



Caso de Estudio:

Diseño e Implementación de la Capa Web de MiniPortal

Marcos de Desarrollo

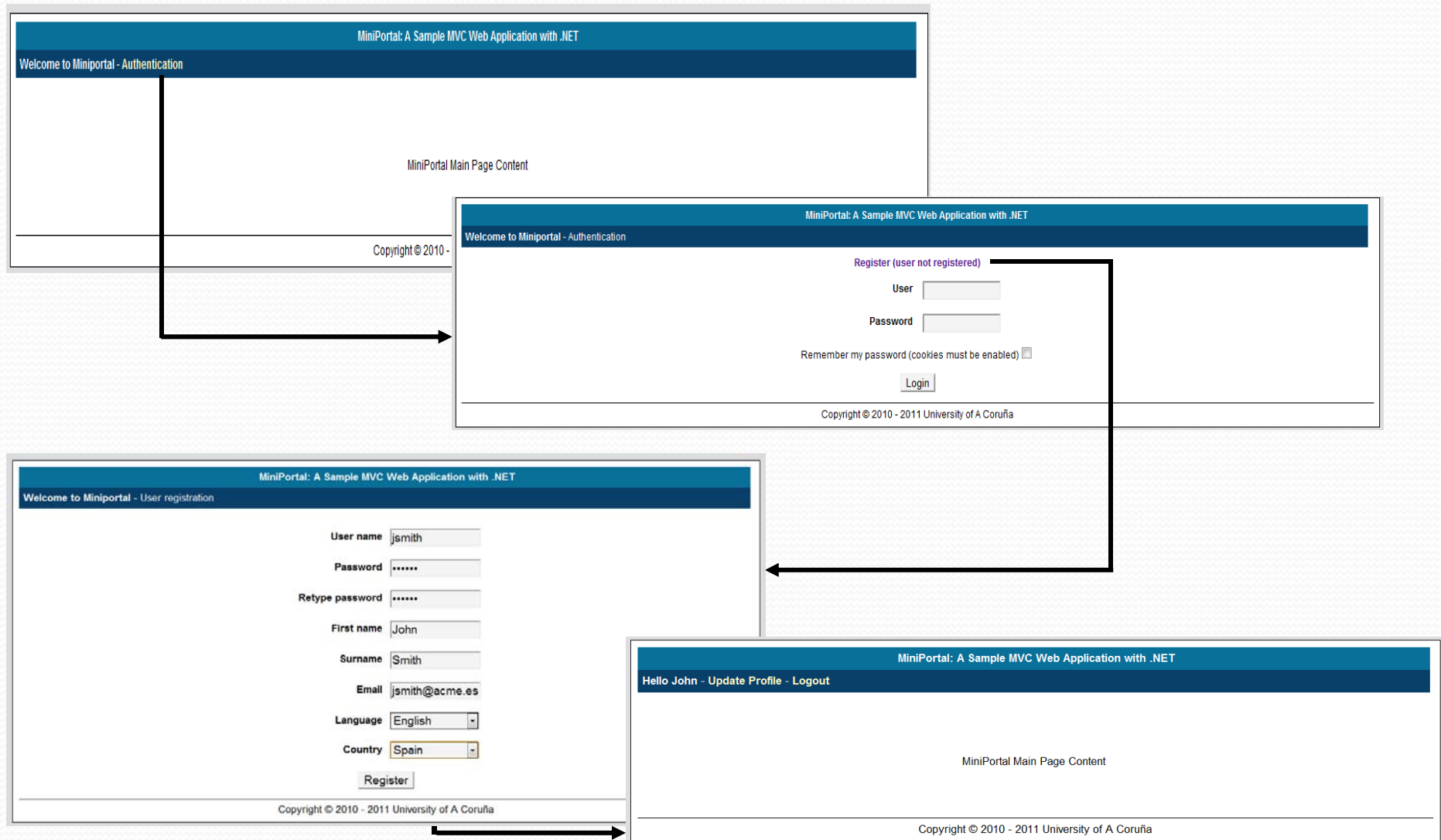
Diseño e implementación de aplicaciones Web con .NET



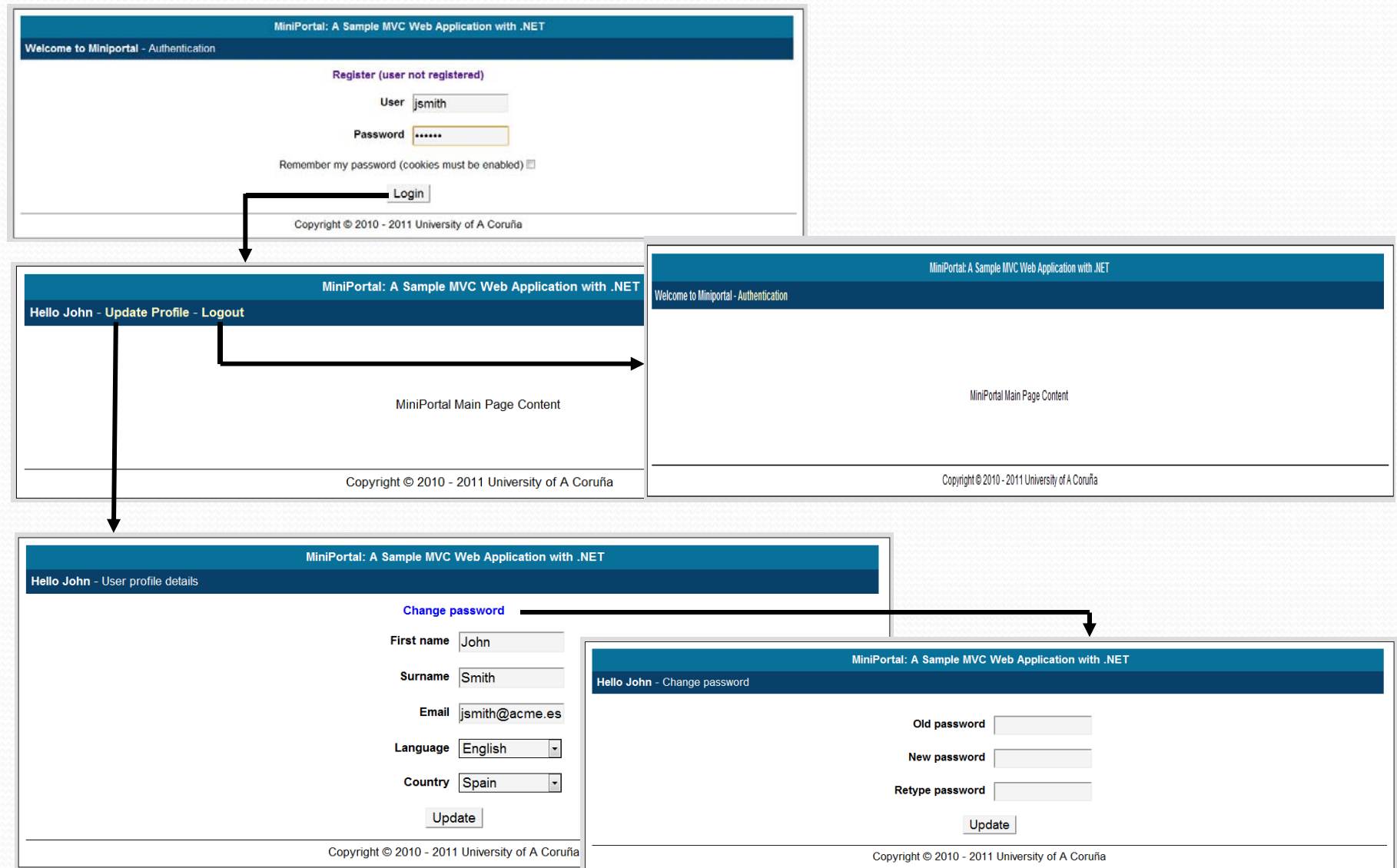
Contenido

- En este apartado estudiaremos el diseño e implementación de las capas controlador y vista de MiniPortal
 - Flujo de información entre páginas
 - Configuración de la aplicación Web
 - Página maestra de MiniPortal
 - Implementación de Autenticación y Salida

Flujo de Información entre Páginas



Flujo de Información entre Páginas





Contenido

- En este apartado estudiaremos el diseño e implementación de las capas controlador y vista de MiniPortal
 - Flujo de información entre páginas
 - Configuración de la aplicación Web
 - Página maestra de MiniPortal
 - Implementación de Autenticación y Salida

Configuración: Web.config (1 de 2)

```
<system.web>
```

```
<authentication mode="Forms">  
  <forms name=".ASPXAUTH"  
    loginUrl="~/Pages/User/Authentication.aspx"  
    timeout="30"  
    path="/"  
    defaultUrl="~/Pages/MainPage.aspx"  
    cookieless="AutoDetect"  />
```

En "desarrollo", usar
"UseUri"

```
</authentication>
```

```
<authorization>  
  <deny users="?" />  
</authorization>
```

Deniega acceso a
usuarios anónimos

```
<<...>>
```

```
</system.web>
```

Configuración: Web.config (2 de 2)

<<...>>

```
<location path="Pages/User/Register.aspx">
```

```
  <system.web>
```

```
    <authorization>
```

```
      <allow users="*" />
```

```
    </authorization>
```

```
  </system.web>
```

```
</location>
```

```
<location path="Pages/Errors/InternalServerError.aspx">
```

```
  <system.web>
```

```
    <authorization>
```

```
      <allow users="*" />
```

```
    </authorization>
```

```
  </system.web>
```

```
</location>
```

<<...>>

Permite acceso a
cualquier usuario



Contenido

- En este apartado estudiaremos el diseño e implementación de las capas controlador y vista de MiniPortal
 - Flujo de información entre páginas
 - Configuración de la aplicación Web
 - Página maestra de MiniPortal
 - Implementación de Autenticación y Salida

MiniPortal.master

- 4 secciones de contenido (ContentPlaceHolder):
 - ContentPlaceHolder_MenuWelcome: por defecto “Bienvenido a MiniPortal”. Si el usuario está autenticado: “Hola” + firstname
 - ContentPlaceHolder_MenuExplanation: por defecto vacío. Páginas específicas utilizan este espacio para incluir una breve explicación del contenido: Autenticación, Cambiar contraseña, Registro de usuario, ...
 - ContentPlaceHolder_MenuLinks: contiene enlaces para
 - a) Autenticación
 - b) Actualizar información de usuario y SalirSe muestran a) o b) en función de si el usuario está autenticado o no. Esto se controla en CB.
 - Las páginas específicas, que utilizan MenuExplanation, no utilizan MenuLinks (lo sobrescriben dejándolo como vacío)
 - ContentPlaceHolder_BodyContent: el contenido propio de cada página

MiniPortal.master

```
<%@ Master Language="C#" AutoEventWireup="true" Codebehind="Miniportal.master.cs"
    Inherits="Es.Udc.DotNet.MiniPortal.Web.Miniportal" %>
```

```
<html xmlns="http://www.w3.org/1999/xhtml">
<head id="Head1" runat="server">
    <link href="~/Css/Styles.css" rel="stylesheet" type="text/css" />
</head>

<body>
    <div id="window">
        <div id="pageTitle">
            <asp:Localize ID="lclPageTitle" runat="server"
                meta:resourcekey="lclPageTitle" />
        </div>

        <div id="menu">
            <span id="menuWelcome">
                <asp:ContentPlaceHolder ID="ContentPlaceHolder_MenuWelcome"
                    runat="server">
                    <asp:Label ID="lblWelcome" runat="server"
                        meta:resourcekey="lblWelcome" />
                </asp:ContentPlaceHolder>
            </span>
        </div>
    </div>
</body>
</html>
```

MiniPortal.master

```
<span id="menuExplanation">
  <asp:ContentPlaceHolder ID="ContentPlaceHolder_MenuExplanation"
    runat="server">
  </asp:ContentPlaceHolder>
</span>
<span id="menuLinks">
  <asp:ContentPlaceHolder ID="ContentPlaceHolder_MenuLinks"
    runat="server">
    <asp:Label ID="lblDash1" runat="server" Text="-" />
    <asp:HyperLink ID="lnkAuthenticate" runat="server"
      NavigateUrl="~/Pages/User/Authentication.aspx"
      meta:resourcekey="lnkAuthenticate" />
    <asp:Label ID="lblDash2" runat="server" Text="-" />
    <asp:HyperLink ID="lnkUpdate" runat="server"
      NavigateUrl="~/Pages/User/UpdateUserProfile.aspx"
      meta:resourcekey="lnkUpdate" />
    <asp:Label ID="lblDash3" runat="server" Text="-" />
    <asp:HyperLink ID="lnkLogout" runat="server"
      NavigateUrl="~/Pages/User/Logout.aspx"
      meta:resourcekey="lnkLogout" />
  </asp:ContentPlaceHolder>
</span>
</div>
```

MiniPortal.master

```
<div id="content">
    <asp:ContentPlaceHolder ID="ContentPlaceHolder_BodyContent"
        runat="server" />
</div>

<div id="footer">
    <asp:Localize ID="lclFooter" runat="server"
        meta:resourcekey="lclFooter" />
</div>

</div>

</body>
</html>
```

MiniPortal.master.cs

```
protected void Page_Load(object sender, EventArgs e)
{
    if (!SessionManager.IsUserAuthenticated(Context))
    {
        if (lblDash2 != null)
            lblDash2.Visible = false;
        if (lnkUpdate != null)
            lnkUpdate.Visible = false;
        if (lblDash3 != null)
            lblDash3.Visible = false;
        if (lnkLogout != null)
            lnkLogout.Visible = false;
    }
    else
    {
        if (lblWelcome != null)
            lblWelcome.Text =
                GetLocalResourceObject("lblWelcome.Hello.Text").ToString()
                + " " + SessionManager.GetUserSession(Context).FirstName;
        if (lblDash1 != null)
            lblDash1.Visible = false;
        if (lnkAuthenticate != null)
            lnkAuthenticate.Visible = false;
    }
}
```



Contenido

- En este apartado estudiaremos el diseño e implementación de las capas controlador y vista de
 - Flujo de información entre páginas
 - Configuración de la aplicación Web
 - Página maestra de MiniPortal
 - Implementación de Autenticación y Salida

MainPage.aspx

Usuario no Autenticado

MiniPortal: A Sample MVC Web Application with .NET

Welcome to Miniportal - Authentication

MiniPortal Main Page Content

Copyright © 2010 - 2011 University of A Coruña

Authentication.aspx

Usuario no Autenticado

MiniPortal: A Sample MVC Web Application with .NET

Welcome to Miniportal - Authentication

Register (user not registered)

User

Password

Remember my password (cookies must be enabled) ☐

Login

Copyright © 2010 - 2011 University of A Coruña

Authentication.aspx

Usuario no Autenticado

```
protected void BtnLoginClick(object sender, EventArgs e)
{
    if (Page.IsValid)
    {
        try
        {
            SessionManager.Login(Context, txtLogin.Text,
                                txtPassword.Text, checkRememberPassword.Checked);

            FormsAuthentication.
                RedirectFromLoginPage(txtLogin.Text,
                                    checkRememberPassword.Checked);
        }
        catch (InstanceNotFoundException)
        {
            lblLoginError.Visible = true;
        }
        catch (IncorrectPasswordException)
        {
            lblPasswordError.Visible = true;
        }
    }
}
```

Es.Udc.DotNet.MiniPortal.HTTP.Session.SessionManager.cs

```
public static void Login(HttpContext context, String loginName,
    String clearPassword, Boolean rememberMyPassword)
{
    /* Try to login, and if successful, update session with the necessary
     * objects for an authenticated user. */
    LoginResult loginResult = DoLogin(context, loginName,
        clearPassword, false, rememberMyPassword);

    /* Add cookies if requested. */
    if (rememberMyPassword)
    {
        CookiesManager.LeaveCookies(context, loginName,
            loginResult.EncryptedPassword);
    }
}
```

Es.Udc.DotNet.Miniportal.HTTP.Session.SessionManager.cs

```
private static LoginResult DoLogin(HttpContext context, String loginName,
                                   String password, Boolean passwordIsEncrypted,
                                   Boolean rememberMyPassword)
{
    LoginResult loginResult =
        userService.Login(loginName, password, passwordIsEncrypted);

    /* Insert necessary objects in the session. */
    UserSession userSession = new UserSession();
    userSession.UserProfileId = loginResult.UserProfileId;
    userSession.FirstName = loginResult.FirstName;

    Locale locale =
        new Locale(loginResult.Language, loginResult.Country);

    UpdateSessionForAuthenticatedUser(context, userSession, locale);

    return loginResult;
}
```

Es.Udc.DotNet.Miniportal.HTTP.Util.CookiesManager.cs

```
public static void LeaveCookies(HttpContext context, String loginName,
    String encryptedPassword) {

    int timeToLive = REMEMBER_MY_PASSWORD_AGE;

    /* Create the loginName cookie. */
    HttpCookie loginNameCookie = new HttpCookie(LOGIN_NAME_COOKIE, loginName);

    /* Create the encryptedPassword cookie. */
    //...

    /* Set maximum age to cookies. */
    loginNameCookie.Expires = DateTime.Now.AddSeconds(timeToLive);
    //...

    /* Add cookies to response. */
    context.Response.Cookies.Add(loginNameCookie);
    context.Response.Cookies.Add(encryptedPasswordCookie);
}
```

MiniPortal

SessionManager: logout

```
/// <summary>
/// Destroys the session, and removes the cookies if the user had
/// selected "remember my password".
/// </summary>
/// <param name="context">Http Context includes request, response, etc.</param>
public static void Logout(HttpContext context)
{
    /* Remove cookies. */
    CookiesManager.RemoveCookies(context);

    /* Invalidate session. */
    context.Session.Abandon();

    /* Invalidate Authentication Ticket */
    FormsAuthentication.SignOut();
}
```