



deCost

Social Me

Preparado para: David Riu Herraiz, Consultor

Preparado por: Francisco Javier Costillo López

28 de mayo de 2012

TFC.NET - ETIS



Contenido

Documentación de despliegue	1	
Componentes necesarios		1
Archivo de configuración (Web.Config)		1
Páginas de error		2
Manual de usuario	3	
Registro		3
Acceso		3
Publicaciones		4
Imágenes		5
Programación		6
Usuarios enlazados (Amigos)		7
Mensajes		8
Configuración y generación de reglas de privacidad		9
Comentarios relevantes	11	



Documentación de despliegue

Al tratarse de una aplicación web, es necesario que el servidor contenga una instalación funcional del IIS. Éste IIS deberá de contener un *ApplicationPool* de máximo 1 hilo de ejecución sin límite de memoria física o virtual. Esto es debido a que se hace uso de una caché de sistema para almacenar las sesiones de usuario y cualquier limitación puede provocar un reciclaje constante de esta caché.

Por otro lado, la carpeta de despliegue debe de tener permisos tanto de lectura como de escritura para la aplicación, ya que se realizan dichas acciones en el funcionamiento normal de la aplicación.

Debido a la necesidad de persistencia de datos, se requiere el acceso a una base de datos SQL Server 2008 en la que se generará una instancia haciendo uso del script adjunto. La cadena de conexión a la base de datos se configurará siguiendo las instrucciones comentadas más adelante.

Componentes necesarios

Para el correcto despliegue de la aplicación, el servidor deberá de tener instalado los siguientes componentes:

- Framework .NET 4 o superior
- AjaxControlToolkit 4.1.60501 o superior (DLL adjunta en la carpeta ExternalReferences)

Archivo de configuración (Web.Config)

Para la configuración del envío de correos, se hace uso de las llaves de configuración situadas en el bloque de <appSettings>:

```
<add key ="adminEmail" value="fcostillo.uoc@gmail.com"/>
<add key ="noReply" value="fcostillo.uoc@gmail.com"/>
<add key ="smtpServer" value="smtp.gmail.com"/>
<add key ="smtpUser" value="fcostillo.uoc@gmail.com"/>
<add key ="smtpPass" value="Pr18opev."/>
```

Debido a algunos problemas con el smtp no he podido comprobar correctamente el envío de los emails, por lo que esta parte no ha sido posible completar su testing.

Para la conexión a la base de datos de la aplicación se ha de modificar la llave de ApplicationDB dentro del bloque de <connectionStrings>



```
<add name="ApplicationDB" connectionString="Data Source=COSTWIN-PC
\SQLEXPRESS;Initial Catalog=tfcnet;Integrated Security=True;Pooling=False"
providerName="System.Data.SqlClient" />
```

Para la compresión de los archivos javascript, css y contenido de la página he hecho uso de un componente llamado Miron. Para activar las opciones, se deben modificar los parámetros del bloque Miron.Web a lo siguiente:

```
<Miron.web>
  <!-- cachingStorage="OutputCache / FileSystem"-->
  <CompressorSettings compressCSS="true"
    reflectionAllowed="true"
    compressJavaScript="true"
    compressPage="true"
    combineCSS="true"
    combineHeaderScripts="false"
    compressWebResource="true"
    minifyContent="true"
    cachingStorage="FileSystem"
    autoMode="true"
    scriptsVersion="1"
    cssVersion="1">
  </CompressorSettings>
</Miron.web>
```

Páginas de error

Se debe configurar la redirección a la página de error a través de la configuración de IIS o haciendo uso de la entrada de configuración correspondiente en el archivo de [Web.Config](#). En este caso existe un archivo de página de error genérico situado en “~/Error.aspx” el cual debería de hacerse referencia de la siguiente manera dentro del bloque de <system.web>:

```
<customErrors mode="RemoteOnly" defaultRedirect="Error.aspx"/>
```



Manual de usuario

Registro

El usuario accede al formulario de registro a través del link situado en la pantalla de acceso. Una vez completado el registro se enviará un email con los datos confirmados y se podrá acceder al sitio web:

The screenshot shows a web browser window with the URL `http://192.168.1.212/Register.aspx`. The page has a light blue background and a central content area. At the top left of the content area is the logo "Social ME!" with a globe icon. Below the logo is a photograph of a woman in a white dress sitting on a white sofa. To the right of the photo is a registration form with the following fields and labels:

- Nombre completo**: Two input boxes labeled "Nombre" and "Apellidos".
- Fecha nacimiento**: One input box.
- Email**: Two input boxes, the first containing "correo@social" and the second labeled "Repetir email".
- Nombre usuario**: One input box labeled "Usuario".
- Contraseña**: Two input boxes, the first labeled "Contraseña" and the second labeled "Repetir contraseña".
- A checkbox labeled "Acepto la Política de privacidad".
- An "Acceder" button.

Below the photo, there is a section titled "Tu sitio está aquí" with the text: "Estamos reservando tu sitio justo a nuestro lado. Regístrate y comparte con nosotros tus fotos, pensamientos, ideas y sobretodo, disfruta de la experiencia de estar Socializado." Below this are two columns of text with small icons:

- Left column: "Comparte con tus amigos las fotos de tu último viaje por el mundo o tu excursión por la montaña. Te ofrecemos una visión original de tu álbum de fotos, para que tus amigos se mueran de la envidia. ¡Sonríe!"
- Right column: "Comunicate con tus amigos y familiares y comparte contenido. En esta vida estresante, casi no nos vemos unos a otros, ¿qué mejor manera de estar siempre conectados que aprovechar la conectividad de internet?"

At the bottom of the page, there is a footer: "© Copyright deCost 2012 | Política de privacidad | Contacto".

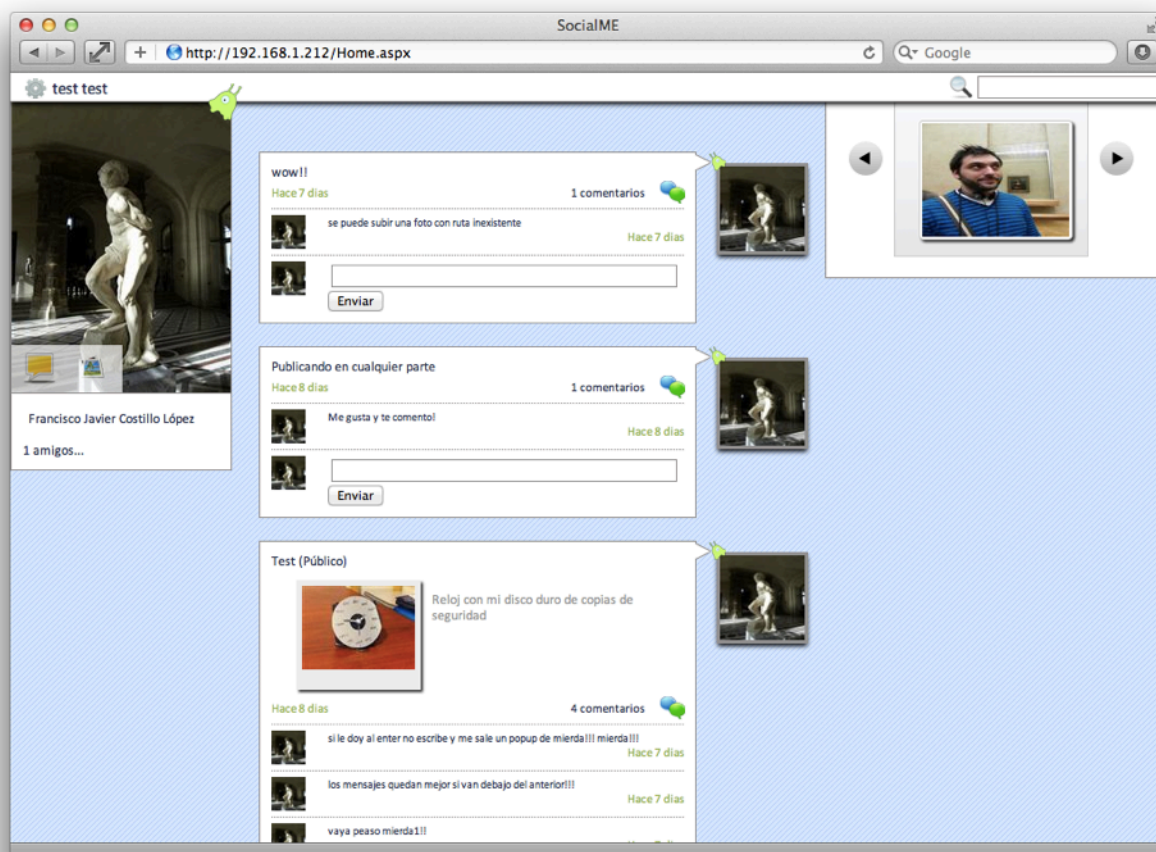
Acceso

El usuario podrá acceder al sitio web con los datos generados:



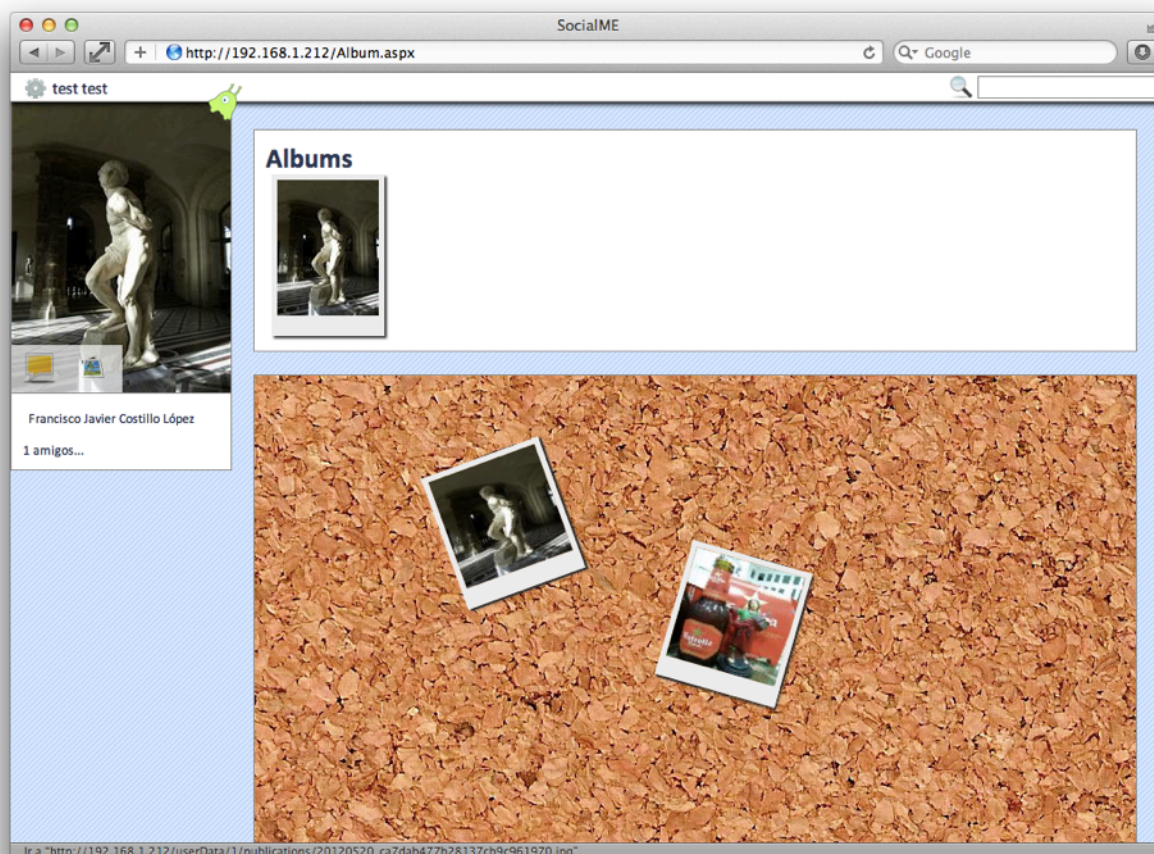
Publicaciones

Una vez se haya accedido al portal, se presentarán las publicaciones accesibles por reglas de privacidad:



Imágenes

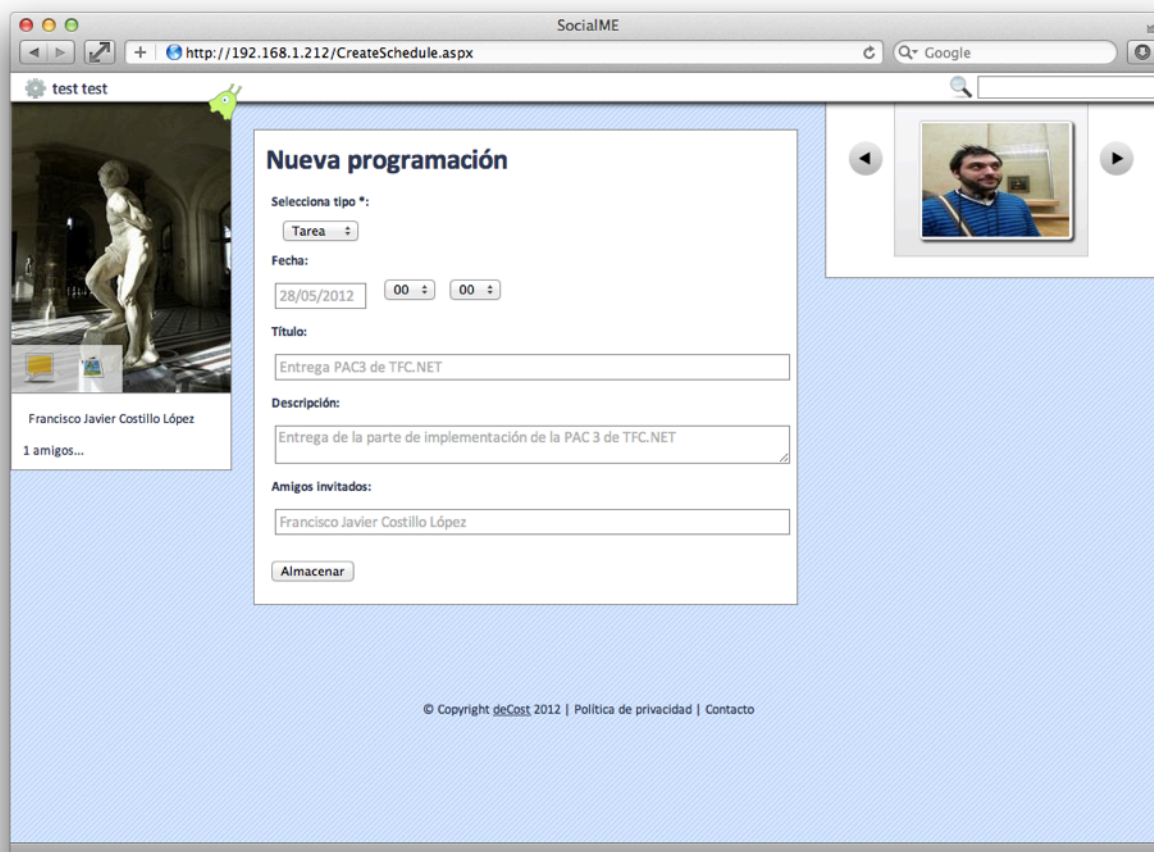
El usuario podrá acceder a sus álbumes a través del botón situado en la parte inferior de la foto de perfil:



Por otro lado, el usuario podrá acceder a un portal de subida de imágenes y generación de álbumes desde donde podrá subir imágenes al portal:

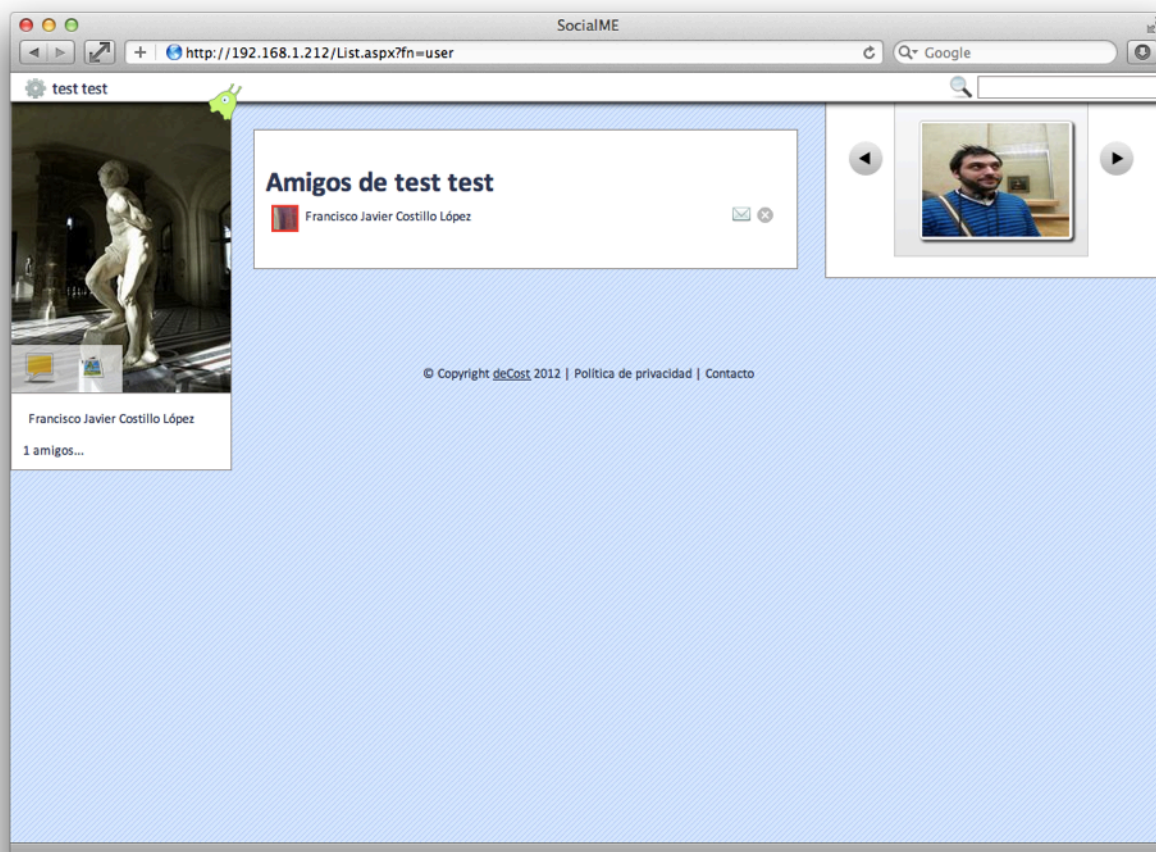
Programación

Haciendo uso del formulario accesible desde el menú, el usuario podrá completar la información necesaria para generar una nueva programación:



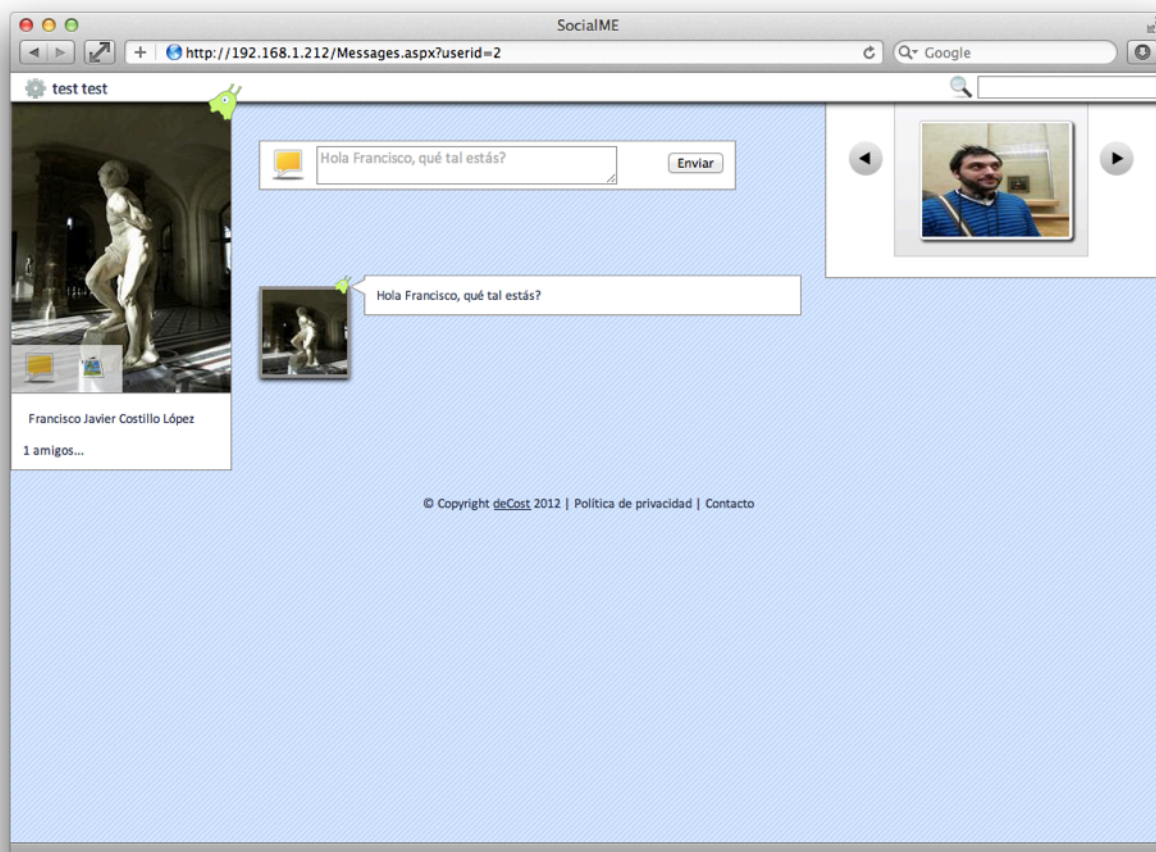
Usuarios enlazados (Amigos)

El usuario podrá acceder a un listado de sus amigos. En caso de tratarse de el listado del propio usuario, tendrá acceso a opciones como aceptación de solicitud de enlace, generar un mensaje o eliminar el enlace:



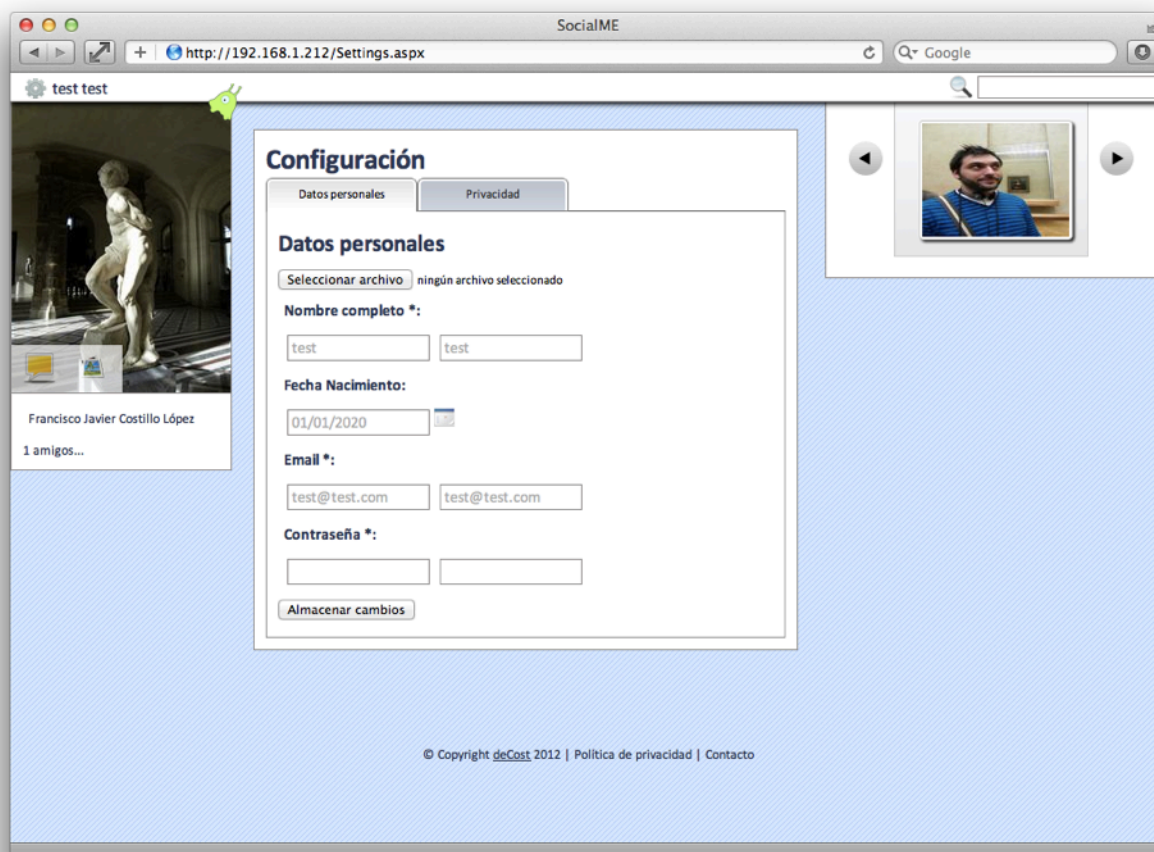
Mensajes

El usuario podrá compartir mensajes privados con otro usuario, haciendo uso del icono de mensajes en el listado de usuarios:

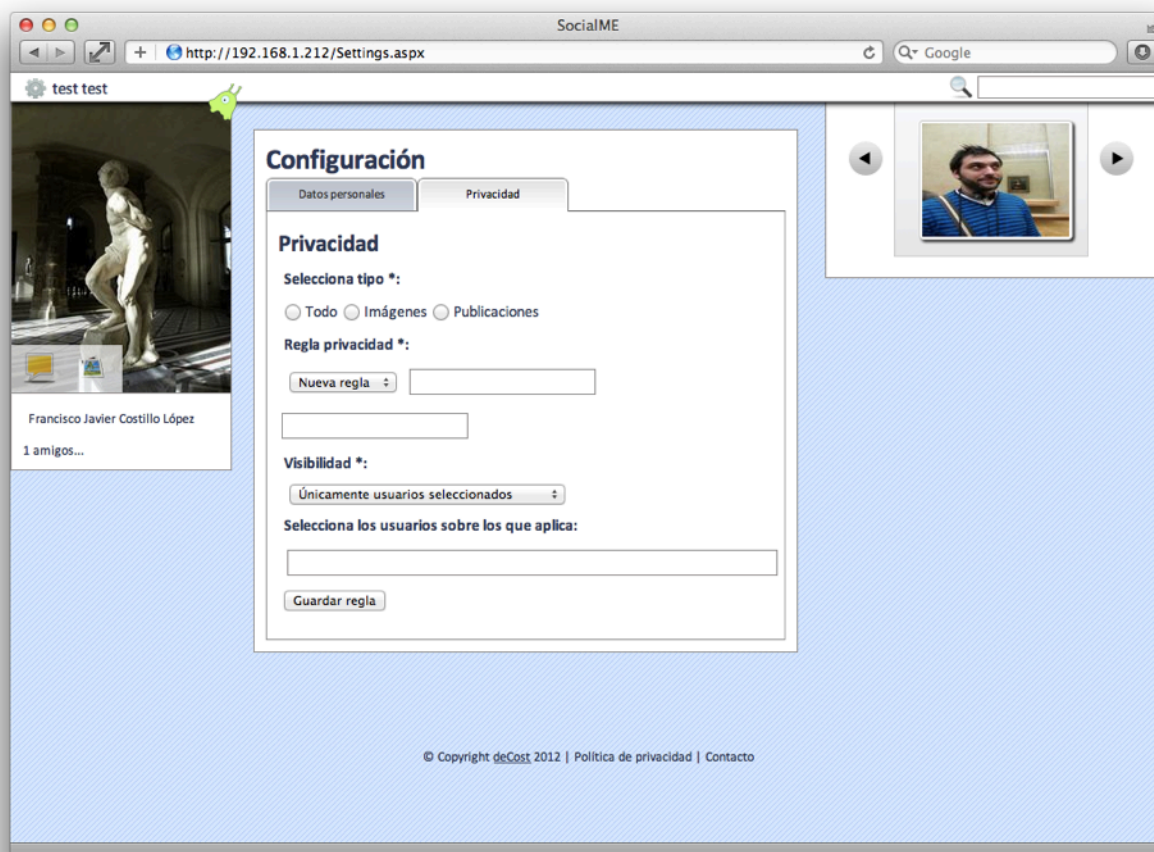


Configuración y generación de reglas de privacidad

El usuario puede modificar la configuración de su cuenta accediendo al formulario a través de la opción de menú:



Por otro lado, también podrá generar y modificar reglas de privacidad, accediendo a la pestaña correspondiente dentro de la misma página de configuración:



Comentarios relevantes

Debido a que el principal tema sobre el que se debía desarrollar la aplicación era la implementación de AJAX y la carga asíncrona, he hecho uso tanto de componentes de AjaxControlToolkit como de llamadas asíncronas generadas a partir del componente jQuery.

Uno de los problemas más complejos de solucionar ha sido la subida asíncrona de imágenes con el posterior almacenamiento en base de datos. Debido a restricciones del framework, el componente de subida de ficheros no es compatible con las llamadas asíncronas, por lo que al realizar Postback la página mediante un UpdatePanel, el fichero no persistía. Para solucionar este punto, he tenido que



hacer uso de un sistema de almacenado temporal en sesión del objeto generado por la imagen para su posterior tratamiento.

He intentado seguir una estructura simple donde cada método tenga un buen control y manejo de excepciones, generando una aplicación sólida y donde el usuario obtenga siempre una información útil de un posible problema ocurrido.

Por otro lado, he concentrado todo el trabajo de base de datos en procedimientos almacenados de base de datos, lo cual permite un mantenimiento de la transacción a nivel de base de datos, protegiendo la integridad de la base de datos, así como optimizando el rendimiento de algunas acciones.