|  |  |
| --- | --- |
| Título: | Addins para visual studio Fwk Software Factory |
| Autor | Marcelo Oviedo |
| Versión | 1.0 |
| Versión Fwk | 5.0 |
| Fecha de confección: | 14/01/2011 |

**Contenido**

**Proyects wizards**

Este addin por el momento permite por un lado generar todos los proyectos del BackEnd

**Items wizards:**

Se trata de una serie de ítems templates que permiten generar.

Entidades

Request

Responses

Services

DAC

~~SP:~~

**Instalación**

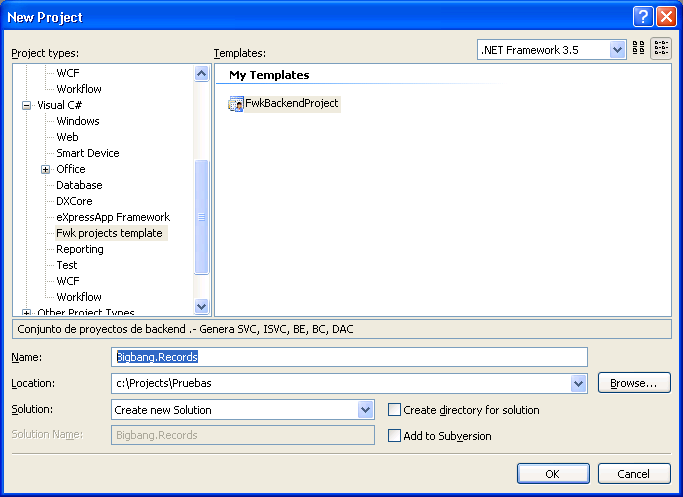
Ejecutar Fwk.Wizard.Setup.msi y seguir el asistente dando next, next, etc

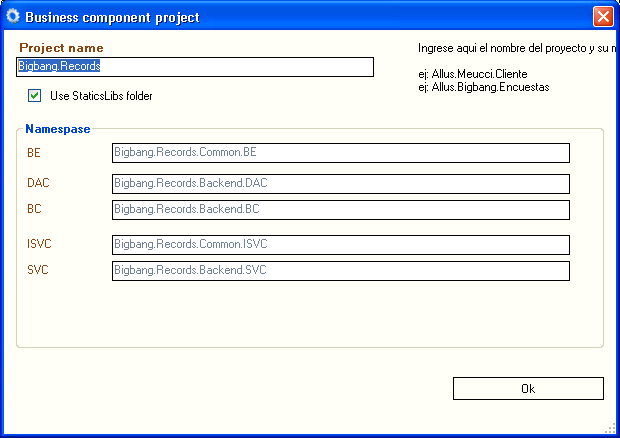
Dirigirse a la carpeta %programfiles%\Allus Global BPO\FwkTemplates

Ejecutar el archivo **Install.bat**

Uso:

1. Cree un nuevo proyecto en el Visual Studio 2008
2. Ubique el nodo Fwk Projects Template en el árbol izquierdo dentro del nodo Visual C#
3. Seleccione FwkBackEndProject
4. Siga el asistente





Nota: El nombre del proyecto es el nombre del proyecto de visual studio .- Generalmente esta bajo el siguiente patrón;

[Sistema][Dominio]

Con lo cual nos quedan los namespases:

* [Sistema][Dominio]BackEnd.BC
* [Sistema][Dominio]BackEnd.SVC
* [Sistema][Dominio]BackEnd.DAC
* [Sistema][Dominio]Common.ISVC
* [Sistema][Dominio]Common.BE

**Code generator**

Se trata de una aplicación externa que permite generar clases store procedures que utiliza cualquier desarrollo que se base en la arquitectura fwk.-

El bloque Fwk.Security cubre los siguientes ámbitos de seguridad:

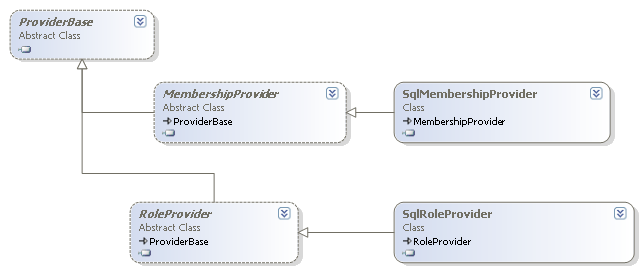
* Memberships y Roles
* LDAP

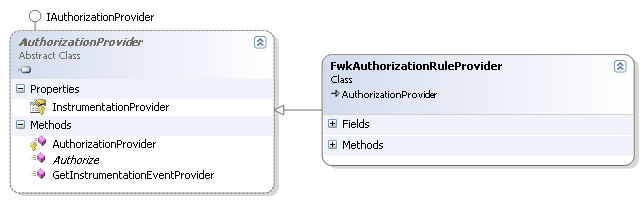
FwkMembership

Este componente actúa como wrapper entre las Memberships, RolManager y la Security Application Blocks

FwkMembership se configura por medio del modelo de proveedores (providers). Los providers hacen extensible el desarrollo de código ya que facilitan diseñar sus propios proveedores.- La Memberships, RolManager y el modelo de autorización siguen el patrón de providers.

La Fwkmemberships internamente está desarrollada para trabajar con providers ya definidos y se pueden ver en el siguiente grafico:





**SqlMembershipProvider**

Administra el almacenamiento de información de suscripciones de usuario para una aplicación en una base de datos de SQL Server.

Dada esta cadena de conexión

<add name=" **LocalSqlServer** "

    connectionString="data source=.\SQLEXPRESS;Integrated Security=SSPI;AttachDBFilename=|DataDirectory|aspnetdb.mdf;User Instance=true"

    providerName="System.Data.SqlClient" />

Podríamos tener este proveedor:

<membership>

  <providers>

    <add

      name="**bigbang**"

      type="**System.Web.Security.SqlMembershipProvider, ..."**

      connectionStringName="**bigbangAmex** "

      enablePasswordRetrieval="false"

      enablePasswordReset="true"

      requiresQuestionAndAnswer="true"

      applicationName="**bigbangAmex** "

      requiresUniqueEmail="false"

      passwordFormat="Hashed"

      maxInvalidPasswordAttempts="5"

      minRequiredPasswordLength="7"

      minRequiredNonalphanumericCharacters="1"

      passwordAttemptWindow="10"

      passwordStrengthRegularExpression=""

    />

  </providers>

</membership>

**SqlRolProvdider**

Las clases [Roles](http://msdn.microsoft.com/es-es/library/system.web.security.roles(v=VS.80).aspx) y [RolePrincipal](http://msdn.microsoft.com/es-es/library/system.web.security.roleprincipal(v=VS.80).aspx) que se utilizan dentro de FwkMembership, emplean SqlRolProvdider

para proporcionar servicios de administración de funciones para una aplicación usando una base de datos de SQL Server

Configuración:

<system.web>

<roleManager defaultProvider=" **bigbangAmex** "

enabled="true"

cacheRolesInCookie="true"

cookieName=".ASPROLES"

cookieTimeout="30"

cookiePath="/"

cookieRequireSSL="true"

cookieSlidingExpiration="true"

cookieProtection="All" >

<providers>

<add

name=" **bigbangAmex** "

type="System.Web.Security.SqlRoleProvider"

connectionStringName=" **bigbangAmex** "

applicationName="**bigbangAmex** " />

</providers>

</roleManager>

</system.web>

**FwkAuthorizationRuleProvider**

Este proveedor se encarga de brindar todo lo necesario para aplicar reglas de autorizacion (reglas para usuarios y roles). FwkAuthorizationRuleProvider almacena toda su informacion en la una base de datos de SQL y trabaja en conjunto a los proveedores anteriores, por lo tanto es necesatio tener na base d edatos aspnetdb embebida o externa donde alojar las reglas de una determinada aplicación.-

**Configuracion de seguridad**

Ver documento **Aquitectura Tecnológica Security Services.pdf**

**Instalación de base de datos de seguridad:**

Ver documento Fwk.Security.InstallDatabase.pdf

Fwk ActiveDirectory (LDAP Support)

El framework de seguridad soporta autenticación contra un controlador de dominios LDAP.