Gora LEYE ~ Expert C#.NET

How to configure Custom Membership Provider using ASP.NET MVC4 with external login like facebook, yahoo , google or other relying party accounts.

28 Mercredi août 2013

PUBLIÉ PAR GORALEYE IN ARCHITECTURE, ASP.NET, C#.NET, EXTERNAL LOGIN, MEMBERSHIP, OAUTH PROVIDERS, ROLE PROVIDER, SECURITY

≈ 3 COMMENTAIRES

Tags
Application Services, Architecture, ASP.NET, C#.NET, Custom Membership Provider, OAuth Providers, Role provider, Security

ASP.NET membership is designed to enable you to easily use a number of different membership providers for your ASP.NET applications. You can use the supplied membership providers that are included with the .NET Framework, or you can implement your own providers.

There are two primary reasons for creating a custom membership provider.

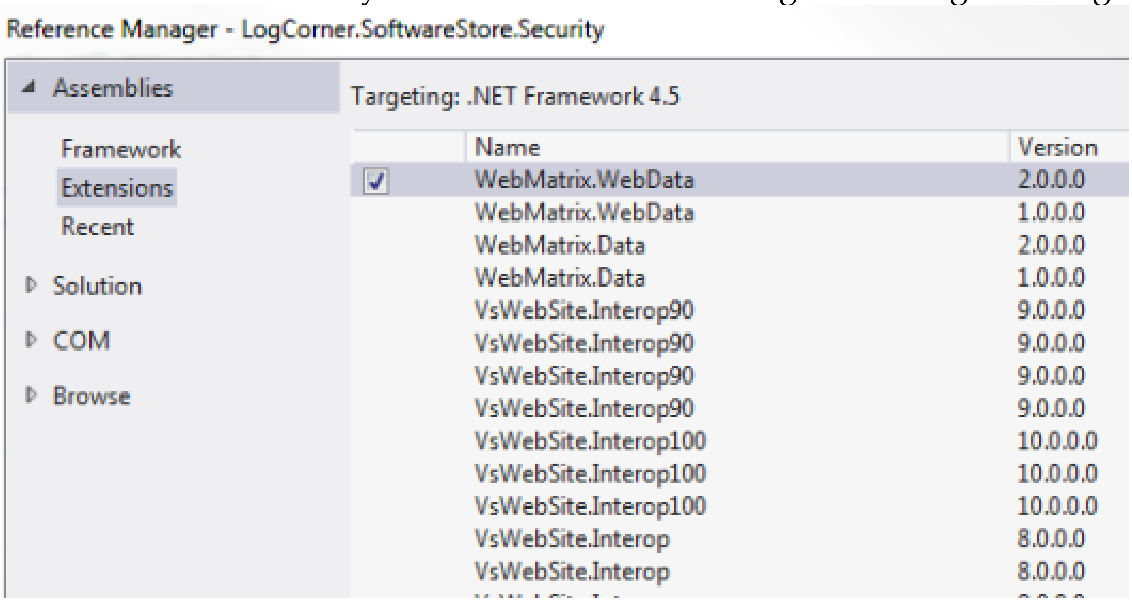
- You need to store membership information in a data source that is not supported by the membership providers included with the .NET Framework, such as a MysQL database, an Oracle database, or other data sources.
- You need to manage membership information using a database schema that is different from the database schema used by the providers that ship with the .NET Framework. A common example of this would be membership data that already exists in a SQL Server database for a company or Web site.

In tis tutorial, we are going to implement and configure a custom Membership Provider using ASP.NET MVC4 that enable external login like facebook, yahoo , google or other relying party accounts.

Let's go

A. Create a Custom MemberShip Application

class Library

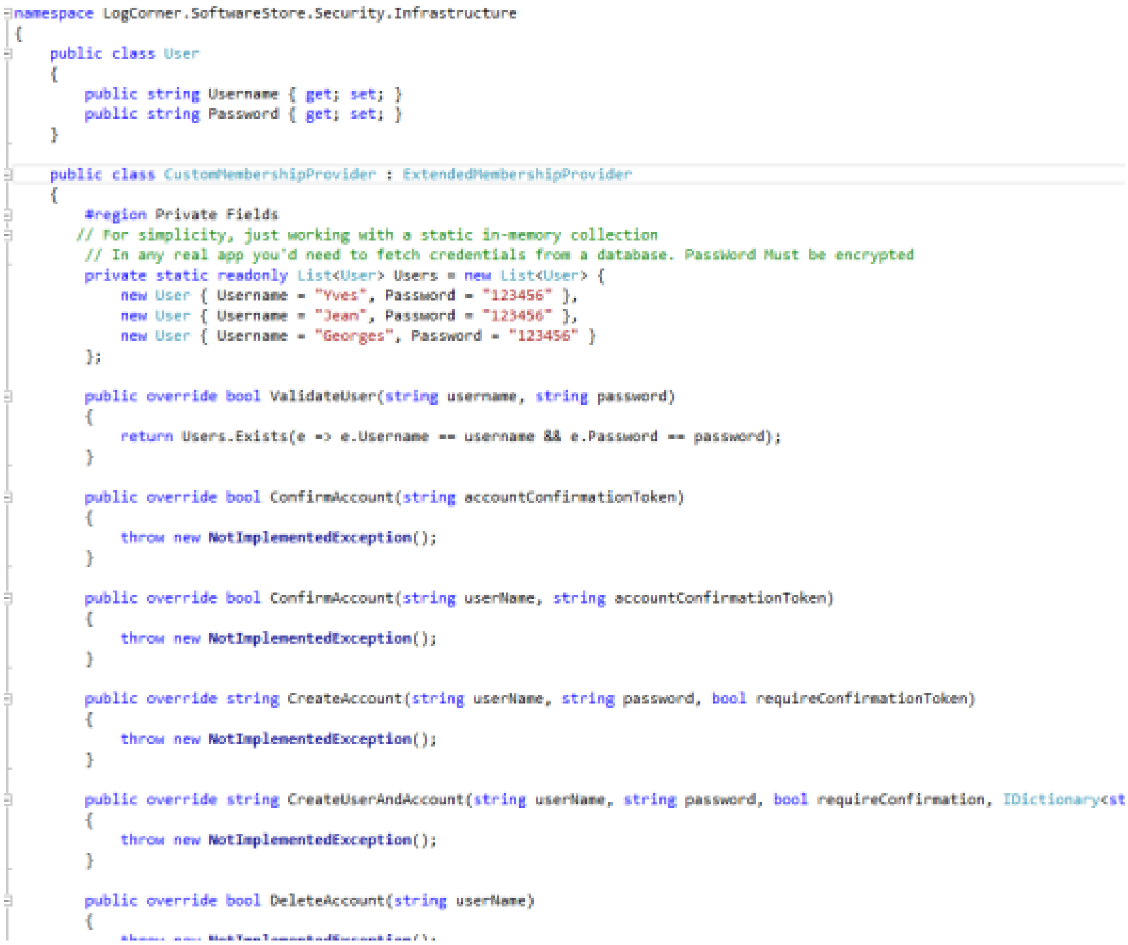
1. Create a class Library Project (our sample Projet name is **LogCorner.SoftwareStore.Security**)
2. Reference the assembly **System.Web.ApplicationServices** (Right Click Reference => Add reference and Select Assemblies => navigate to System.Web.ApplicationServices and add it)
3. Create a Class **CustomMembershipProvider** . Here we are going to enable external login like facebook, yahoo , google or other relying party accounts.So derive our custom class from **ExtendedMembershipProvider** whose base class is **MembershipProvider**
4. So reference the assembly **WebMatrix.WebData** or get it via nuget Packages
5. 

Reference Manager - LogCorner.SoftwareStore.Security

Targeting: .NET Framework 4.5

	Name	Version
☑	WebMatrix.WebData	2.0.0.0
	WebMatrix.WebData	1.0.0.0
	WebMatrix.Data	2.0.0.0
	WebMatrix.Data	1.0.0.0
	VsWebSite.Interop90	9.0.0.0
	VsWebSite.Interop90	9.0.0.0
	VsWebSite.Interop90	9.0.0.0
	VsWebSite.Interop90	9.0.0.0
	VsWebSite.Interop100	10.0.0.0
	VsWebSite.Interop100	10.0.0.0
	VsWebSite.Interop100	10.0.0.0
	VsWebSite.Interop	8.0.0.0
	VsWebSite.Interop	8.0.0.0

6. Override ValidateUser as follow

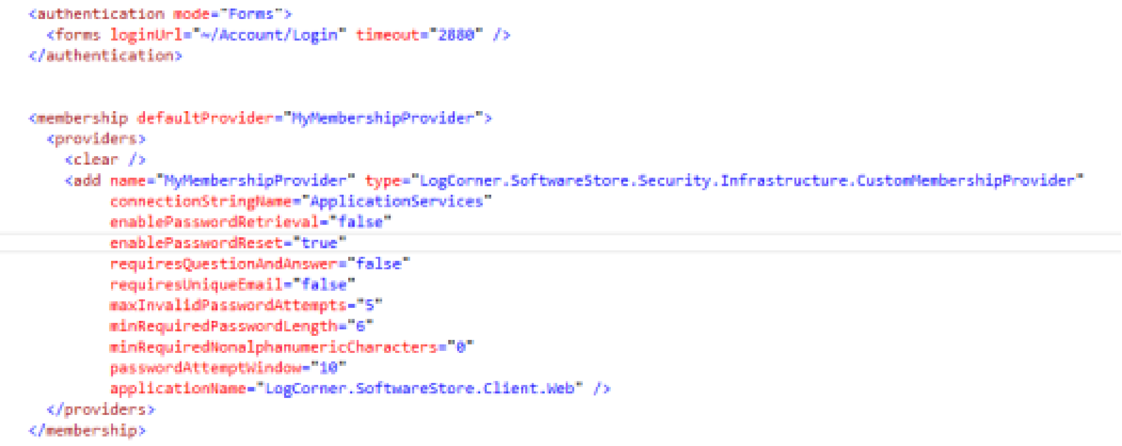


```
namespace LogCorner.SoftwareStore.Security.Infrastructure
{
    public class User
    {
        public string Username { get; set; }
        public string Password { get; set; }
    }

    public class CustomMembershipProvider : ExtendedMembershipProvider
    {
        #region Private Fields
      // For simplicity, just working with a static in-memory collection
        // In any real app you'd need to fetch credentials from a database. PassWord Must be encrypted
        private static readonly List<User> Users = new List<User> {
            new User { Username = "Yves", Password = "123456" },
            new User { Username = "Jean", Password = "123456" },
            new User { Username = "Georges", Password = "123456" }
        };

        public override bool ValidateUser(string username, string password)
        {
            return Users.Exists(e => e.Username == username && e.Password == password);
        }

        public override bool ConfirmAccount(string accountConfirmationToken)
        {
            throw new NotImplementedException();
        }

        public override bool ConfirmAccount(string userName, string accountConfirmationToken)
        {
            throw new NotImplementedException();
        }

        public override string CreateAccount(string userName, string password, bool requireConfirmationToken)
        {
            throw new NotImplementedException();
        }

        public override string CreateUserAndAccount(string userName, string password, bool requireConfirmation, IDictionary<st
        {
            throw new NotImplementedException();
        }

        public override bool DeleteAccount(string userName)
        {
            throw new NotImplementedException();
```

For now we have what we need for our application security. To go further in the implementation of Custom Membership Provider, please see our tutorial *Mastering Custum ASP.NET MemberShip Provider using ASP.NET MVC*

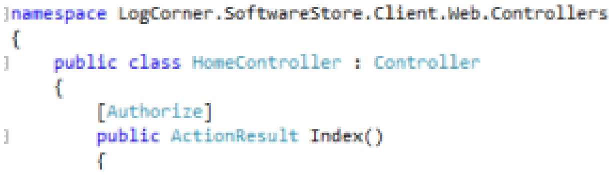
B. Create an ASP.NET MVC4 application Client

1. Create an ASP.NET MVC4 application Client (Add New projet è ASP.NET MVC4 Web Application è Select Template Internet Web Appliction and Click OK)
2. Open Web.config file
3. Add or Replace membership section as follow



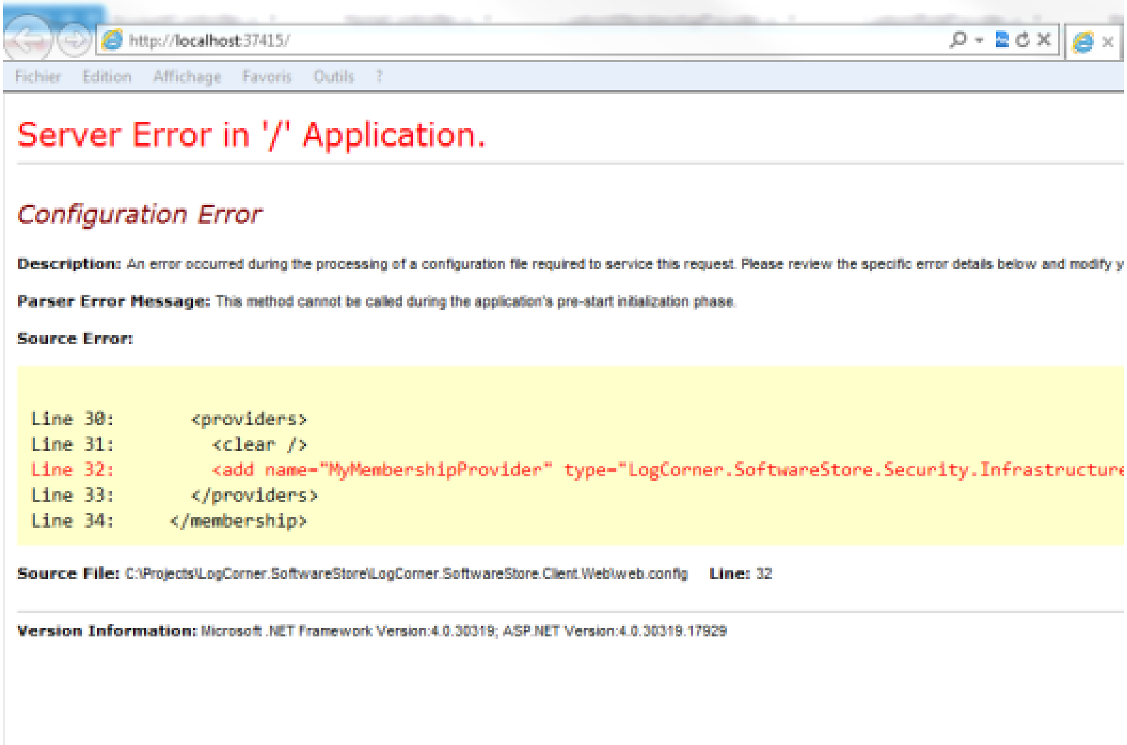
```
<authentication mode="Forms">
  <forms loginUrl="~/Account/Login" timeout="2880" />
</authentication>


<membership defaultProvider="MyMembershipProvider">
  <providers>
    <clear />
    <add name="MyMembershipProvider" type="LogCorner.SoftwareStore.Security.Infrastructure.CustomMembershipProvider"
         connectionStringName="ApplicationServices"
         enablePasswordRetrieval="false"
         enablePasswordReset="true"
         requiresQuestionAndAnswer="false"
         requiresUniqueEmail="false"
         maxInvalidPasswordAttempts="5"
         minRequiredPasswordLength="6"
         minRequiredNonalphanumericCharacters="0"
         passwordAttemptWindow="10"
         applicationName="LogCorner.SoftwareStore.Client.Web" />
  </providers>
</membership>
```

4. Open **HomeController** and **Authorize** Attribute to Index **ActionResult**



```
namespace LogCorner.SoftwareStore.Client.Web.Controllers
{
    public class HomeController : Controller
    {
        [Authorize]
        public ActionResult Index()
        {
```

5. Run the application ASP.NET MVC4 application Client, you ll have the errors below



Server Error in '/' Application.

Configuration Error

Description: An error occurred during the processing of a configuration file required to service this request. Please review the specific error details below and modify y

Parser Error Message: This method cannot be called during the application's pre-start initialization phase.

Source Error:

```
Line 30:       <providers>
Line 31:         <clear />
Line 32:         <add name="MyMembershipProvider" type="LogCorner.SoftwareStore.Security.Infrastructure
Line 33:       </providers>
Line 34:     </membership>
```

Source File: C:\Projects\LogCorner.SoftwareStore\LogCorner.SoftwareStore.Client.Web\web.config **Line:** 32

Version Information: Microsoft .NET Framework Version:4.0.30319; ASP.NET Version:4.0.30319.17929

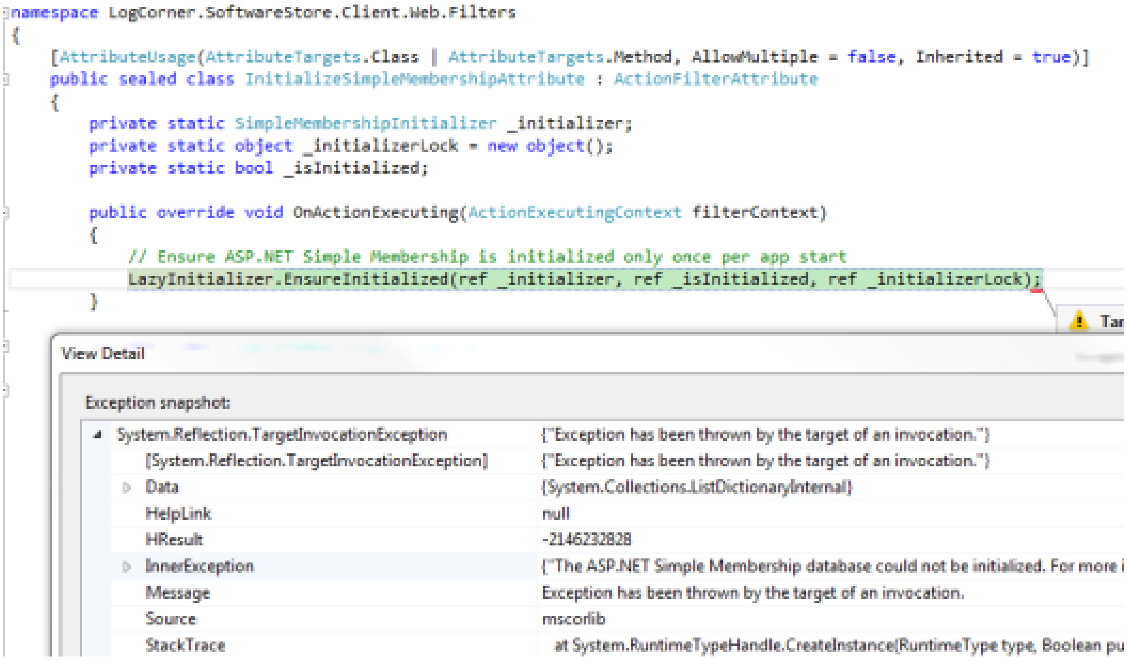
6. do not panic, proceed as follows :

Add this in your web.config (in the appSettings section):

<add key="enableSimpleMembership" value="false"/>

<add key="autoFormsAuthentication" value="false"/>

7. Run the application ASP.NET MVC4 application Client, you ll have another error



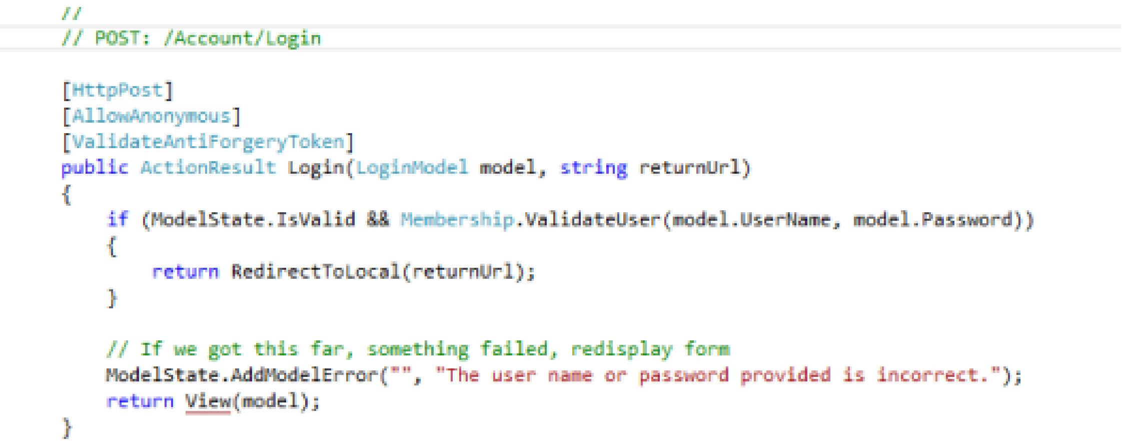
```
namespace LogCorner.SoftwareStore.Client.Web.Filters
{
    [AttributeUsage(AttributeTargets.Class | AttributeTargets.Method, AllowMultiple = false, Inherited = true)]
    public sealed class InitializeSimpleMembershipAttribute : ActionFilterAttribute
    {
        private static SimpleMembershipInitializer _initializer;
        private static object _initializerLock = new object();
        private static bool _isInitialized;

        public override void OnActionExecuting(ActionExecutingContext filterContext)
        {
            // Ensure ASP.NET Simple Membership is initialized only once per app start
            LazyInitializer.EnsureInitialized(ref _initializer, ref _isInitialized, ref _initializerLock);
        }
```

Exception snapshot:

System.Reflection.TargetInvocationException	{"Exception has been thrown by the target of an invocation."}
[System.Reflection.TargetInvocationException]	{"Exception has been thrown by the target of an invocation."}
Data	{System.Collections.ListDictionaryInternal}
HelpLink	null
HResult	-2146232828
InnerException	{"The ASP.NET Simple Membership database could not be initialized. For more i
Message	Exception has been thrown by the target of an invocation.
Source	mscorlib
StackTrace	at System.RuntimeTypeHandle.CreateInstance(RuntimeType type, Boolean pu

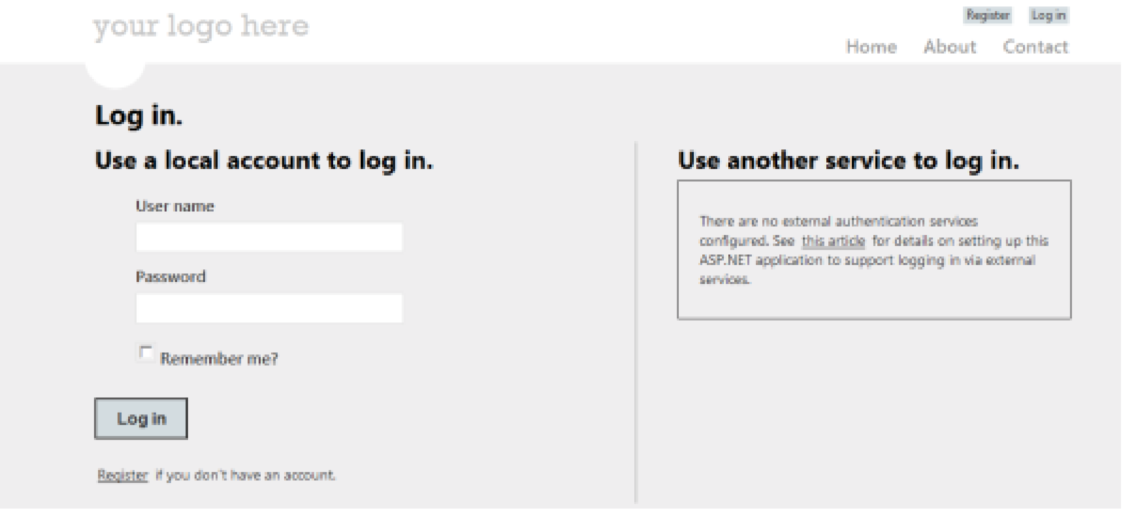
8. To fix it Open **AccountController** and comment **InitializeSimpleMembership** , because we using Custom Membership Provider instead of Simple Membership

9. Override **Login** Action of **AccountController** as follow :



```
    //
    // POST: /Account/Login

    [HttpPost]
    [AllowAnonymous]
    [ValidateAntiForgeryToken]
    public ActionResult Login(LoginModel model, string returnUrl)
    {
        if (ModelState.IsValid && Membership.ValidateUser(model.UserName, model.Password))
        {
            return RedirectToLocal(returnUrl);
        }

        // If we got this far, something failed, redisplay form
        ModelState.AddModelError("", "The user name or password provided is incorrect.");
        return View(model);
    }
```

10. Run the application ASP.NET MVC4 application Client, you'll have the form authentication below. Note that external logins are disabled. Will going to configure external logins further



Log in.

Use a local account to log in.

User name

Password

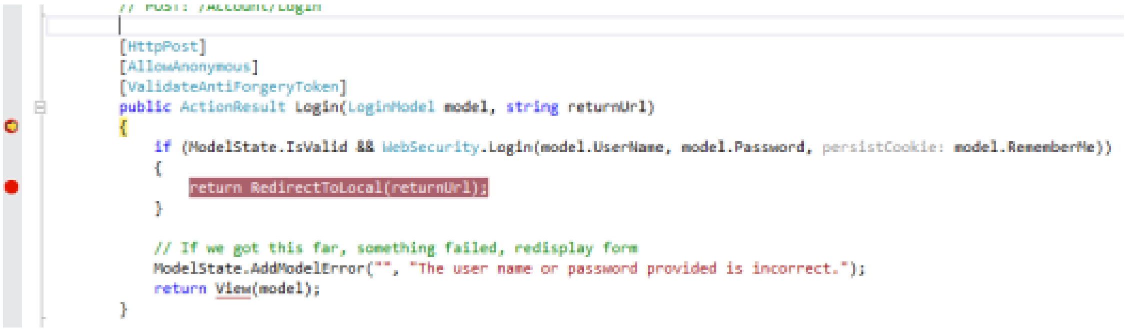
□ Remember me?

Register if you don't have an account.

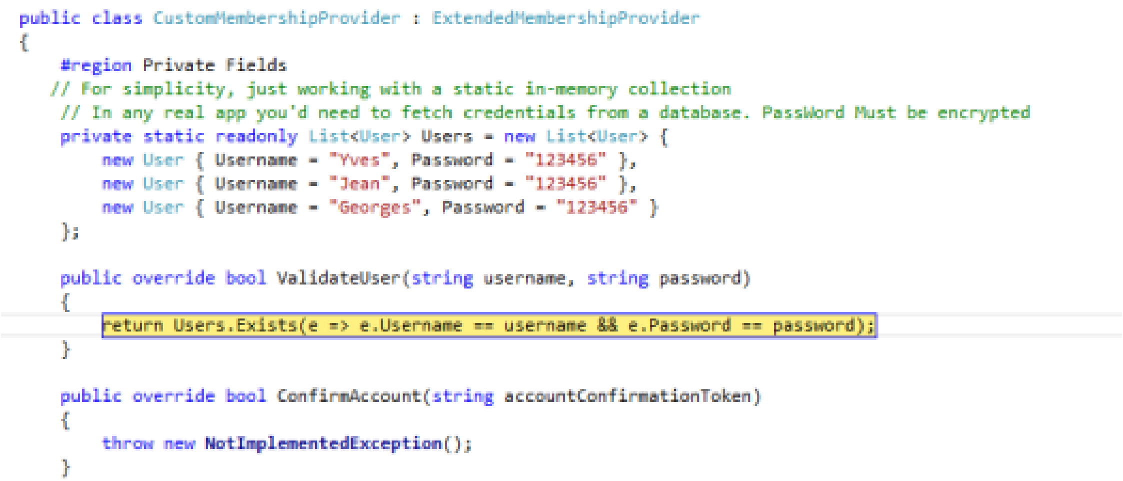
Use another service to log in.

There are no external authentication services configured. See this article for details on setting up this ASP.NET application to support logging in via external services.

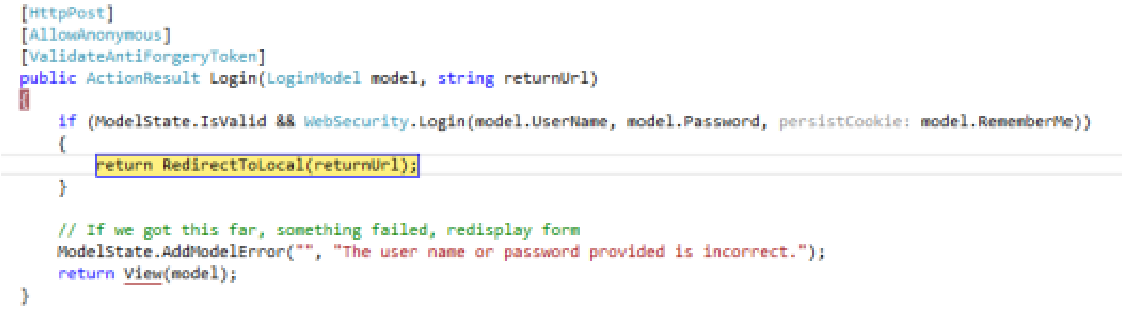
11. Enter user credentials and click Log In, then you will have the execution workflow below :



```
        // POST: /Account/Login

        [HttpPost]
        [AllowAnonymous]
        [ValidateAntiForgeryToken]
        public ActionResult Login(LoginModel model, string returnUrl)
        {
            if (ModelState.IsValid && WebSecurity.Login(model.UserName, model.Password, persistCookie: model.RememberMe))
            {
                return RedirectToLocal(returnUrl);
            }

            // If we got this far, something failed, redisplay form
            ModelState.AddModelError("", "The user name or password provided is incorrect.");
            return View(model);
        }
```



```
public class CustomMembershipProvider : ExtendedMembershipProvider
{
    #region Private Fields
   // For simplicity, just working with a static in-memory collection
    // In any real app you'd need to fetch credentials from a database. PassWord Must be encrypted
    private static readonly List<User> Users = new List<User> {
        new User { Username = "Yves", Password = "123456" },
        new User { Username = "Jean", Password = "123456" },
        new User { Username = "Georges", Password = "123456" }
    };

    public override bool ValidateUser(string username, string password)
    {
        return Users.Exists(e => e.Username == username && e.Password == password);
    }

    public override bool ConfirmAccount(string accountConfirmationToken)
    {
        throw new NotImplementedException();
    }
```



```
[HttpPost]
[AllowAnonymous]
[ValidateAntiForgeryToken]
public ActionResult Login(LoginModel model, string returnUrl)
{
    if (ModelState.IsValid && WebSecurity.Login(model.UserName, model.Password, persistCookie: model.RememberMe))
    {
        return RedirectToLocal(returnUrl);
    }

    // If we got this far, something failed, redisplay form
    ModelState.AddModelError("", "The user name or password provided is incorrect.");
    return View(model);
}
```

B. Configuration Custom Membership Provider using ASP.NET MVC4 with external login like facebook, yahoo , google or other relying party accounts.

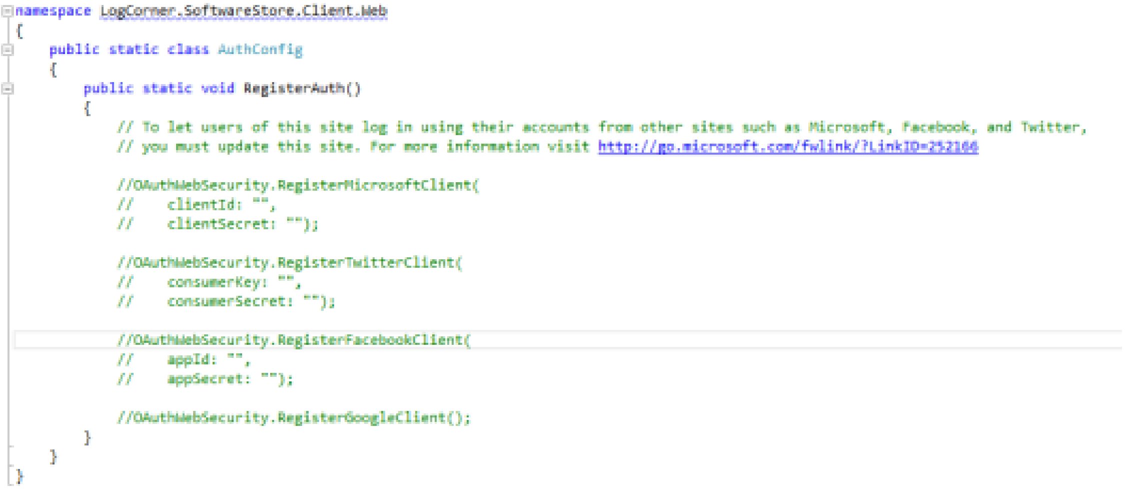
1. **Registering with an external provider**

To authenticate users with credentials from an external provider, you must register your web site with the provider. When you register your site, you will receive the parameters (such as key or id, and secret) to include when registering the client. You must have an account with the providers you wish to use.

To successfully register your site, follow the instructions provided on this sites :

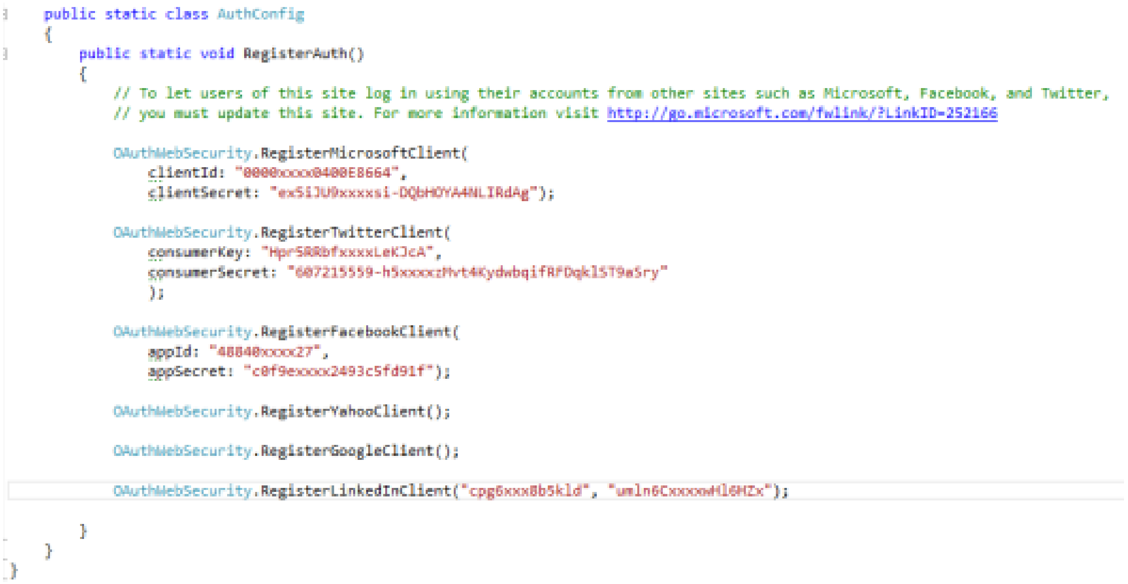
- FaceBook developer
- Google Developer
- Microsoft developer
- Twitter developer

2. navigation to App_Start of your web application ,locate AuthConfig.cs file and open it



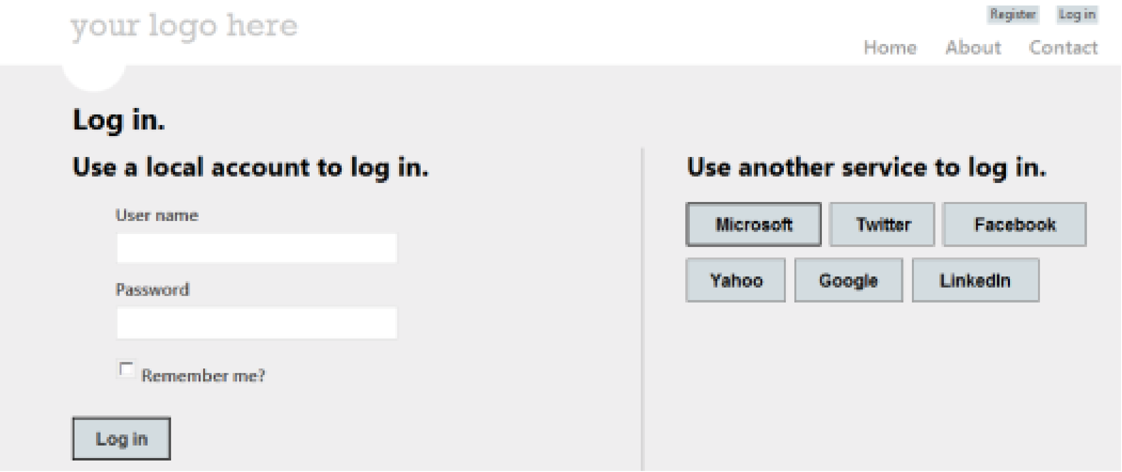
```
namespace LogCorner.SoftwareStore.Client.Web
{
    public static class AuthConfig
    {
        public static void RegisterAuth()
        {
            // To let users of this site log in using their accounts from other sites such as Microsoft, Facebook, and Twitter,
            // you must update this site. For more information visit http://go.microsoft.com/fwlink/?LinkID=252166

            //OAuthWebSecurity.RegisterMicrosoftClient(
            //    clientId: "",
            //    clientSecret: "");

            //OAuthWebSecurity.RegisterTwitterClient(
            //    consumerKey: "",
            //    consumerSecret: "");

            //OAuthWebSecurity.RegisterFacebookClient(
            //    appId: "",
            //    appSecret: "");

            //OAuthWebSecurity.RegisterGoogleClient();
        }
    }
}
```

3. Update **RegisterAuth** method with parameters (such as key or id, and secret) you have received from facebook, twitter, google, yahoo. etc……

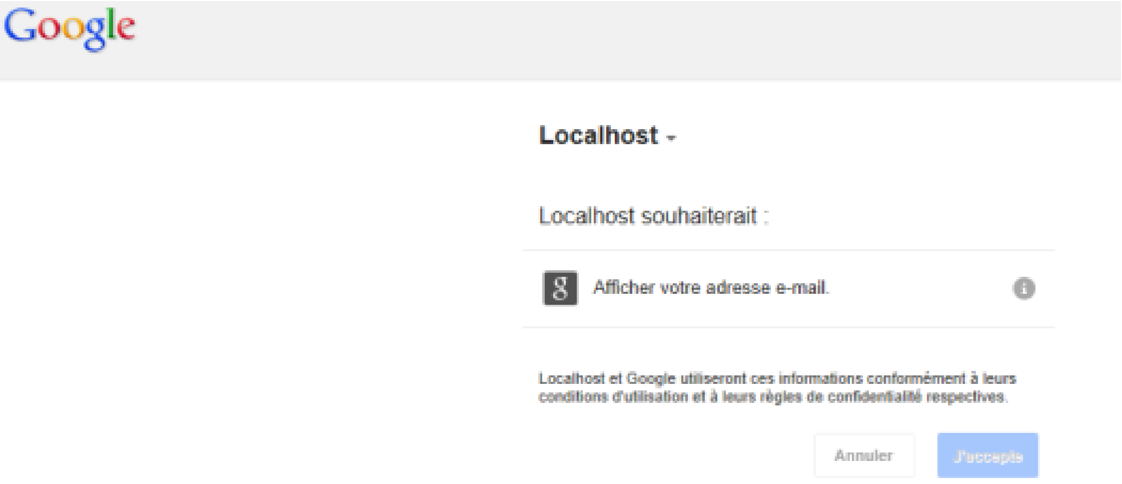


```
    public static class AuthConfig
    {
        public static void RegisterAuth()
        {
            // To let users of this site log in using their accounts from other sites such as Microsoft, Facebook, and Twitter,
            // you must update this site. For more information visit http://go.microsoft.com/fwlink/?LinkID=252166

            OAuthWebSecurity.RegisterMicrosoftClient(
                clientId: "0000xxxx0400E8664",
                clientSecret: "exSiJU9xxxxsi-DQbHOYA4NLIRdAg");

            OAuthWebSecurity.RegisterTwitterClient(
                consumerKey: "HprSRRbfxxxxLeKJcA",
                consumerSecret: "607215559-h5xxxxzMvt4KydwbqifRFDqklST9aSry"
                );

            OAuthWebSecurity.RegisterFacebookClient(
                appId: "48840xxxx27",
                appSecret: "c0f9exxxx2493c5fd91f");

            OAuthWebSecurity.RegisterYahooClient();

            OAuthWebSecurity.RegisterGoogleClient();

            OAuthWebSecurity.RegisterLinkedInClient("cpg6xxx8b5kld", "umln6CxxxxwHl6HZx");

        }
    }
}
```

4. Next run your web application, then your logon form must be updated with external logins



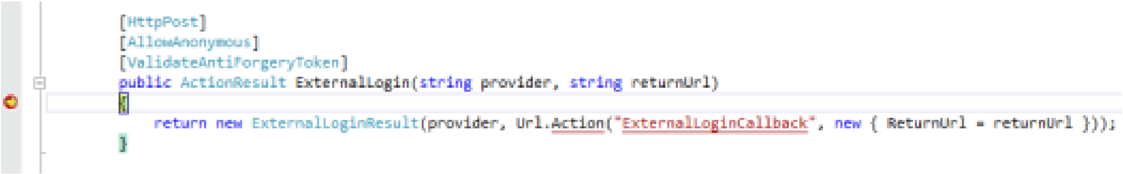
5. Click on Google button (for example)



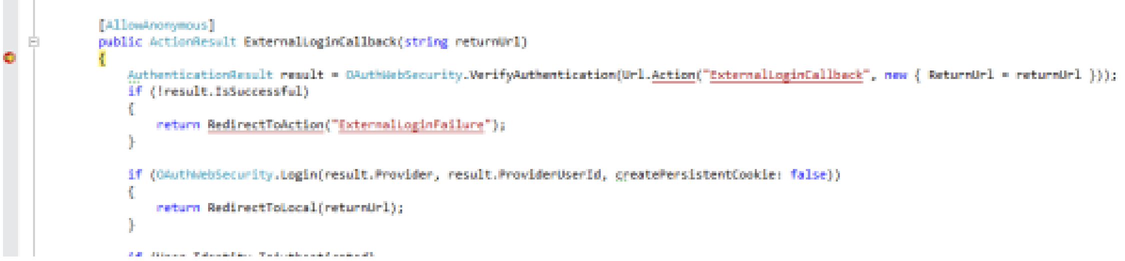
Localhost souhaiterait :

Afficher votre adresse e-mail.

Localhost et Google utiliseront ces informations conformément à leurs conditions d'utilisation et à leurs règles de confidentialité respectives.



```
[HttpPost]
[AllowAnonymous]
[ValidateAntiForgeryToken]
public ActionResult ExternalLogin(string provider, string returnUrl)
{
    return new ExternalLoginResult(provider, Url.Action("ExternalLoginCallback", new { ReturnUrl = returnUrl }));
}
```



```
[AllowAnonymous]
public ActionResult ExternalLoginCallback(string returnUrl)
{
    AuthenticationResult result = OAuthWebSecurity.VerifyAuthentication(Url.Action("ExternalLoginCallback", new { ReturnUrl = returnUrl }));
    if (!result.IsSuccessful)
    {
        return RedirectToAction("ExternalLoginFailure");
    }

    if (OAuthWebSecurity.Login(result.Provider, result.ProviderUserId, createPersistentCookie: false))
    {
        return RedirectToLocal(returnUrl);
    }

    if (User.Identity.IsAuthenticated)
```

If you seek information about encoding and decoding password, please read our article ASP.NET Custom Membership Password Encoding and Decoding based on key SALT using SHA-3 algorithm

We have at the end of this tutorial, thank you for feedbacks.

réflexions à propos de “ How to configure Custom Membership Provider using ASP.NET MVC4 with external login like facebook, yahoo , google or other relying party accounts. ”

1. Ping: How to configure Custom Membership Provider using ASP.NET MVC4 | Gora LEYE
2. Ping: How to configure Custom Membership and Role Provider using ASP.NET MVC4 | Gora LEYE
3. Ping: ASP.NET MVC Tracking User Activity | Gora LEYE

3