArgs) \ Add/ edit detail.`
- Members of the differentiation - FrmMemberofDifferentiation.aspx
  - 1- `Protected Sub Page_Load(ByVal sender As Object, ByVal e As System.EventArgs) \ Bind Member of Differentiation list`
  - 2- `Protected Sub GVdiffMember_SelectedIndexChanged(ByVal sender As Object, ByVal e As System.EventArgs) \ Bind member of diffrentiation to grid`
  - 3- `Protected Sub GVdiffMember_RowCommand(ByVal sender As Object, ByVal e As System.Web.UI.WebControls.GridViewCommandEventArgs) \ Delete selected member`
  - 4- `Protected Sub BtnSave_Click(ByVal sender As Object, ByVal e As System.EventArgs) \ Add/ edit detail.`
- Rank and position differentiation member - FrmGroupofDifferentiationMemeber.aspx
  - 1- `Protected Sub Page_Load(ByVal sender As Object, ByVal e As System.EventArgs) \ Bind Differentiation Member list`
  - 2- `Protected Sub GVRankMember_SelectedIndexChanged(ByVal sender As Object, ByVal e As System.EventArgs) \ Bind member of diffrentiation to grid`
  - 3- `Protected Sub GVRankMember_RowCommand(ByVal sender As Object, ByVal e As System.Web.UI.WebControls.GridViewCommandEventArgs) \ Delete selected member`
  - 4- `Protected Sub BtnSave_Click(ByVal sender As Object, ByVal e As System.EventArgs) \ Add/ edit detail.`
- Non rank and position differentiation member - FrmNonrankandposition.aspx
  - 1- `Protected Sub Page_Load(ByVal sender As Object, ByVal e As System.EventArgs) \ Bind Non rank and position Member list`
  - 2- `Protected Sub GVRankMember_SelectedIndexChanged(ByVal sender As Object, ByVal e As System.EventArgs) \ Bind Non rank and position to grid`
  - 3- `Protected Sub GVRankMember_RowCommand(ByVal sender As Object, ByVal e As System.Web.UI.WebControls.GridViewCommandEventArgs) \ Delete selected Non rank and position`

- 4- `Protected Sub BtnSave_Click(ByVal sender As Object, ByVal e As System.EventArgs) \ Add/ edit detail.`
- Types of conditions - FrmTypeOfCondition.aspx
  - 1- `Protected Sub Page_Load(ByVal sender As Object, ByVal e As System.EventArgs) \ Bind Type of Condition list`
  - 2- `Protected Sub GVRankMember_SelectedIndexChanged(ByVal sender As Object, ByVal e As System.EventArgs) \ Bind Type of Condition to controls`
  - 3- `Protected Sub GVRankMember_RowCommand(ByVal sender As Object, ByVal e As System.Web.UI.WebControls.GridViewCommandEventArgs) \ Delete Type of Condition`
  - 4- `Protected Sub BtnSave_Click(ByVal sender As Object, ByVal e As System.EventArgs) \ Add/ edit detail.`
- Condition groups - FrmConditionsGroups.aspx
  - 1- `Protected Sub Page_Load(ByVal sender As Object, ByVal e As System.EventArgs) \ Bind groups list`
  - 2- `Protected Sub GVRankMember_SelectedIndexChanged(ByVal sender As Object, ByVal e As System.EventArgs) \ Bind group to controls`
  - 3- `Protected Sub GVRankMember_RowCommand(ByVal sender As Object, ByVal e As System.Web.UI.WebControls.GridViewCommandEventArgs) \ Delete condition group`
  - 4- `Protected Sub BtnSave_Click(ByVal sender As Object, ByVal e As System.EventArgs) \ Add/ edit detail.`
- Details of conditions groups - FrmDetailofConditionGroup.aspx
  - 1- `Protected Sub Page_Load(ByVal sender As Object, ByVal e As System.EventArgs) \ Bind groups list`
  - 2- `Protected Sub GVRankMember_SelectedIndexChanged(ByVal sender As Object, ByVal e As System.EventArgs) \ Bind group to controls`
  - 3- `Protected Sub GVRankMember_RowCommand(ByVal sender As Object, ByVal e As System.Web.UI.WebControls.GridViewCommandEventArgs) \ Delete Detail of condition group`
  - 4- `Protected Sub BtnSave_Click(ByVal sender As Object, ByVal e As System.EventArgs) \ Add/ edit detail.`
- Group of Differentiation Elements - FrmGroupofDifferentiationElements.aspx
  - 1- `Protected Sub Page_Load(ByVal sender As Object, ByVal e As System.EventArgs) \ Bind groups list`
  - 2- `Protected Sub GVRankMember_SelectedIndexChanged(ByVal sender As Object, ByVal e As System.EventArgs) \ Bind group to controls`
  - 3- `Protected Sub GVRankMember_RowCommand(ByVal sender As Object, ByVal e As System.Web.UI.WebControls.GridViewCommandEventArgs) \ Delete group of differentiation element.`



- 4- `Protected Sub BtnSave_Click(ByVal sender As Object, ByVal e As System.EventArgs) \ Add/ edit detail.`

## Main Reference Code

- Ranks codes - FrmRankCodes.aspx
  1. `Protected Sub Page_Load(ByVal sender As Object, ByVal e As System.EventArgs) ' get the data of ranks`
  2. `Protected Sub BtnAddNew_Click(ByVal sender As Object, ByVal e As System.EventArgs) ' add new rank`
  3. `Protected Sub BtnSave_Click(ByVal sender As Object, ByVal e As System.EventArgs) ' save the changes`
  4. `Protected Sub BtnHR_Click(ByVal sender As Object, ByVal e As System.EventArgs) get the rank codes from the HR`
- Main entities codes - FrmMainEntityCode.aspx
  1. `Protected Sub Page_Load(ByVal sender As Object, ByVal e As System.EventArgs) ' get the data of the main entity`
  2. `Protected Sub BtnSave_Click(ByVal sender As Object, ByVal e As System.EventArgs) ' save the changes of the main entity`
  3. `Protected Sub BtnAddNew_Click(ByVal sender As Object, ByVal e As System.EventArgs) ' add new entity`
  4. `Protected Sub BtnHR_Click(ByVal sender As Object, ByVal e As System.EventArgs) ' get the main entity from the hr`
  5. `Protected Sub GvMainEntity_SelectedIndexChanged(ByVal sender As Object, ByVal e As System.EventArgs) ' get the details of the selected main entity`
- Administrative entities codes - FrmAdminEntityCode.aspx
  1. `Protected Sub Page_Load(ByVal sender As Object, ByVal e As System.EventArgs) 'get the data of the admin. Entity`
  2. `Protected Sub BtnSave_Click(ByVal sender As Object, ByVal e As System.EventArgs) ' save the changes of the admin. Entity`
  3. `Protected Sub BtnAddNew_Click(ByVal sender As Object, ByVal e As System.EventArgs) ' add new admin. Entity`
  4. `Protected Sub GvAdminEntity_SelectedIndexChanged(ByVal sender As Object, ByVal e As System.EventArgs) ' get the details of the selected admin. Entity`
  5. `Protected Sub BtnHR_Click(ByVal sender As Object, ByVal e As System.EventArgs) ' get the admin. Entities from the HR`
- Qualifications codes - FrmQualificationCodes.aspx
  1. `Protected Sub Page_Load(ByVal sender As Object, ByVal e As System.EventArgs) 'show the information about the qualifications`
  2. `Protected Sub BtnSave_Click(ByVal sender As Object, ByVal e As System.EventArgs) ' save the modification of the qualifications`

رموز البيانات الأساسية	
جدول رموز المراتب	
جدول رموز الجهات الرئيسية	
جدول رموز الجهات الإدارية	
جدول رموز المؤهل	
جدول رموز المتصّب	
جدول رموز الجنسية	
جدول رموز التخصصات	
جدول إجادة اللغة الإنجليزية	
جدول بيانات الجنس	
جدول الإحتياجات الخاصة	
جدول مصدر التسجيل	
جدول حالة التصديق	
جدول المهنة	
نوع التقديم على الدورة	
قناة المتدرب	

3. Protected Sub BtnAddNew\_Click(ByVal sender As Object, ByVal e As System.EventArgs) ' add new qualification
4. Protected Sub BtnHR\_Click(ByVal sender As Object, ByVal e As System.EventArgs) ' get the qualifications form the HR
5. Protected Sub GvQualification\_SelectedIndexChanged(ByVal sender As Object, ByVal e As System.EventArgs) ' get the details of the selected qualifications
- Positions codes - FrmPositionCodes.aspx
  1. Protected Sub Page\_Load(ByVal sender As Object, ByVal e As System.EventArgs) ' get the data of the positions
  2. Protected Sub BtnSave\_Click(ByVal sender As Object, ByVal e As System.EventArgs) ' save changes to position
  3. Protected Sub BtnAddNew\_Click(ByVal sender As Object, ByVal e As System.EventArgs) ' add new position
  4. Protected Sub GvPositionCode\_SelectedIndexChanged(ByVal sender As Object, ByVal e As System.EventArgs) ' get the details of the selected position
- Nationalities codes - FrmNationalityCodes.aspx
  1. Protected Sub Page\_Load(ByVal sender As Object, ByVal e As System.EventArgs) ' show the data of the nationality
  2. Protected Sub BtnSave\_Click(ByVal sender As Object, ByVal e As System.EventArgs) ' save the changes to the nationality
  3. Protected Sub BtnHR\_Click(ByVal sender As Object, ByVal e As System.EventArgs) get the nationality from the HR
  4. Protected Sub GvNationalityCode\_SelectedIndexChanged(ByVal sender As Object, ByVal e As System.EventArgs) ' get the details of the selected nationality
- Specialties codes - FrmSpecialityCodes.aspx
  1. Protected Sub Page\_Load(ByVal sender As Object, ByVal e As System.EventArgs) ' get the information of the specialties
  2. Protected Sub BtnSave\_Click(ByVal sender As Object, ByVal e As System.EventArgs) ' save the changes of the specialties
  3. Protected Sub BtnAddNew\_Click(ByVal sender As Object, ByVal e As System.EventArgs) ' add new specialty
  4. Protected Sub GvSpecialityCode\_SelectedIndexChanged(ByVal sender As Object, ByVal e As System.EventArgs) ' get the details of the selected speciality
- Proficiency in English - FrmEnglishProficiency.aspx
  1. Protected Sub Page\_Load(ByVal sender As Object, ByVal e As System.EventArgs) ' show the information of the English proficiency
  2. Protected Sub BtnSave\_Click(ByVal sender As Object, ByVal e As System.EventArgs) ' save the changes to the English proficiency
  3. Protected Sub BtnAddNew\_Click(ByVal sender As Object, ByVal e As System.EventArgs) ' add new proficiency
  4. Protected Sub GvEngProficiency\_SelectedIndexChanged(ByVal sender As Object, ByVal e As System.EventArgs) ' get the details of the selected English proficiency

- Gender code - FrmGenderCode.aspx
  1. Protected Sub Page\_Load(ByVal sender As Object, ByVal e As System.EventArgs) ' show the information of the gender
  2. Protected Sub BtnSave\_Click(ByVal sender As Object, ByVal e As System.EventArgs) save the changes to the gender
  3. Protected Sub GvGender\_SelectedIndexChanged(ByVal sender As Object, ByVal e As System.EventArgs) get the details of the selected gender
- Special needs codes - FrmSpecialNeedsCodes.aspx
  1. Protected Sub Page\_Load(ByVal sender As Object, ByVal e As System.EventArgs) ' get the special needs codes
  2. Protected Sub BtnSave\_Click(ByVal sender As Object, ByVal e As System.EventArgs) save the changes to the existing codes
  3. Protected Sub GvSpecialNeed\_SelectedIndexChanged(ByVal sender As Object, ByVal e As System.EventArgs) ' get the details of the selected code
  4. Protected Sub BtnAddNew\_Click(ByVal sender As Object, ByVal e As System.EventArgs) ' add new code
- Source of registration - FrmSourceofRegistration.aspx
  1. Protected Sub Page\_Load(ByVal sender As Object, ByVal e As System.EventArgs) ' show the source codes
  2. Protected Sub BtnSave\_Click(ByVal sender As Object, ByVal e As System.EventArgs) ' save the changes of the source codes
  3. Protected Sub GvSourceOfReg\_SelectedIndexChanged(ByVal sender As Object, ByVal e As System.EventArgs) ' get the details of the selected source code
  4. Protected Sub BtnAddNew\_Click(ByVal sender As Object, ByVal e As System.EventArgs) ' add new source code
- Certification status schedule - FrmCertiStatusSchedule.aspx
  1. Protected Sub Page\_Load(ByVal sender As Object, ByVal e As System.EventArgs) ' show the certification status schedule data
  2. Protected Sub BtnSave\_Click(ByVal sender As Object, ByVal e As System.EventArgs) ' save the changes of the status schedule data
  3. Protected Sub GvCertiSchedule\_SelectedIndexChanged(ByVal sender As Object, ByVal e As System.EventArgs) ' get the details of the selected status schedule data
  4. Protected Sub BtnAddNew\_Click(ByVal sender As Object, ByVal e As System.EventArgs) ' add new status schedule data
- Profession - FrmProfession.aspx
  1. Protected Sub Page\_Load(ByVal sender As Object, ByVal e As System.EventArgs) ' show the profession code
  2. Protected Sub BtnSave\_Click(ByVal sender As Object, ByVal e As System.EventArgs) ' save the changes to professions
  3. Protected Sub GvProfession\_SelectedIndexChanged(ByVal sender As Object, ByVal e As System.EventArgs) ' get the details of the selected profession

4. Protected Sub BtnAddNew\_Click(ByVal sender As Object, ByVal e As System.EventArgs) ' add new profession
- Type of submission to the course - FrmTypeOfSubmissionCourse.aspx
  1. Protected Sub Page\_Load(ByVal sender As Object, ByVal e As System.EventArgs) ' show data of submission course
  2. Protected Sub BtnSave\_Click(ByVal sender As Object, ByVal e As System.EventArgs) ' save the changes to the submission course
  3. Protected Sub GVSubType\_SelectedIndexChanged(ByVal sender As Object, ByVal e As System.EventArgs) ' get the details of the selected submission course
  4. Protected Sub BtnAddNew\_Click(ByVal sender As Object, ByVal e As System.EventArgs) ' add new submission course
- Applicant's category - FrmApplicantCategory.aspx
  1. Protected Sub Page\_Load(ByVal sender As Object, ByVal e As System.EventArgs) ' show application category information
  2. Protected Sub BtnSave\_Click(ByVal sender As Object, ByVal e As System.EventArgs) ' save the changes to the category information
  3. Protected Sub BtnAddNew\_Click(ByVal sender As Object, ByVal e As System.EventArgs) ' add new category information
  4. Protected Sub GvApplicantCat\_SelectedIndexChanged(ByVal sender As Object, ByVal e As System.EventArgs) ' get the details of the selected category

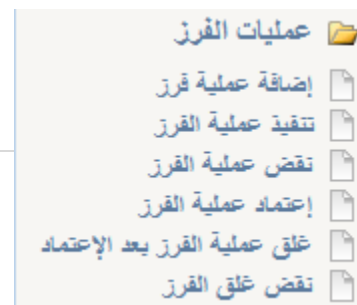
## Update Application

- Update application fields - FrmUpdateApplicationField.aspx
  - 1- Protected Sub BtnSearch\_Click(ByVal sender As Object, ByVal e As System.EventArgs) ' Search applications
  - 2- Protected Sub BtnSave\_Click(ByVal sender As Object, ByVal e As System.EventArgs) ' Save / Update Application information
- Certify applications data - FrmCertifyApplicationData.aspx
  - 1- Protected Sub BtnSearch\_Click(ByVal sender As Object, ByVal e As System.EventArgs) ' Search Application by Id and by certify status
  - 2- Protected Sub BtnSaveSelected\_Click(ByVal sender As Object, ByVal e As System.EventArgs) ' Save certify status for Selected application
  - 3- Protected Sub BtnSave\_Click(ByVal sender As Object, ByVal e As System.EventArgs) ' Save/update and change certify status of all fields at once



## Filtration Operations

- Defining new filtration - FrmNewFiltration.aspx



1. Protected Sub Page\_Load(ByVal sender As Object, ByVal e As System.EventArgs) ' show the filter operations
2. Protected Sub BtnDiffElementsGroupSave\_Click(ByVal sender As Object, ByVal e As System.EventArgs) ' save filtration element group
3. Protected Sub BtnCondtionGroupCodeSave\_Click(ByVal sender As Object, ByVal e As System.EventArgs) ' save conditions group
4. Protected Sub BtnCourseFilterSave\_Click(ByVal sender As Object, ByVal e As System.EventArgs) ' add the courses and assign it to filtration operation
5. Protected Sub BtnMainSave\_Click(ByVal sender As Object, ByVal e As System.EventArgs) ' save and edit the whole filter operation
- Filtration process execution - FrmFilterProcessExecution.aspx
  1. Protected Sub BtnImpFilter\_Click(ByVal sender As Object, ByVal e As System.EventArgs) ' execute the selected filtration operation
  2. Protected Sub Page\_Load(ByVal sender As Object, ByVal e As System.EventArgs) ' show the filtration operations
- Filtration operation reversal - FrmFiltrationoperationreversal.aspx
  1. Protected Sub Page\_Load(ByVal sender As Object, ByVal e As System.EventArgs) ' get the filters
  2. Protected Sub BtnImpFilter\_Click(ByVal sender As Object, ByVal e As System.EventArgs) ' reverse the selected filtration operation
- Filtration operation approval - FrmFiltrationoperationapproval.aspx
  1. Protected Sub Page\_Load(ByVal sender As Object, ByVal e As System.EventArgs) ' get the filters
  2. Protected Sub BtnImpFilter\_Click(ByVal sender As Object, ByVal e As System.EventArgs) ' change the selected filtration operation to be approved
- Close filtration operation after approval - FrmClosefiltration.aspx
  1. Protected Sub Page\_Load(ByVal sender As Object, ByVal e As System.EventArgs) ' get the filters
  2. Protected Sub BtnImpFilter\_Click(ByVal sender As Object, ByVal e As System.EventArgs) ' change the selected filtration operation to be approved
- Reversal of closed filtration operation - FrmReversalofclosed.aspx
  1. Protected Sub Page\_Load(ByVal sender As Object, ByVal e As System.EventArgs) ' get the filters
  2. Protected Sub BtnImpFilter\_Click(ByVal sender As Object, ByVal e As System.EventArgs) ' change the selected filtration operation to be approved

### Trainee's nomination

- Trainees nomination - FrmTraineeNominations.aspx
  - 1- Protected Sub Page\_Load(ByVal sender As Object, ByVal e As System.EventArgs) ' Bind Offers, Courses
  - 2- Protected Sub btnAddRequest\_Click(ByVal sender As Object, ByVal e As System.EventArgs) ' Add Application requests to selected students
  - 3- Protected Sub BtnSave\_Click(ByVal sender As Object, ByVal e As System.EventArgs) ' Register selected students on the selected courses

ترشيح متدربين	
ترشيح متدربين	
إضافة متدرب من قبل الإداري أو المقوض	
ترشيح على دورات	
شاشة ترشيح متدربين لبرنامج تدريبي	
إضافة مقبولين على نتائج الفرز بعد الاعتماد	
إدخال و تحديث بيانات متدرب	
إدخال الإعتذار و حالات التخلف و تحديثها لمجموعة	
إدخال الإعتذار و حالات التخلف و تحديثها	
إدخال إعفاء من المتطلبات السابقة	
إدخال السجلات الإفتتاحية	
إضافة و تعديل بيانات مدرب	
للإطلاع على تقييم الدورات من قبل المتقدمين	

- Add new trainee to the system by admin - FrmAddNewTrainee.aspx
  - 1- `Protected Sub BtnCopyDataFromHR_Click(ByVal sender As Object, ByVal e As System.EventArgs) \ Bring Employee profile data from madar service`
  - 2- `Protected Sub BtnCopyDataFromFile_Click(ByVal sender As Object, ByVal e As System.EventArgs) \ Copy Student data from Student service`
  - 3- `Protected Sub BtnDelete_Click(ByVal sender As Object, ByVal e As System.EventArgs) Handles BtnDelete.Click \ Delete User Profile from System`
  - 4- `Protected Sub BtnSave_Click(ByVal sender As Object, ByVal e As System.EventArgs) \ Add New Applicant to system.`
- Course nomination - FrmCourseNomination.aspx
  - 1- `Protected Sub BtnSearch_Click(ByVal sender As Object, ByVal e As System.EventArgs) \ Search Applicants`
  - 2- `Protected Sub GvCourseRegistered_RowCommand(ByVal sender As Object, ByVal e As System.Web.UI.WebControls.GridViewCommandEventArgs) \ Nominate Selected course for Resulted Applicants From Search`
- Nominating applicants to programs - FrmNominateApplicantProgram.aspx
  - 1- `Protected Sub Page_Load(ByVal sender As Object, ByVal e As System.EventArgs) \ Get list of All programs in Application`
  - 2- `Protected Sub BtnSearch_Click(ByVal sender As Object, ByVal e As System.EventArgs) \ Search programs by program No.`
  - 3- `Protected Sub btnAddRequest_Click(ByVal sender As Object, ByVal e As System.EventArgs) \ Add Applications for listed students on the selected program`
  - 4- `Protected Sub BtnSave_Click \ Add\ Register Listed users on selected program course.`
- Add applicant to filtration result after approval - FrmAddApplicationtoFilterResultApproval.aspx
  - 1- `Protected Sub Page_Load(ByVal sender As Object, ByVal e As System.EventArgs) \ bind Filters`
  - 2- `Protected Sub ddFilterNo_SelectedIndexChanged(ByVal sender As Object, ByVal e As System.EventArgs) Publish Selected Filter Data`
  - 3- `Protected Sub BtnSearch_Click(ByVal sender As Object, ByVal e As System.EventArgs) \ Search specific Applicant`
  - 4- `Protected Sub BtnAddNew_Click(ByVal sender As Object, ByVal e As System.EventArgs) \ Append Applicant to Accepted applicants by filter`
- Applicant's profile management - FrmApplicantprofilemanagement.aspx
  - 1- `Protected Sub BtnCopyDataFromHR_Click(ByVal sender As Object, ByVal e As System.EventArgs) \ bring Employee data from service`
  - 2- `Protected Sub BtnCopyDataFromFile_Click(ByVal sender As Object, ByVal e As System.EventArgs) \ bring student data from Students service`

- 3- `Protected Sub BtnDelete_Click(ByVal sender As Object, ByVal e As System.EventArgs) \ Delete Applicant profile`
  - 4- `Protected Sub BtnSearch_Click(ByVal sender As Object, ByVal e As System.EventArgs) \ bring existing profile from system`
  - 5- `Protected Sub BtnSave_Click(ByVal sender As Object, ByVal e As System.EventArgs) \ Add/Update applicant profile to system`
- Add pardon and attendance failure For Group - `FrmAddPardonAttendanceFailure.aspx`
    - 1- `Protected Sub Page_Load(ByVal sender As Object, ByVal e As System.EventArgs) \ bind courses`
    - 2- `Protected Sub ddCourseCode_SelectedIndexChanged(ByVal sender As Object, ByVal e As System.EventArgs) \ Return Registered applicants on selected course`
    - 3- `Protected Sub BtnImpFilter_Click(ByVal sender As Object, ByVal e As System.EventArgs) \ Save Excuse and absence selection for applicants to system`
  - Enter pardon from course and attendance failure - `FrmEnterpardonFailure.aspx`
    - 1- `Protected Sub Page_Load(ByVal sender As Object, ByVal e As System.EventArgs) \ Bind Applicants`
    - 2- `Protected Sub BtnImpFilter1_Click(ByVal sender As Object, ByVal e As System.EventArgs) \ bind all courses that the selected user can pardon`
    - 3- `Protected Sub BtnImpFilter_Click(ByVal sender As Object, ByVal e As System.EventArgs) \ Save new status to the selected applicant "Accepted, Not Accepted, Pardoned, Abscent"`
  - Add pardon from course prerequisite - `FrmAddPardonPreRequisite.aspx`
    - 1- `Protected Sub BtnImpFilter1_Click(ByVal sender As Object, ByVal e As System.EventArgs) \ Search Applicants by User no`
    - 2- `Protected Sub BtnIntroExemption_Click(ByVal sender As Object, ByVal e As System.EventArgs) \ Enter Course PreRequisite Exemption to Selected Applicant & Course`
  - Add and update the opening records - `FrmAddUpdateOpenRecords.aspx`
    - 1- `Protected Sub BtnSearch_Click(ByVal sender As Object, ByVal e As System.EventArgs) \ Search Applicant by Id or Name`
    - 2- `Protected Sub BtnSave_Click(ByVal sender As Object, ByVal e As System.EventArgs) \ Enter\Update User Filtration opening Records for weights.`
  - Add and update trainer's profile - `FrmAddUpdateTrainerProfile.aspx`
    - 1- `Protected Sub BtnSearch_Click(ByVal sender As Object, ByVal e As System.EventArgs) \ Search and populate Trainer Data`
    - 2- `Protected Sub BtnDelete_Click(ByVal sender As Object, ByVal e As System.EventArgs) \ Delete selected trainer`



3- `Protected Sub BtnSave_Click(ByVal sender As Object, ByVal e As System.EventArgs) \ Add/Update trainer info.`

- Review course evaluation - FrmReviewCourseEval.aspx

1- `Protected Sub Page_Load(ByVal sender As Object, ByVal e As System.EventArgs) \ bind Evaluated courses`

2- `Protected Sub ddCourseCode_SelectedIndexChanged(ByVal sender As Object, ByVal e As System.EventArgs) \ publish course info and registered applicants`

3- `Protected Sub GvApplicants_RowCommand(ByVal sender As Object, ByVal e As System.Web.UI.WebControls.GridViewCommandEventArgs) \ preview selected applicant evaluation`

## Reports

- All Reports lies in Reports Folder
- Reports are designed by crystal reports
- Every Report is designed based on SQL SERVER Stored Procedure
- Every Report published throw web form designed to take report parameter throw form controls
- Every web form publish report throw report viewer
- Every web form pass report data by dataset calling the same stored procedure and passing its parameter
- Sample Report :

1- Report design



2- Form code :

```
Dim objds As DataSet
Dim ClsRpt As New ReportsInfo
Dim ClsCom As New CommonControlFunction
```

```
Private Sub ConfigureReport()
    'If Not Page.IsPostBack Then
        Dim myConnectionInfo As ConnectionInfo = ClsCom.ConnectToReport()
```



```
Dim rptdoc As New ReportDocument

objds = ClsRpt.GetAttendancePercentage(ddCourse.SelectedValue)

Dim ReportPath As String = Server.MapPath("rptAttendancePercentage.rpt")
'GetReport.ReportSource = ReportPath
rptdoc.Load(ReportPath)
Dim parameters As New ParameterFields ' = GetReport.ParameterFieldInfo
Dim CourseNo As New ParameterField ' = New ParameterField()

CourseNo.Name = "CourseNo"
Dim vlas As New ParameterDiscreteValue ' = New ParameterDiscreteValue()
'CourseNo.Name = "par_customerid"
'CourseNo.Name = "CourseNo"
vlas.Value = ddCourse.SelectedValue
CourseNo.CurrentValues.Add(vlas)
parameters.Add(CourseNo)
GetReport.ParameterFieldInfo = parameters
rptdoc.SetDataSource(objds.Tables(0))

GetReport.ReportSource = rptdoc
SetDBLogonForReport(myConnectionInfo)
' End If
End Sub

Private Sub SetDBLogonForReport(ByVal myConnectionInfo As ConnectionInfo)
Dim myTableLogOnInfos As TableLogOnInfos = GetReport.LogOnInfo()
For Each myTableLogOnInfo As TableLogOnInfo In myTableLogOnInfos
    myTableLogOnInfo.ConnectionInfo = myConnectionInfo
Next
End Sub

Protected Sub BtnView0_Click(ByVal sender As Object, ByVal e As System.EventArgs)
Handles BtnView0.Click ` Export Report
Dim myConnectionInfo As ConnectionInfo = ClsCom.ConnectToReport()
Dim rptdoc As New ReportDocument
objds = ClsRpt.GetAttendancePercentage(ddCourse.SelectedValue)
Dim ReportPath As String = Server.MapPath("rptAttendancePercentage.rpt")
rptdoc.Load(ReportPath)
'rptdoc.SetParameterValue("CourseNo", ddCourse.SelectedValue)
rptdoc.SetDataSource(objds.Tables(0))
If (ddExportTypes.SelectedValue = 1) Then

rptdoc.ExportToHttpResponse(CrystalDecisions.Shared.ExportFormatType.WordForWindows
, Response, True, "rptAttendancePercentage")
Else
    If (ddExportTypes.SelectedValue = 2) Then

rptdoc.ExportToHttpResponse(CrystalDecisions.Shared.ExportFormatType.Excel,
Response, True, "rptAttendancePercentage")
Else
    If (ddExportTypes.SelectedValue = 3) Then

rptdoc.ExportToHttpResponse(CrystalDecisions.Shared.ExportFormatType.PortableDocFor
mat, Response, True, "rptAttendancePercentage")
Else
    If (ddExportTypes.SelectedValue = 4) Then
```

```

rptdoc.ExportToHttpResponse(CrystalDecisions.Shared.ExportFormatType.ExcelRecord,
Response, True, "rptAttendancePercentage")
    End If
    End If
End If
End If

End Sub

```

## App\_Code Classes

**PageClass**  
Class  
→ Page

- Fields
  - EntryMode
- Methods
  - CheckPageList
  - GetUserACR
  - InitializeCulture
  - Page\_Load

**MiscFunction**  
Class

- Fields
  - ClsAddNewApp...
- Methods
  - ApplicationBy
  - CheckValueExis...
  - CheckValueinLi...
  - ConvertDateFo...
  - ExposureStatus
  - GetAcceptingSt...
  - GetAcceptingSt...
  - GetColumnNa...
  - GetCourseStatus
  - GetEligibilitySta...
  - GetExamStatus
  - GetExemptionS...
  - GetOfferType
  - GetSubmission...
  - GetViewCode
  - GetViewStatus
  - SendMail
- Nested Types

**GenericCulture...**  
Class  
→ CultureInfo

- Fields
  - cal
  - dtfi
- Properties
  - Calendar
  - DateTimeFormat
- Methods
  - New (+ 1 overl...

**CommonContro...**  
Class

- Fields
  - ClsAppInfo
  - Coninfo
  - ObjDs
- Methods
  - BindCodestoDr...
  - BindToDropDo...
  - CodetoGridView
  - ConnectToRep...
  - ExportReportC...
  - GetCourseCode
  - SortList

**DateConvert**  
Class

- Fields
  - allFormats
  - arCul
  - cur
  - enCul
  - endGreg
  - g
  - getdate
  - h
  - startGreg
- Methods
  - Compare
  - FormatGreg
  - FormatHijri
  - GDateNow (+ 1...
  - GregToHijri (+...
  - GTimeStamp
  - HDateNow (+ 1...
  - HijriToGreg (+...
  - HTimeStamp
  - IsGreg
  - IsHijri
  - New

**SessionClass**  
Class

- Properties
  - ApplicationID
  - ApprovalUploa...
  - AttachmentFile...
  - AttachmentSu...
  - CanApprove
  - CollegeCode
  - CVAttachmentF...
  - DepartmentCode
  - Email
  - EmpID
  - EmpName
  - EmpSocialID
  - IsAdmin
  - IsApplicant
  - IsCollegeDean
  - IsDepartmentH...
  - IsHR
  - IsScientificCou...
  - IsSkillsDevelop...
  - IsSkillsDevelop...
  - IsSkillsDevelop...
  - IsStandingCom...
  - Mode
  - OfferNo
  - ParentID
  - Pasc
  - Profession
  - ReportApplicati...
  - ReportPageStat...
  - UserID
  - UserName
- Methods
  - ClearSession

## Data Access Layer

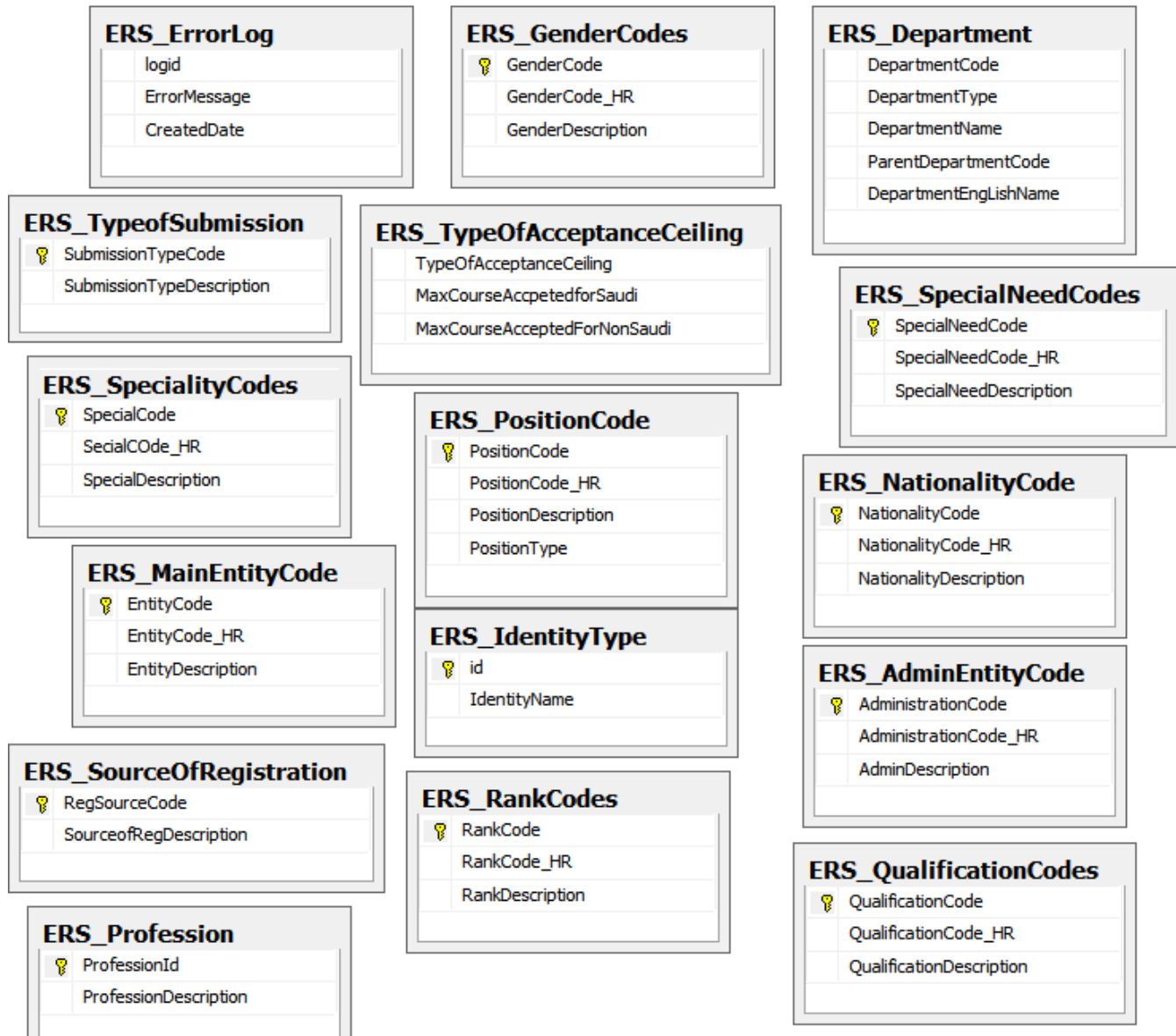
### DAL Project

#### *Public Class OfferInfo*

---

```
Public Function GetAllAvailableOffer() As DataSet
Call DB Stored procedure: ERS_SP_GetAllAvailableOffers
Request All offers that Current date lies between its start date and end date
```


## Database



**ERS\_Country**

CountryID
CountryName
IsSuspend
CountryNameEn

**ERS\_TrainerProfile**

 TrainerID
TrainerName
Email
MobileNo
AboutTrainer
Qualification
PastExperience


**ERS\_ApplicantCategory**

 ApplicantCatCode
ApplicantCatDescription


**ERS\_TrainingRoomProfile**

RoomNo
RoomName
BuildingName
CourseCode
Description
NoOfSeats
BlackBoard
Projector
Computer
OtherEquipment


**ERS\_ApplicantProfile**

 TraineeNo
TraineeName
UserName
Password
IdentityType
IdentityNo
Gender
MainEntity
Status
Profession
RegistrationSource
EmployeeNo
StudentNo
Email

**ERS\_UserRegistration**

 id
FirstName
FirstNameAr
MiddleName
MiddleNameAr
FamilyName
FamilyNameAr
Email
Username
Password
IdentityType
IdentityNo
Nationality
TelephoneNo
MobileNo
AdvancedviewHome
Profession
Gender
Status
OfficeNo
FaxNo
GroupNo
FullName

**ERS\_GroupList**

 id
GroupName
GroupDescription

**ERS\_GroupAccess**

 ID
GroupID
OperationNo
SecurityMode

**ERS\_NewOfferOperation**

OfferNo
OfferDescription
OfferType
SubmissionStartDate
SubmissionEndDate
NumberOfSaudiAllowed
MaxCourseAllowedForSaudi
NumberOfNonSaudiAllow
MaxCourseAllowedForNonSaudi
WeightandExpenseStartDate
WeightandExpenseEndDate
TargetCategory

**ERS\_CourseTrainers**

CourseNo
TrainerId

**ERS\_CourseAttendanceRecord**

CourseNo
SubjectNo
VersionNo
ApplicationNo
TodayDate
TrainingPeriodCode
CountMinuteForLateandabsence
NumberofMinuteAuthorised
ApplicantNo

**ERS\_CourseAttendanceResult**

ApplicantNo
ApplicationNo
OfferNo
CourseNo
InitialAttResult
FinalAttResult
InitialCertiStatus
FinalCertiStatus
ExamPassing

**ERS\_Course**

CourseNo
OfferNo
SubjectNo
VersionNo
CourseStartDate
CourseEndDate
CourseStartDateEn
CourseEndDateEn
CourseDescription
CourseStatus
CoursePlace
PortCourse
CourseCapacity
CourseCity
CourseCertiType
ListCapacity
UpperLimitofApplicant
LowerLimitofApplicant
ToolsRequired
CancelReason
CancelDate

**ERS\_CourseEvaluation**

ID
ApplicationNo
CourseNo
OfferNo
ApplicantNo
Q1
Q2
Q3
Q4
Q5
Q6

**ERS\_Subject**

SubjectNo
SubjectName

**ERS\_SubjectVersion**

ID
VersionNo
SubjectNo
SubjectVersionName
SubjectVersionType
SubjectVersionNature
SubjectVersionUnit
SubjectVersionStatus

**ERS\_RelatedTrainingCourse**

CourseNo
Rel_CourseNo

**ERS\_RelatedTrainingProgram**

ProgramNo
CourseNo
OfferNo

**ERS\_CertificateIssuance**

CertiIssuanceStatusCode
CertiIssuanceStatusDetail
NumberofCertiIssued

**ERS\_AddNewApplication**

ApplicationNo
OfferNo
ApplicantNo
ApplicationBy
MobileNo
Email
SubAdministration
OtherSubAdmin
RankCode
OtherJobCode
QualificationCode
SpecialCode
OtherSpecial
PositionCode

**ERS\_SubjectPrerequisite**

PreRequisiteCode
SubjectNo

**ERS\_SubjectLecture**

id
SubjectNo
VersionNo
UnitTimeofTraining
TrainingDuration

**ERS\_LectureTimeSchedule**

id
CourseNo
SubjectNo
VersionNo
UnitOfTimeTraining
TimeofBeginPeriod
DayCode
PeriodHall

**ERS\_CertificatePrintIssue**

ID
UserId
CourseNo
IssueDate
PrintDate
CertificateType

**ERS\_ApplicationCourse**

OfferNo
ApplicationNo
CourseNo
TypeofSubmission
TraineeClass
CoursePardonDate
TraineeClassCertiStatus
CoursePriority
RegistrationDate
ApplicantNo
RegistrationStatus
RegistrationNo
Administrative_Precedence
Competition_Exempted

**ERS\_DifferentiationElements**

⚡ DifferentiationElementCode  
 DifferentiationElementDescription  
 Isdeleted

**ERS\_DetailsofDifferentiation**

id  
 ⚡ DifferentiationCode  
 ⚡ DifferentiationElementCode

**ERS\_RankPositionDifferentiationMember**

Differentiationelementgroup  
 ⚡ RankPositionCode  
 RankPositionDifferentiationElementWeight  
 IsDeleted

**ERS\_ConditionGroup**

⚡ GroupConditionCode  
 ConditionType  
 GroupConditionDetail

**ERS\_ConditionGroupofFiltrationOperation**

⚡ ID  
 OfferNo  
 FilterOperationNo  
 GroupConditionCode  
 DisplayStatus

**ERS\_FiltrationOperation**

OfferNo  
 ⚡ FilterOperationNo  
 FilterOperationDetail  
 FilterStatus  
 DifferentiationCode  
 SexualDiversityLimit  
 ManageDiversityLimit  
 TypeofAcceptance  
 TypeofSubmission

**ERS\_DetailConditionGroup**

id  
 GroupConditionCode  
 GroupConditionElementCode  
 CodeDescription

**ERS\_ConditionType**

⚡ ConditionType  
 ConditionTypeDescription  
 StageofConditionApplication  
 ConditionStatus

**ERS\_OpeningRecords**

ApplicantNo  
 OfferType  
 RecordOpenDate  
 NumberofTimeAccept  
 NumberoffailuretoAttend  
 NumberofTimeProcessWithou...  
 NumberofTimeCertiIssued

**ERS\_DifferentiationElementGroupofFiltrationOperation**

⚡ ID  
 OfferNo  
 FilterOperationNo  
 DifferentiationElementGroup

**ERS\_FiltrationOperationCourse**

⚡ ID  
 OfferNo  
 FilterOperationNo  
 CourseNo  
 FilterLimit  
 LimitSaudizationWithoutCompetition  
 NumberViewCourse  
 CourseLimit

**ERS\_DifferentiationElementsGroup**

⚡ DifferentiationElementGroupCode  
 DifferentiationElementGroupDescription  
 DifferentiationElementCode  
 Isdeleted

**ERS\_FiltrationOrder**

⚡ ID  
 OfferNo  
 CourseNo  
 FilterOperationNo  
 ApplicationNumber  
 ApplicantNo  
 AdministrativePrecedence  
 AdministrativeDiversityCoeffi...  
 CoursePrecedenceForTheAp...  
 CourseRegistration  
 NationalityCoefficient  
 NationalityDiversityCoefficient  
 AcceptanceWeight  
 AttendanceFailureWeight  
 CertificatesWeight  
 Filingweight  
 RankWeight  
 FilterResult  
 Accepted  
 Nationality  
 PositionWeight  
 Administration

**ERS\_FiltrationResults**

FilterOperationNO  
 OfferNo  
 ApplicantNo  
 ApplicationNo  
 CourseNo  
 Diversitysexual  
 diversityManagement  
 RankingResult  
 FilterResult  
 AcceptStatusAfterFilter  
 Remarks