

Grandes étapes du projet Labo-2

- 1- Bâtir le projet Labo-2 (WebForm vide)
- 2- Importer dans le projet (ClientFormUtilities.js, FormStyle.css et SqlExpressUtilities.cs, Images)
- 3- Créer la base de données MainDB.mdf
- 4- Ajouter la table USERS dans le répertoire App_Data

```
CREATE TABLE [dbo].[USERS] (  
    [ID] BIGINT IDENTITY (1, 1) NOT NULL,  
    [USERNAME] NVARCHAR (50) NOT NULL,  
    [PASSWORD] NVARCHAR (50) NOT NULL,  
    [FULLNAME] NVARCHAR (50) NULL,  
    PRIMARY KEY CLUSTERED ([ID] ASC)  
);
```

- 5- Ajouter dans Global.asax/Application_Start :

```
string DB_Path = Server.MapPath(@"~\App_Data\MainDB.mdf");  
Application["MainDB"] = @"Data Source=(LocalDB)\v11.0;AttachDbFilename='" + DB_Path + "'; Integrated Security=true";
```

- 6- Peupler la table USERS

ID	USERNAME	PASSWORD	FullName
3	ATremblay	Lapin_2015	Alain Tremblay
4	JLapointe	Piano_Bar	Jean Lapointe
5	ADMIN	PASSWORD	Administrateur

- 7- Créer la classe USERS qui dérive de la classe SqlExpressWrapper

```
public class USERS:SqlExpressUtilities.SqlExpressWrapper  
{  
    public long ID { get; set; }  
    public String UserName { get; set; }  
    public String Password { get; set; }  
    public String FullName { get; set; }  
    public USERS(String connexionString, System.Web.UI.Page Page)  
        : base(connexionString, Page)  
    {  
        SQLTableName = "USERS";  
    }  
    public override void GetValues()  
    {  
        ID = long.Parse(FieldsValues[0]);  
        UserName = FieldsValues[1];  
        Password = FieldsValues[2];  
        FullName = FieldsValues[3];  
    }  
    public override void Insert()  
    {  
        InsertRecord(UserName, Password, FullName);  
    }  
    public override void Update()  
    {  
        UpdateRecord(ID, UserName, Password, FullName);  
    }  
}
```

8- Créer la page ShowUsers.aspx et ajouter le panel PN_GridView:

ShowUsers.aspx

```
<%@ Page Language="C#" AutoEventWireup="true" CodeBehind="ShowUsers.aspx.cs" Inherits="Labo_2.ShowUsers" %>
<!DOCTYPE html>
<html xmlns="http://www.w3.org/1999/xhtml">
<head runat="server">
<meta http-equiv="Content-Type" content="text/html; charset=utf-8"/>
<title></title>
</head>
<body>
<form id="form1" runat="server">
<div>
<asp:Panel id="PN_GridView" runat="server"></asp:Panel>
</div>
</form>
</body>
</html>
```

ShowUsers.aspx.cs

```
using System;
using System.Collections.Generic;
using System.Linq;
using System.Web;
using System.Web.UI;
using System.Web.UI.WebControls;

namespace Labo_2
{
    public partial class ShowUsers : System.Web.UI.Page
    {
        protected void Page_Load(object sender, EventArgs e)
        {
            USERS users = new USERS((string)Application["MainDB"], this);
            users.SelectAll();
            users.MakeGridView(PN_GridView, "" /*pas de page d'édition pour l'instant */);
        }
    }
}
```

9- Lancer la page ShowUsers.aspx :

	ID	USERNAME	PASSWORD	FullName
<u>3</u>		ATremblay	Lapin_2015	Alain Tremblay
<u>4</u>		JLapointe	Piano_Bar	Jean Lapointe
<u>5</u>		ADMIN	PASSWORD	Administrateur

10- Créer la page Login.aspx

```
<%@ Page Language="C#" AutoEventWireup="true" CodeBehind="Login.aspx.cs" Inherits="Labo_2.Login" %>
<!DOCTYPE html>
<html xmlns="http://www.w3.org/1999/xhtml">
<head runat="server">
<meta http-equiv="Content-Type" content="text/html; charset=utf-8"/>
<title></title>
<script src="//code.jquery.com/jquery-1.10.2.js"></script>
<link rel="stylesheet" href="FormStyles.css"/>
</head>
<body>
<h2>Login...</h2> <hr />
<form id="form1" runat="server">
<div style="margin:auto; width:300px; background-color:lightgray; padding:20px; border:1px solid black;">
<table>
<tr>
<td> <asp:Label for="TB_UserName" runat="server" Text="Nom d'utilisateur"></asp:Label> </td>
<td> <asp:TextBox ID="TB_UserName" runat="server" CssClass="TextBox"></asp:TextBox> </td>
</tr>
<tr>
<td> <asp:Label for="TB_Password" runat="server" Text="Mot de passe"></asp:Label> </td>
<td> <asp:TextBox ID="TB_Password" runat="server" CssClass="TextBox" TextMode="Password"></asp:TextBox>
</td>
</tr>
<tr>
<td></td>
<td> <asp:Button ID="BTN_Login" runat="server" Text="Soumettre..." class="submitBTN"/> </td>
</tr>
</table>
</div>
</form>
</body>
</html>
```

Login...

Nom d'utilisateur	<input type="text"/>
Mot de passe	<input type="password"/>
<input type="button" value="Soumettre..."/>	

11- Ajouter dans le fichier Web.config `<add key="ValidationSettings:UnobtrusiveValidationMode" value="None" />` :

```
<?xml version="1.0" encoding="utf-8"?>
<!--
  Pour plus d'informations sur la configuration de votre application ASP.NET, consultez
  http://go.microsoft.com/fwlink/?LinkId=169433
-->
<configuration>

  <appSettings>
    <add key="ValidationSettings:UnobtrusiveValidationMode" value="None" />
  </appSettings>

  <system.web>
    <compilation debug="true" targetFramework="4.5" />
    <httpRuntime targetFramework="4.5" />
  </system.web>
</configuration>
```

12- Ajouter les « *validator* » suivants :

À droite de la cellule du nom d'utilisateur (TB_UserName)

```
<td>
  <asp:RequiredFieldValidator ID="RFV_TB_UserName" runat="server"
    Text="!vide"
    ErrorMessage="Le nom d'utilisateur est vide!"
    ControlToValidate="TB_UserName"
    ValidationGroup="VG_Login">
  </asp:RequiredFieldValidator>
</td>
```

À droite de la cellule du mot de passe (TB_Password)

```
<asp:RequiredFieldValidator ID="RFV_TB_Password" runat="server"
  Text="!vide"
  ErrorMessage="Le mot de passe est vide!"
  ControlToValidate="TB_Password"
  ValidationGroup="VG_Login">
</asp:RequiredFieldValidator>
```

En tant que dernière rangée du tableau du formulaire

```
<tr>
  <td colspan="3" style="text-align:left;">
    <asp:ValidationSummary ID="VGS_Logi" runat="server"
      ValidationGroup="VG_Login"
      HeaderText="Résumé des erreurs: &lt;hr/&gt;"
    />
  </td>
</tr>
```

Login...

Nom d'utilisateur	<input type="text"/>	!vide
Mot de passe	<input type="password"/>	!vide
<input type="button" value="Soumettre..."/>		
Résumé des erreurs:		
<ul style="list-style-type: none">• Le nom d'utilisateur est vide!• Le mot de passe est vide!		

13- Ajouter les méthodes suivantes dans la classe USERS :

```
public bool Exist(String userName)
{
    bool exist = false;
    QuerySQL("SELECT * FROM " + SQLTableName + " WHERE USERNAME='" + userName + "'");
    if (reader.HasRows)
    {
        Next();
        GetValues();
        exist = true;
    }
    return exist;
}

public bool Valid(String userName, String Password)
{
    bool valid = false;
    if (Exist(userName))
        valid = (this.Password == Password);
    return valid;
}
```

14- Ajouter le code suivant dans Login.aspx.cs :

```
protected void Page_Load(object sender, EventArgs e)
{
    if (!Page.IsPostBack)
    {
        Session["CurrentUser"] = new USERS((String)Application["MainDB"], this);
        Session["UserValid"] = false;
    }
}
```

15- Insérer le « custom validator » suivant dans Login.asp comme dernier élément du formulaire

Login.aspx

```
<asp:CustomValidator ID="CV_TB_UserName" runat="server"
    ErrorMessage="Ce nom d'utilisateur n'existe pas!"
    Display="None"
    ValidationGroup="VG_Login"
    OnServerValidate="CV_TB_UserName_ServerValidate">
</asp:CustomValidator>

<asp:CustomValidator
    ID="CV_Password" runat="server"
    ErrorMessage="Le mot de passe est incorrect!"
    ControlToValidate="TB_Password"
    ValidationGroup="VG_Login"
    Display="None" OnServerValidate="CV_Password_ServerValidate">
</asp:CustomValidator>
```

Login.aspx.cs

```
protected void CV_TB_UserName_ServerValidate(object source, ServerValidateEventArgs args)
{
    USERS users = (USERS)Session["CurrentUser"];
    args.IsValid = users.Exist(TB_UserName.Text);
}

protected void CV_Password_ServerValidate(object source, ServerValidateEventArgs args)
{
    USERS users = (USERS)Session["CurrentUser"];
    args.IsValid = users.Valid(TB_UserName.Text, TB_Password.Text);
}
```

16- Lancer Login.aspx et tester :

Login...

Nom d'utilisateur

Vide!

Mot de passe

Vide!

Sommaire des erreurs

- Le nom de l'utilisateur est vide!
- Le mot de passe est vide!

Login...

Nom d'utilisateur

Mot de passe

Sommaire des erreurs

- Le mot de passe est incorrect!

17- Ajouter une nouvelle page aspx :

Accueil.aspx

```
<%@ Page Language="C#" AutoEventWireup="true" CodeBehind="Accueil.aspx.cs" Inherits="Labo_22.Accueil" %>
<!DOCTYPE html>
<html xmlns="http://www.w3.org/1999/xhtml">
<head runat="server">
  <meta http-equiv="Content-Type" content="text/html; charset=utf-8" />
  <title></title>
</head>
<body>
  <form id="form1" runat="server">
    <h2>Bienvenue</h2>
    <asp:Label ID="LBL_FullName" runat="server" Text=""></asp:Label>
    <hr />
    <div>
    </div>
  </form>
</body>
</html>
```

Accueil.aspx.cs

```
protected void Page_Load(object sender, EventArgs e)
{
  Session["message"] = "vous devez vous authentifier!!!";
  if (Session["UserValid"] == null)
    Response.Redirect("Login.aspx");

  Session["message"] = "";

  if (!(bool)Session["UserValid"])
    Response.Redirect("Login.aspx");
  else
    LBL_FullName.Text = ((USERS)Session["CurrentUser"]).FullName;
}
```

18- Ajouter dans Login.aspx.cs un delegate sur le clic du bouton BTN_Click:

```
    <asp:Button ID="BTN_Login" runat="server"
      Text="Soumettre..."
      class="submitBTN"
      ValidationGroup="VG_Login" OnClick="BTN_Login_Click"
    />

    protected void BTN_Login_Click(object sender, EventArgs e)
    {
      if (Page.IsValid)
      {
        Session["UserValid"] = true;
        Response.Redirect("Accueil.aspx");
      }
    }
  }
```

19- Ajouter dans Login.aspx un le label LBL_Mesage:

```
<body>
  <h2>Login...</h2>
  <asp:Label ID="LBL_Message" runat="server" Text=""></asp:Label>
  <hr />
  <form id="form1" runat="server">
    <div style="margin: auto; width: 350px; background-color: lightgray; padding: 20px; border: 1px solid black;">
      <table>
```