

JoomGApps

Joomla

Google Apps

Integration

## Índice de contenido

Introducción.....	4
Motivación.....	4
El problema.....	5
Solución propuesta.....	5
Contexto.....	5
Estructura de Trabajo.....	6
Estado del artefacto.....	6
Análisis de Requisitos.....	8
Documentación de Usuario (Instalación / Configuración).....	11
Instalación de la extensión en Joomla.....	11
Configuración del componente desde el Back-End de Joomla.....	14
Ajustes en el Panel de Control de Google Apps.....	23
Comprobación de requisitos.....	28
Sincronización de Usuarios.....	29
Creación de un ítem de Menú.....	31
Verificación de la funcionalidad de Inicio de Sesión Unica.....	33
Detalle de los procesos de implementación.....	35
Preparación del entorno de desarrollo.....	35
El proceso de Inicio de Sesión Único.....	36
Inicio de sesión único SAML para Google Apps.....	36
Implementación del Inicio de Sesión Único en el componente.....	39
Proceso de generación de la claves.....	42
Proceso de Sincronización de Usuarios.....	44

---

Proceso de Comprobación de Requerimientos.....	46
Creación estructura de soporte Multilenguaje.....	46
Publicación del componente.....	47
Promoción de la extensión.....	54
Conclusiones.....	54
Bibliografía.....	55
Futuras ampliaciones de funcionalidad.....	55

## Introducción

### Motivación

La idea surge ante la necesidad de aumentar la funcionalidad de un website perteneciente a un centro educativo. En el contexto mencionado existen dos sistemas web independientes, el primero se dedica a la publicación de contenidos y esta implementado en Joomla, mientras que el otro cumple la función de herramienta de comunicación y trabajo colaborativo. Este último esta implementado mediante la plataforma Google Apps for Education.

Para aquellos que no conozcan ambos sistemas/servicios voy a detallar brevemente sus características.

- Joomla : Es un Sistema de Gestión de Contenidos Web o CMS que permite desarrollar sitios web de manera relativamente sencilla, proporcionando funciones básicas de categorización de la información, control de acceso, publicación asistida de contenidos, etc. Esta considerado uno de los CMS más extendidos en la actualidad
- Plataforma Google Apps for Education: Google Apps es un servicio que permite que organizaciones de mayor o menor tamaño utilicen las herramientas web 2.0 que ofrecen las cuentas de google de una manera integrada. De ese modo con Google Apps las organizaciones pueden migrar sus sistemas de correo electrónico, agenda de contactos, repositorio de documentos, calendario de eventos a este servicio.

Otra de las principales motivaciones a la hora de elegir este proyecto es aprender a programar sobre el Framework de Joomla. Considero que Joomla es un gran CMS con muchas perspectivas de futuro y me gustaría conocer la metodología de

desarrollo de sus extensiones.

## **El problema**

---

La principal desventaja del sitio web en la actualidad radica en la baja integración entre ambos sistemas. Por ello se realizó un estudio acerca de las extensiones de Joomla disponibles. La conclusión obtenida fue que existen muchos componentes Joomla que permiten integrar de manera independiente y sencilla las diferentes aplicaciones Google (maps, calendar, docs) pero ninguna de manera asociada a Google Apps. También descubrimos que no existía ninguna extensión que permitiera integrar la aplicación de correo Gmail con Joomla, proporcionando un acceso único (SSO - Single Sign On). Este último punto es el principal requerimiento del centro educativo, ya que se considera una característica indispensable para el uso regular del correo electrónico.

## **Solución propuesta**

---

Por ello he decidido embarcarme en la tarea de desarrollar un componente que supla las carencias arriba mencionadas, centrándome principalmente en la integración del correo electrónico y su proceso de identificación única.

En consecuencia, surge JoomGApps Component, procedente de la abreviatura Joomla Google Apps. Del nombre podemos intuir que se trata de un componente para el Sistema de Gestión de Contenidos (CMS) Joomla que tiene relación con el conjunto de herramientas colaborativas Google Apps.

## **Contexto**

---

Este componente es el que pretendo desarrollar y sugerir como PFC de mis estudios de Ingeniería Técnica de Gestión (plan 2001). Actualmente el componente se encuentra en fase de desarrollo inicial y una vez presentado ante mi tutor Ismael

Torres Boigues, perteneciente al Departamento de Sistemas Informáticos y Computación, hemos acordado un plan de trabajo que queda estructurado en el siguiente punto.

### **Estructura de Trabajo**

---

El proyecto en un principio constará de:

- Análisis de Requisitos
- Documentación de Usuario (Instalación/Configuración)
- Detalle de los procesos de implementación
- Desarrollo de un sitio web que incorpore el componente y muestre su funcionalidad
- Creación y promoción de una comunidad de desarrollo open source

### **Estado del artefacto**

---

En este apartado especificaré, qué tecnologías resuelven el problema y qué posibilidades de implementación dispongo.

En principio el componente utilizará el framework de Joomla, aunque también utilizó otras librerías de programación (Zend Framework, OpenSSL, XMLSec,...).

La tecnología que subyace en el proceso integrador se basa en Servicios Web, Protocolos ATOM, y Certificación de clave Asimétrica.

La parte de implementación del proceso de Inicio de Sesión Única (Single Sign On) esta basada en el lenguaje SAML (Security Assertion Markup Language) el cual se apoya en la transmisión de documentos XML firmados mediante el uso de protocolo SOAP sobre HTTP. En este punto hacemos uso de las librerías OpenSSL y XMLSec para generar las claves y firmar las repuestas SAML. Este proceso es cerrado y conforme a las especificaciones que requiere Google.

Por otro lado la sincronización de usuarios entre ambos sistemas se sustenta en el uso de identificación ClientLogin y el intercambio de datos mediante el protocolo de publicación ATOM. ClientLogin tiene el inconveniente en que delega en la aplicación web la gestión de las credenciales de identificación. Es decir, es necesario manejar el nombre de usuario y su contraseña para realizar un acceso acreditado a Google Apps. En este último punto hago uso de framework Zend para Google Apps que permite abstraer mediante un API simplificado la interacción con Google Apps.

En este punto podríamos haber seleccionado otro tipo de identificación, tales como:

- AuthSub, la cual realiza el proceso de autenticación del servidor a través de un proxy de autenticación, sin tener que manejar la información de acceso de sus usuarios. Para más información:

<http://code.google.com/intl/es-ES/apis/accounts/docs/AuthSub.html>

- OAuth, protocolo abierto que permite la compartición de información a terceros de una manera limitada y temporal a través de tokens sin hacer uso de la credenciales de identificación principales (usuario y contraseña). A diferencia de AuthSub, este protocolo es abierto y valido para cualquier tipo de aplicación: móvil, escritorio, web, etc.

Para más información:

<http://oauth.net/documentation/getting-started/>

---

## Análisis de Requisitos

---

Cualquier desarrollo de software comienza con el Análisis de Requisitos Software. Esta etapa ha estado marcada principalmente por la necesidad de de integrar ambas plataformas en mi centro de trabajo, un instituto de educación secundaria. El sitio web del centro llevaba funcionando un par de años con Joomla, y en el último curso se había introducido el sistema de correo de Google.

Hay que destacar que cualquier usuario propietario de un dominio puede contratar el producto Google Apps, no obstante ciertas funciones avanzadas como el API de administración o la posibilidad de Inicio de Sesión Única no están disponibles sin contratar el producto de Google Apps for Bussiness. Este último conlleva una tarifa mensual asociada al número de usuarios creados.

Sin embargo, por ser un centro educativo y tras una verificación de sus funciones, Google ofrece sin coste alguno el producto Google Apps for Education que es análogo al producto Google Apps for Bussiness.

La combinación de los sistemas Joomla y Google Apps esta muy extendida en los centros educativos, aunque los docentes y usuarios se ven en la obligación identificarse dos veces cuando desean acceder a los recursos que nos ofrecen estos sistemas. Además, al ser dos plataformas independientes, los administradores deben mantener dos conjuntos de usuarios sincronizados, y los usuarios deben recordar las contraseñas del usuario en los dos sistemas. Todo esto dificulta el acceso a los recursos por parte de los docentes.

En este contexto, y como coordinador TIC del Instituto me planteo en desarrollar un componente de código abierto con licencia GNU GPL que implemente esta característica, y proponerlo a mi tutor como Proyecto Final de Carrera de tipo B.

Uno de los referentes de la idea, fue un componente llamado Joodle



(<http://www.joomla.com>), el cual realiza un función similar pero integrando los sistemas Joomla y Moodle (Sistema de Gestión de Enseñanza a Distancia - Learning Management System). Este componente se presento inicialmente como un PFC de una Universidad Española (cubriendo los aspectos básicos de la integración) y posteriormente se ha convertido en un proyecto de Código Abierto que abarca varios componentes, módulos y plugins todos ellos relacionados con la integración completa de ambas plataformas. El proyecto posee una comunidad de desarrollo bastante activa.

En consecuencia y basándome en el éxito de Joomla, analizo las posibilidades que podría ofrecer en un futuro mi extensión. Existen multitud de puntos o características de integración entre Joomla y las herramientas de Google.

Por tanto planifico mi proyecto no solo como la implementación de la funcionalidad principal de Inicio de Sesión Única, si no como la creación de una estructura de soporte de una comunidad de desarrollo que colabore con la evolución del proyecto. Dicha estructura de soporte estaría compuesta de:

- Dominio propio del proyecto
  - Previa aceptación de Joomla por derechos de marca ( JoomGApps contiene parte del nombre de Joomla, y se considera un derivado )
- Hosting que de soporte al website del proyecto
- Alta en Google Apps for Bussiness para el dominio del proyecto ( habrá que sufragar la tarifa mensual o convencer a Google que mi componente promociona su sistema de Gestión, en ello estoy...)
- Foro
- Documentación en una wiki
- Espacio del proyecto en un repositorio de código abierto (JoomlaCode)

- Promoción de la extensión
- Subida y aprobación en Joomla Extensions Directory.

Una vez acotado el ámbito del proyecto decido marcar las funcionalidades a implementar:

- Inicio de Sesión Única para las siguientes aplicaciones de Google
  - Gmail
  - Calendar
  - Docs
- Sincronización de Usuarios entre plataformas para facilitar la tarea de los administradores

quedando otras características reseñadas en la introducción como propuestas a la comunidad de desarrollo.

## Documentación de Usuario (Instalación / Configuración)

### Instalación de la extensión en Joomla

El proceso de Instalación/Configuración del componente comienza por la Instalación de la Extensión en el sistema Joomla. Para ello deberemos acceder al Back-End del sitio web generado con Joomla, acreditar nos como Administrador y posteriormente instalar la extensión a través del Gestor de Extensiones del CMS.



Captura 1: Acceso al Back-End



Captura 2: Identificación en el Back-End



Captura 3: Menú de Instalación de Extensiones



Joomla! es software libre liberado bajo la Licencia GNU/GPL.  
Pack creado por Joomla! Spanish 2010 - Patrocinado por Web Empresa

Captura 4: Instalar extensión



Captura 5: Extensión instalada con éxito

## Configuración del componente desde el Back-End de Joomla

Una vez instalada la extensión podemos acceder al menú de configuración situado en la sección de Componentes



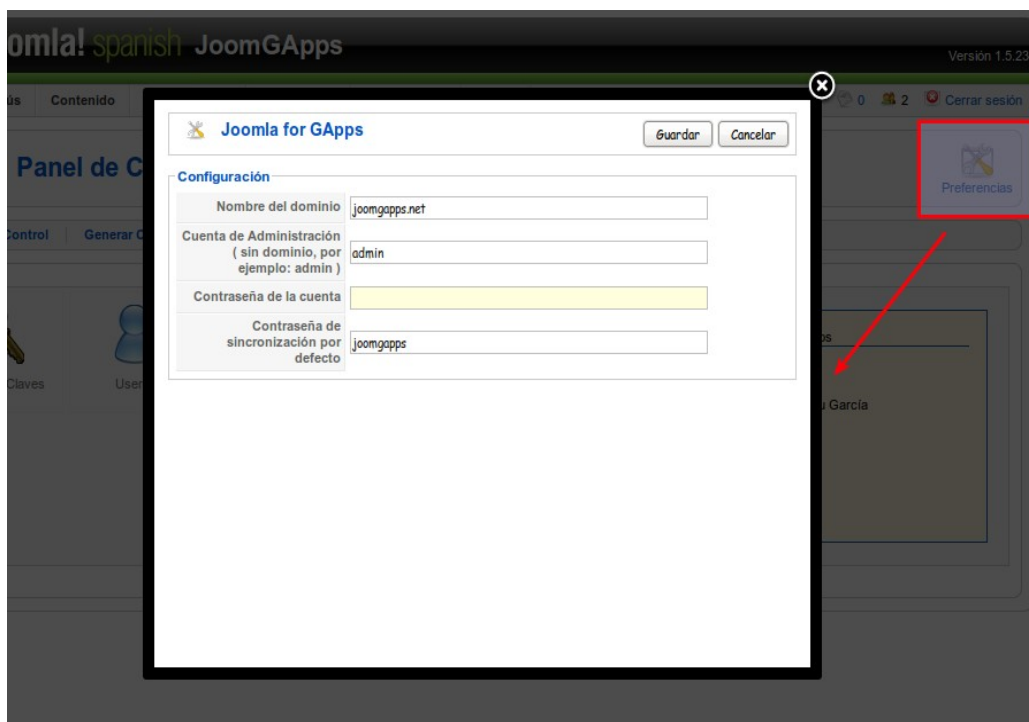
Captura 6: Menú JoomGApps

El componente posee un Panel de Control desde el cual se accede a todos los ajustes.



Captura 7: Panel de Control JoomGApps

El primer paso para la correcta configuración del componente es definir el dominio de Google Apps asociado. Dicho dominio se determina en la sección de Preferencias del componente.



Captura 8: Preferencias Globales

Junto al dominio podemos determinar las credenciales de acceso del administrador del dominio en la plataforma de Google, así como la contraseña por defecto para la sincronización.

Estos últimos parámetros son necesarios en el proceso de sincronización de usuarios, ya que el componente debe conectarse como administrador al sistema Google Apps para la lectura/creación de cuentas de usuario. La contraseña de sincronización se utiliza para establecer una contraseña por defecto a los usuarios importados desde el sistema Google Apps.

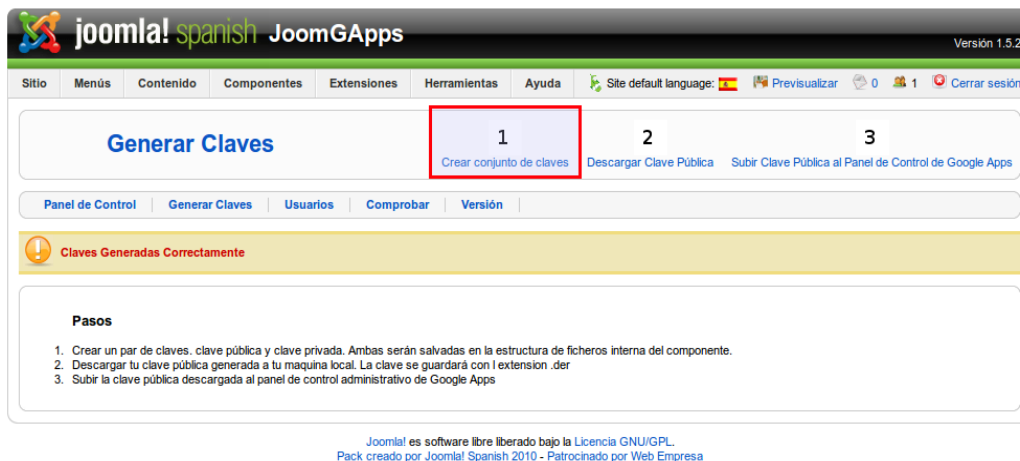
Una vez establecidos los parámetros básicos, debemos generar las claves que nos permitirán el proceso de identificación única. Para ello haremos uso del menú

Generar Claves, el cuál se divide en 3 pasos.



Captura 9: Vista Generar Claves

1. Crear conjunto de claves: Genera un sistema de claves asimétrico, en el cual existen dos claves: una privada que se almacena internamente en el sistema, y otra pública que a modo de certificado se exporta a Google



Captura 10: Opción Crear Conjunto de Claves



2. Descargar clave pública: Se descarga la clave pública a modo de certificado en un fichero local.



Captura 11: Descargar Clave Pública

3. Acceso al Panel de Control de Google Apps: Enlace que nos dirige al panel de control de Google Apps para seguir con la configuración.



Captura 12: Acceso al Panel de Control GApps

## Ajustes en el Panel de Control de Google Apps

Debemos configurar el sistema Google Apps para que interactúe con nuestro componente. Ello nos obliga a acceder al panel de administración o panel de control mediante las credenciales de la cuenta administradora del dominio.

Una vez identificados, el sistema nos muestra el panel de control.

Google apps Google Apps para joomgapps.net - Google Apps for Business admin@joomgapps.net Bandeja de entrada Calendar Ayuda Salir

Buscar cuentas Buscar en el Centro de asistencia

Panel Organización y usuarios Grupos Personalización del dominio **Herramientas avanzadas** Establecimiento Asistencia

Configuración

⚠ El acceso a la API está habilitado. Las actualizaciones que efectúes mediante este panel de control no se transferirán al sistema de gestión del usuario. [Más información](#)

Continuar con la guía de configuración » Sigue utilizando esta guía para configurar correctamente Google Apps para tu organización. [Cerrar esta guía](#)

**joomgapps.net**  
joomgapps.net, joomgapps.net.test-google-a.com  
[Administrar información de cuenta](#) [Nombres de dominio](#)

2 usuarios  
Puedes crear hasta 10 cuentas de usuario para esta organización.

✓ Todos los servicios de Google Apps funcionan correctamente. [Más información](#)

Usuarios activos durante siete días de los últimos 90 días gráficas de usuarios activos durante siete de los últimos 90 días actual 0

Actualizado el 24 de agosto de 2011 [Más información](#)

Configuración del servicio [Añadir más servicios](#)

Google Apps Marketplace Nuevo [Comprar en el mercado »](#)  
Consigue más aplicaciones como, por ejemplo, herramientas de contabilidad, de CRM, de marketing, de gestión de proyectos y de administración.

Captura 13: Panel de Control GApps

Posteriormente nos dirigiremos a la sección de Herramientas Avanzadas, apartado Autenticación, Configurar inicio de sesión único.

Google apps Google Apps para joomgapps.net - Google Apps for Business admin@joomgapps.net Bandeja de entrada Calendar Ayuda Salir

Buscar cuentas Buscar en el Centro de asistencia

Panel Organización y usuarios Grupos Personalización del dominio **Herramientas avanzadas** Establecimiento Asistencia

Configuración

**Herramientas avanzadas** [Preguntas frecuentes](#)

Crear varios usuarios

Subida masiva  
Si quieres crear y actualizar varias cuentas de usuario a la vez, sube un archivo CSV.

Descarga de Google Apps Directory Sync  
Si dispones de un servidor de directorio LDAP en tu organización, con Google Apps Directory Sync puedes importar automáticamente usuarios y grupos a Google Apps. Google Apps Directory Sync es una aplicación cliente que establece reglas para sincronizar Microsoft Active Directory, IBM Lotus Domino y otros servidores LDAP con Google Apps. Una vez que se hayan creado las reglas, deberás ejecutar la sincronización en tu interfaz de línea de comandos.

Autenticación

**Configurar inicio de sesión único (SSO)**  
El servicio de inicio de sesión único (SSO) basado en SAML te permite autenticar cuentas de usuario para aplicaciones basadas en Internet, como Gmail o Google Calendar. Para aplicaciones de escritorio, como Google Talk o acceso POP a Gmail, los usuarios deben continuar accediendo directamente con su nombre de usuario y contraseña de Google

Captura 14: Herramientas Avanzadas

En dicha sección debemos habilitar el inicio de sesión único, configurar 3 parámetros requeridos, además de subir al panel de control el fichero de certificado de clave pública que antes hemos generado.

Google Apps para joomgapps.net - Google Apps for Business admin@joomgapps.net [Bandeja de entrada](#) [Calendar](#) [Ayuda](#) [Salir](#)

Buscar cuentas Buscar en el Centro de asistencia

Panel Organización y usuarios Grupos Personalización del dominio **Herramientas avanzadas** Establecimiento Asistencia

Configuración « Volver a Herramientas avanzadas

### Configurar inicio de sesión único (SSO)

Para configurar SSO, proporcionanos la siguiente información: [Referencia de SSO](#)

**Habilitar inicio de sesión único**

**URL de la página de acceso \***  
 URL para acceder a tu sistema y a Google Apps

**URL de la página de fin de sesión \***  
 URL para redirigir usuarios cuando finalizan la sesión

**Cambiar URL de contraseña \***  
 URL para que los usuarios cambien su contraseña en tu sistema.

**Certificado de verificación \***  
Se ha subido un archivo certificado. [Sustituir certificado](#)

El archivo certificado debe contener la clave pública para que Google pueda verificar las solicitudes de acceso. [Más información](#)

**Utilizar una determinada entidad emisora de dominios**

Debes seleccionar esta opción si tu dominio utiliza un agregador IDP para gestionar las solicitudes SAML. Si se habilita, el valor del emisor indicado en la solicitud SAML será `google.com/a/joomgapps.net` en lugar de simplemente `google.com`. [Más información](#)

**Máscaras de red**

Las máscaras de red determinan qué direcciones se ven afectadas por el inicio de sesión único. Si no se especifican máscaras, se aplicará la funcionalidad SSO a toda la red.  
Utiliza un punto y coma para separar las máscaras. Ejemplo: (64.233.187.99/8; 72.14.0.0/16)  
Utiliza el guión para indicar rangos. Ejemplo: (64.233.167-204.99/32)  
Todas las máscaras de red deben terminar con un CIDR. [Más información](#)

[Condiciones del servicio](#) [Condiciones de facturación](#) [Política de privacidad](#) [Sugerir una función](#) [Página principal de Google](#)  
©2011 Google Inc.

Captura 15: Configurar Inicio de Sesión Único

Si entramos en detalles los campos requeridos que debemos completar son:

- URL de la página de acceso
- URL de la página de fin de sesión
- URL para el cambio de contraseña

Estos campos permiten a Google Apps redirigir la solicitudes de acceso a sus formularios de login, y cambio de contraseña, a las URL de Joomla que asumen dicha funcionalidad. Debemos entender que Google Apps delega la identificación de usuarios en Joomla, y por tanto es necesario definir a qué URL serán redirigidos los usuarios que quieran acceder al sistema Google Apps. El establecimiento de estos campos implica que los usuarios convencionales accederán al sistema a través de website desarrollado en Joomla y nunca a través del formulario de identificación de Google. En esta cuestión existe una excepción: el usuario administrador siempre tendrá un acceso diferenciado que le permitirá acceder de manera directa (no a través de Joomla)

En este apartado debemos cargar el certificado descargado previamente. Cabe destacar que esta operación debe repetirse cada vez que se genere un nuevo sistema de claves, es decir cada vez que se presione al botón 1 de la sección *GenerarClaves*, ya que en ese caso dejan de corresponder la clave privada almacenada internamente por el componente y el certificado de clave pública cargado en Google Apps.

### Configurar inicio de sesión único (SSO)

Para configurar SSO, proporcionanos la siguiente información: [Referencia de SSO](#)

**Habilitar inicio de sesión único**

**URL de la página de acceso \***

URL para acceder a tu sistema y a Google Apps

**URL de la página de fin de sesión \***

URL para redirigir usuarios cuando finalizan la sesión

**Cambiar URL de contraseña \***

URL para que los usuarios cambien su contraseña en tu sistema.

**Certificado de verificación \***

El archivo certificado debe contener la clave pública para que Google pueda verificar las solicitudes de acceso. [Más información](#)

*Captura 16: Detalle Configurar Inicio de Sesión Único*

Una vez tenemos configurado el inicio de sesión único, debemos habilitar el API de administración. A través de dicha funcionalidad podemos sincronizar los usuarios existentes en ambos sistemas. Esto se debe al requisito "el usuario debe estar definido en ambas plataformas con el mismo identificador para que pueda efectuar el inicio de sesión único".

Google apps **Google Apps para Joomgapps.net - Google Apps for Business** admin@joomgapps.net [Bandeja de entrada](#) [Calendario](#) [Ayuda](#)

Buscar cuentas Buscar en el Centro de asistencia

Panel Organización y usuarios Grupos Personalización del dominio Herramientas avanzadas Establecimiento Asistencia Configuración

**Personalización del dominio**

General Suscripciones y facturación Nombres de dominio Configuración de usuario Apariencia

Compartir contactos  **Habilitar la función de compartir contactos**  
Comparte contactos automáticamente en . La información de contacto no se compartirá fuera de . [Más información](#)

Elige las direcciones de correo electrónico de los usuarios que deben mostrarse a otros usuarios:

Mostrar todas las direcciones de correo electrónico

Ocultar seudónimos

Oculta la dirección de correo electrónico principal si el usuario tiene un seudónimo.

**Inhabilitar la función de compartir contactos**  
No habilitas automáticamente el uso compartido de los contactos en . [Más información](#)

**Habilitar API de administración**  **Habilitar API de administración**  
La API de administración te permite administrar cuentas de usuario de forma programada y sincronizar tu base de usuarios de Google Apps con tu propio sistema de administración de usuarios. Esta función está disponible sólo en inglés. [Más información](#)

Captura 17: Habilitar API Administración

**Habilitar API de administración**  **Habilitar API de administración**  
La API de administración te permite administrar cuentas de usuario de forma programada y sincronizar tu base de usuarios de Google Apps con tu propio sistema de administración de usuarios. Esta función está disponible sólo en inglés. [Más información](#)

**Página de acceso** [Configurar inicio de sesión único \(SSO\)](#)  
El servicio de inicio de sesión único (SSO) basado en SAML te permite autenticar cuentas de usuario para aplicaciones basadas en Internet, como Gmail o Google Calendar. Para aplicaciones de escritorio, como Google Talk o acceso POP a Gmail, los usuarios deben continuar accediendo directamente con su nombre de usuario y contraseña de Google Apps. [Más información](#)

**Configuración avanzada de la contraseña** [Ajustar configuración de la contraseña](#)  
Permite controlar la longitud de las contraseñas que se utilizan en y supervisar la seguridad de las contraseñas de los usuarios existentes.

**Guardar cambios** Cancelar

Captura 18: Guardar Cambios en Habilitar API

Siempre debemos acordarnos de guardar los cambios de la configuración.

## Comprobación de requisitos

Una de las vistas del componente, nos permite realizar la comprobación de los requisitos necesarios para el correcto funcionamiento de la misma



Captura 19: Menú Comprobar

En ella podemos observar un lista de comprobaciones, junto al resultado de las mismas.



Captura 20: Vista Comprobar

## Sincronización de Usuarios

Una de las principales funcionalidades del componente es la sincronización de usuarios entre ambas plataformas. Para ello haremos uso del menú Usuarios del Panel de Control



Joomla! es software libre liberado bajo la Licencia GNU/GPL.  
Pack creado por Joomla! Spanish 2010 - Patrocinado por Web Empresa

Captura 21: Menú Usuarios

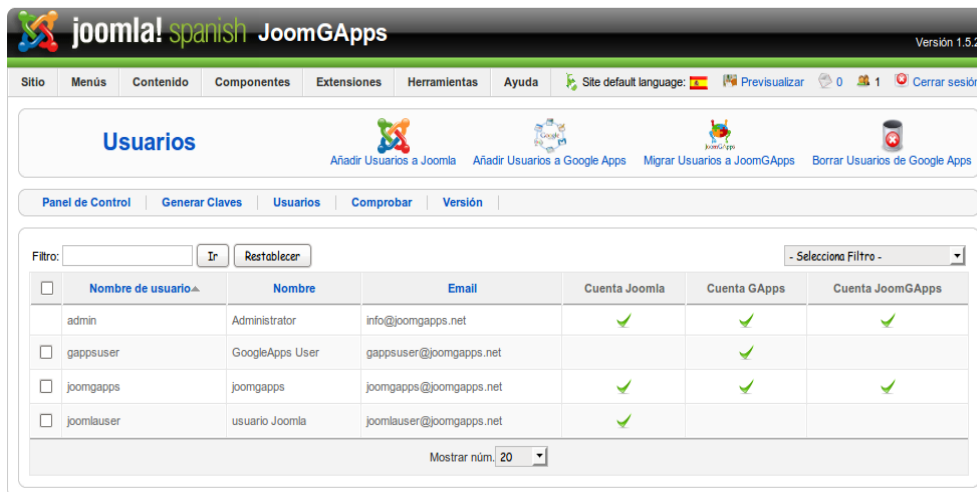
Si todo esta configurado de manera correcta, accederemos a una vista donde se muestran los usuarios definidos en ambos sistemas. En la vista se muestran los principales atributos:

- Nombre de usuario: identificador del usuario
- Nombre: Nombre y Apellidos
- E-mail: Correo electrónico
- Cuenta Joomla: Indica si es un usuario de Joomla
- Cuenta Gapps: Indica si es un usuario de Google Apps
- Cuenta JoomGApps: Indica si es un usuario de ambos sistemas

La vista de usuario dispone de 4 controles que ejecutan los siguientes procesos:+

- Añadir Usuarios a Joomla: Agrega los usuarios marcados al sistema Joomla

- (si no existen en el mismo). Los usuarios nuevos adquirirán la contraseña por defecto definida en los parámetros globales
- **Añadir Usuarios a Google Apps:** Agrega los usuarios marcados al sistema Google Apps, (si no existen en el mismo). Los usuarios nuevos adquirirán la contraseña del usuario Joomla
  - **Migrar usuarios a JoomGApps:** Sincroniza los usuarios de ambos sistemas. Los usuario de Joomla que no existen en Google Apps se crean en el sistema de Google y los usuarios de Google Apps que no existen en Joomla se crean en el CMS
  - **Borrar usuarios de Google Apps:** Permite el borrado de Usuarios de Google Apps. Debemos ser cautos con esta operación puesto que Google Apps realiza borrados lógicos de sus cuentas, y existe un periodo de varios días en los que no se pueden generar un usuario borrado anteriormente.



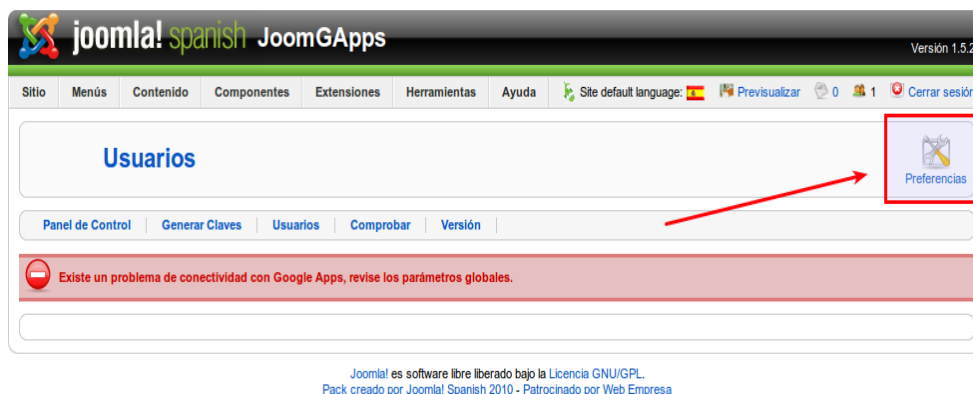
Joomla! es software libre liberado bajo la Licencia GNU/GPL.  
Pack creado por Joomla! Spanish 2010 - Patrocinado por Web Empresa

Captura 22: Vista Usuarios

En caso de no poder contactar con el API de administración de Google, la vista muestra un mensaje de error y nos invita a revisar los parámetros de configuración



globales.



*Captura 23: Error Conectividad API Administración*

No obstante si observamos este mensaje también es conveniente revisar la configuración del panel de control de Google Apps ( sección de activación del API de administración, ver captura 21)

La función de sincronización es útil cuando tenemos el conjunto de usuarios definido en uno de los sistemas y queremos migrarlo al sistema contrario. Debemos tener en cuenta que es necesario que un usuario coexista en ambas plataformas para que se pueda realizar el inicio de sesión único.

## **Creación de un ítem de Menú**

Una vez configurado el componente y sincronizados los usuarios candidatos al inicio de sesión único, nos dispondremos a definir los ítems de menú que habiliten el acceso a las diferentes aplicaciones de Google.

Así pues, accederemos a la configuración de ítems del menú que nos interese, para crear un nuevo ítem de menú.



Captura 24: Nuevo ítem de Menú

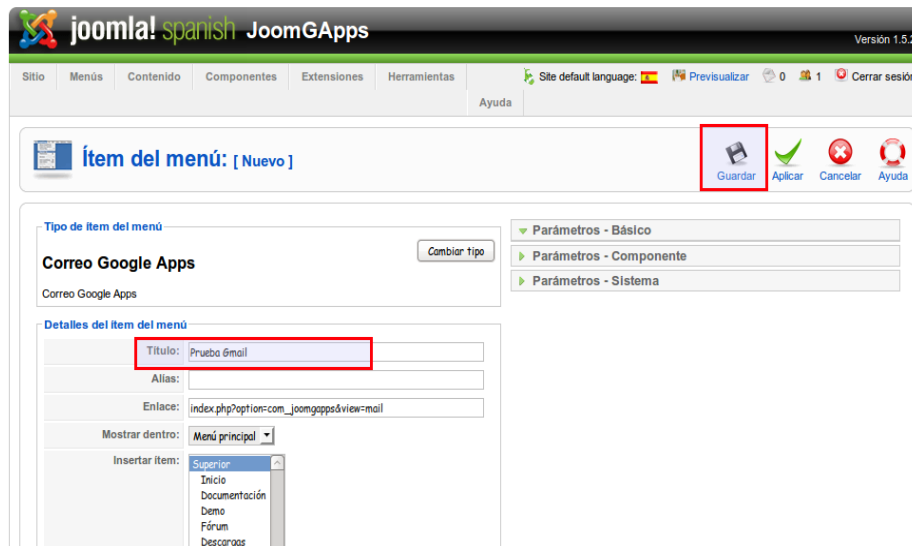
A continuación el sistema nos preguntará acerca del tipo de ítem de menú. Así que deberemos elegir cualquiera de los contenidos en la opción JoomGApps en función de la aplicación de Google que deseamos mostrar:

- Gmail
- Google Docs
- Calendar



Captura 25: Selección tipo ítem de menú

Posteriormente definiremos un nombre para el ítem, junto con algunos parámetros adicionales y seleccionaremos la opción de guardar.



Captura 26: Definición ítem de menú

## Verificación de la funcionalidad de Inicio de Sesión Unica

Ahora ya es momento de comprobar la creación del ítem de menú y su esperada funcionalidad



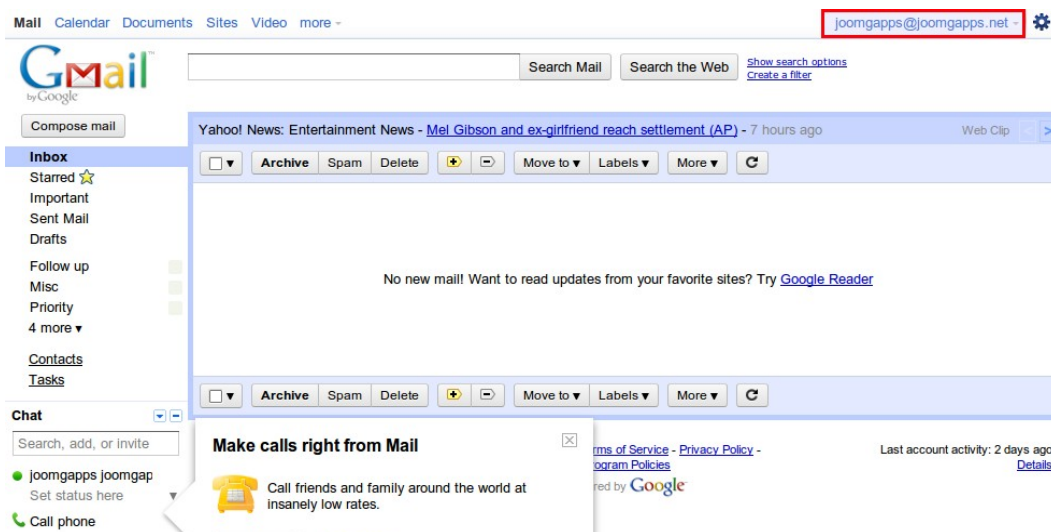
Captura 27: Prueba ítem de menú

El componente nos llevará a la pantalla de inicio de sesión por defecto en Joomla, en caso de no estar identificados en el sistema



Captura 28: Pantalla de identificación en Joomla

o al correo electrónico de Google en caso de estar identificado o tras haber realizado la correspondiente introducción de credenciales de acceso



Captura 29: Correo de Google Apps

## Detalle de los procesos de implementación

### Preparación del entorno de desarrollo

Antes de comenzar con el desarrollo, preparo el entorno de desarrollo. Así que trabajando sobre una maquina Ubuntu Lucid Lynx, instalo

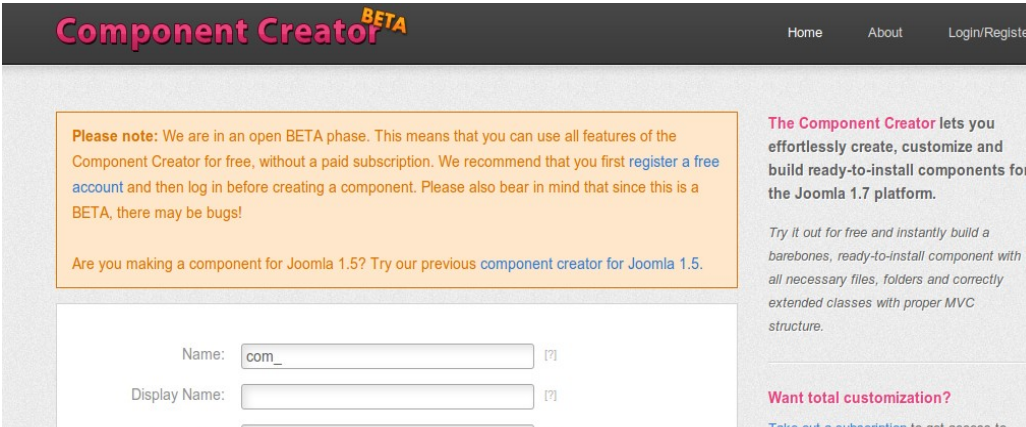
- apache como servidor web
- el interprete PHP
- eclipse como herramienta de desarrollo
- phing como herramientas para la sincronización de ficheros entre proyectos eclipse
- xdebug como depurador php

Todos estos pasos estan correctamente documentados en la documentación de Joomla ( <http://docs.joomla.org> )

- [Setting up your workstation for Joomla! Development](#)
- [Setting up your workstation for extension development](#)

Para generar la estructura inicial del componente utilicé un generador on-line que crea la estructura inicial de ficheros y carpetas

<http://www.notwebdesign.com/joomla-component-creator/index.php>



The screenshot shows the Joomla Component Creator website. At the top, there is a navigation bar with 'Home', 'About', and 'Login/Register' links. The main content area features a prominent orange box with a 'Please note' message about the BETA phase and a recommendation to register for a free account. Below this, there is a form with three input fields: 'Name' (containing 'com\_'), 'Display Name', and 'Description'. To the right of the form, there is a text block describing the tool's capabilities and a link to 'Want total customization?'. The overall layout is clean and professional, with a focus on user guidance.

Captura 30: Generador MVC Joomla Extension

## **El proceso de Inicio de Sesión Único**

---

Security Assertion Markup Language (SAML) es un estándar XML que permite que los dominios web seguros intercambien datos de autenticación y autorización de usuarios. Con SAML, un proveedor de servicios online puede ponerse en contacto con otro proveedor de identidad para autenticar a los usuarios que intenten acceder a contenido seguro.

Google Apps ofrece un servicio de inicio de sesión único basado en SAML que ofrece a las organizaciones un control total de la autorización y de la autenticación de cuentas de usuario alojadas que pueden acceder a aplicaciones basadas en web como Gmail o Google Calendar. Con el modelo SAML, Google actúa como **proveedor** y ofrece servicios como Gmail y páginas de inicio. Los partners de Google actúan como **proveedores de identidad** y administran los nombres de usuario, las contraseñas y otra información para identificar, autenticar y autorizar a los usuarios en aplicaciones web alojadas por Google.

Es importante tener en cuenta que la solución de inicio de sesión único solo sirve para aplicaciones web.

El servicio de inicio de sesión único de Google Apps se basa en las [especificaciones SAML 2.0](#).

### **Inicio de sesión único SAML para Google Apps.**

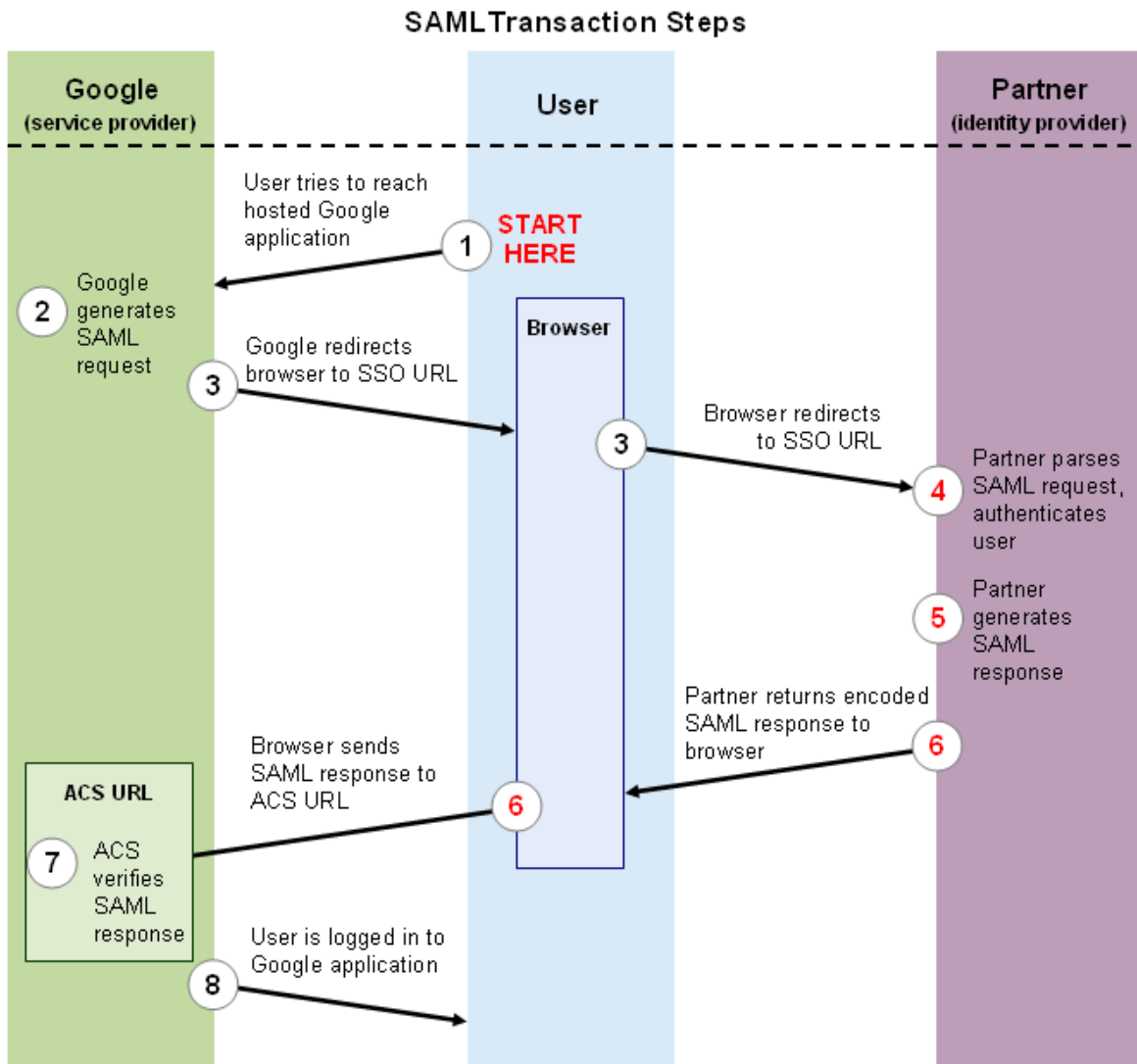
---

El siguiente punto explica cómo accede un usuario a una aplicación alojada de Google mediante un servicio de inicio de sesión único basado en SAML ofrecido por un partner.

La captura que se muestra a continuación, ilustra el proceso por el que un usuario accede a una aplicación de Google Apps, como Gmail, a través de un servicio de inicio de sesión único basado en SAML. La lista numerada a continuación de la imagen explica

cada paso con más detalle.

**Nota:** para que se realice este proceso, el partner debe facilitar a Google la URL para su servicio de inicio de sesión único, así como la clave pública que Google debería usar para verificar las respuestas SAML.



Captura 31: Acceso a Google Apps usando SAML

Esta imagen describe los pasos siguientes.

1. El usuario intenta acceder a una aplicación alojada en Google, como Gmail, páginas de inicio u otro servicio de Google.
2. Google genera una solicitud de autenticación SAML. La solicitud SAML se

codifica y se inserta en la URL del servicio de inicio de sesión único del partner. El parámetro "RelayState" que contiene la URL codificada de la aplicación de Google a la que el usuario intenta acceder también está incrustada en la URL de inicio de sesión único. Este parámetro también pretende ser un identificador opaco que se facilita sin ninguna modificación ni inspección.

3. Google envía una redirección al navegador del usuario. La URL de redireccionamiento incluye la solicitud de autenticación SAML codificada que debe enviarse al servicio de inicio de sesión único del partner.

4. El partner descodifica la solicitud SAML y extrae la URL del servicio ACS (Assertion Consumer Service) de Google y la URL de destino del usuario (parámetro "RelayState"). A continuación, el partner autentica al usuario. Los partners pueden autenticar a los usuarios solicitando credenciales de acceso válidas o comprobando las cookies de sesión válidas.

5. El partner genera una respuesta SAML que contiene el nombre de usuario del usuario autenticado. Según la especificación SAML 2.0, esta respuesta se firma digitalmente con las claves DSA/RSA públicas y privadas del partner.

6. El partner codifica la respuesta SAML y el parámetro "RelayState" y, a continuación, devuelve esa información al navegador del usuario. El partner ofrece un mecanismo para que el navegador pueda enviar esa información al servicio ACS de Google. Por ejemplo, el partner podría insertar la respuesta SAML y la URL de destino en un formulario y facilitar un botón para que el usuario pueda hacer clic y enviarlo a Google. El partner también podría incluir JavaScript en la página que envía el formulario a Google de forma automática.

7. El servicio ACS de Google verifica la respuesta SAML usando la clave pública del partner. Si la respuesta se verifica correctamente, el servicio ACS



redirecciona al usuario a la URL de destino.

8. Se ha redireccionado al usuario hacia la URL de destino y ha accedido a Google Apps.

### **Implementación del Inicio de Sesión Único en el componente**

---

El proceso arriba detallado se implementa en el fichero *components/com\_joomgapps/helpers/helper.php*.

Allí podemos encontrar diferentes funciones determinantes dentro del proceso SAML.

- function **createAuthnRequest**(\$acsURL, \$providerName), genera una solicitud de autenticación codificada a partir de los datos pasados por parámetro (puntos 3 y 4). Esta solicitud corresponde al esquema siguiente:

```
<?xml version="1.0" encoding="UTF-8"?>
<samlp:AuthnRequest
  xmlns:samlp="urn:oasis:names:tc:SAML:2.0:protocol"
  ID="<AUTHN_ID>"
  Version="2.0"
  IssueInstant="<ISSUE_INSTANT>"
  ProtocolBinding="urn:oasis:names:tc:SAML:2.0:bindings:HTTP-Redirect"
  ProviderName="<PROVIDER_NAME>"
  AssertionConsumerServiceURL="<ACS_URL>"/>
```

- function **buildSAMLResponse**(\$username, \$acsURL,&\$error), genera una respuesta SAML a partir de los datos recibidos por Google. Invoca a la función anterior para decodificar la solicitud de autenticación, construye la respuesta y la firma con la clave privada.
- function **signResponse**(\$responseXmlString, &\$error) , realiza la firma de la respuesta SAML de acuerdo con los estándares de facto. (<http://www.w3.org/TR/xmldsig-core/>). Para realizar este proceso inicialmente me apoye en la librería en C xmlsec1, disponible en el

repositorio de los sistemas Debian (<http://www.aleksey.com/xmlsec/>). Sin embargo al trasladar el componente a un servicio de Hosting, tuve muchos problemas puesto que la mayoría de los proveedores no tenían instalado esa librería y tampoco permitían el acceso a comandos de consola desde PHP (mediante el la función `exec`). En consecuencia tuve que buscar una solución alternativa, así que acabe encontrando una librería php que emulaba el proceso de firmado de XML. Esta librería se encontraba en el repositorio de proyecto de Google (<http://code.google.com/p/xmlseclibs/>) y tras algunas modificaciones conseguí adaptarla para que firmara la respuesta de manera correcta. Debo destacar la ayuda prestada por su desarrollador Rob Richards quien me asesoró en el correcto uso de la misma. Los principales problemas que tuve fueron:

- la Canonicalización de la respuesta, al final opté por el tipo `EXC_C14N_COMMENTS`
- el firmado mediante claves RSA, debido a que la librería tan solo permitía el firmado mediante certificados de tipo X. 509. La solución fue agregar a la librería un método que permitía el firmado con este tipo de claves.
- La adaptación de la librería al firmado de esquemas xml a modo de plantilla. Esta adaptación no se realizó dentro de la librería, si no que más bien fue una adaptación de su uso ( modificando la respuesta antes de firmarla )
- funciones de conversión de claves entre formatos pem y der. Esto se debe a que Google acepta certificados y claves en formato binario (der), mientras que la librería php tan solo acepta claves en formato base64

(pem)

A continuación presento el formato de respuesta SAML enviada:

```
<samlp:Response ID="<RESPONSE_ID>" IssueInstant="<ISSUE_INSTANT>" Version="2.0"
Destination="<DESTINATION>" InResponseTo="<REQUEST_ID>"
  xmlns="urn:oasis:names:tc:SAML:2.0:assertion"
  xmlns:samlp="urn:oasis:names:tc:SAML:2.0:protocol"
  xmlns:xenc="http://www.w3.org/2001/04/xmlenc#">
  <Signature xmlns="http://www.w3.org/2000/09/xmldsig#">
    <SignedInfo>
      <CanonicalizationMethod Algorithm="http://www.w3.org/2001/10/xml-exc-
c14n#WithComments" />
      <SignatureMethod Algorithm="http://www.w3.org/2000/09/xmldsig#RSADSA-sha1" />
      <Reference URI="">
        <Transforms>
          <Transform Algorithm="http://www.w3.org/2000/09/xmldsig#enveloped-signature" />
        </Transforms>
        <DigestMethod Algorithm="http://www.w3.org/2000/09/xmldsig#sha1" />
        <DigestValue></DigestValue>
      </Reference>
    </SignedInfo>
    <SignatureValue></SignatureValue>
    <KeyInfo>
      <KeyValue></KeyValue>
    </KeyInfo>
  </Signature>
  <samlp:Status>
    <samlp:StatusCode Value="urn:oasis:names:tc:SAML:2.0:status:Success"/>
  </samlp:Status>
  <Assertion ID="<ASSERTION_ID>" IssueInstant="<ISSUE_INSTANT>" Version="2.0"
  xmlns="urn:oasis:names:tc:SAML:2.0:assertion">
    <Issuer><ISSUER_DOMAIN></Issuer>
    <Subject>
      <NameID
        Format="urn:oasis:names:tc:SAML:1.1:nameid-format:emailAddress">
        <USERNAME_STRING>
      </NameID>
      <SubjectConfirmation Method="urn:oasis:names:tc:SAML:2.0:cm:bearer">
        <SubjectConfirmationData
          Recipient="<DESTINATION>"
          NotOnOrAfter="<NOT_ON_OR_AFTER>"
          InResponseTo="<REQUEST_ID>"/>
        </SubjectConfirmation>
      </Subject>
    </Assertion>
  </Response>
```

```
<Conditions NotBefore="<NOT_BEFORE>"
  NotOnOrAfter="<NOT_ON_OR_AFTER>"
  <AudienceRestriction>
    <Audience><DESTINATION></Audience>
  </AudienceRestriction>
</Conditions>
<AuthnStatement AuthnInstant="<AUTHN_INSTANT>">
  <AuthnContext>
    <AuthnContextClassRef>
      urn:oasis:names:tc:SAML:2.0:ac:classes:Password
    </AuthnContextClassRef>
  </AuthnContext>
</AuthnStatement>
</Assertion>
</samlp:Response>
```

La librería xmlseclibs posee licencia BSD de 3 clausulas, en consecuencia es compatible con la licencia GPL del componente y del sistema Joomla.

## Proceso de generación de la claves

---

El proceso de generación de las claves se encuentra ubicado en el fichero ***administrator/components/com\_joomgapps/controllers/generatekeys.php***. El código se divide principalmente en tres funciones correspondientes a las 3 acciones de usuario que existen en la vista de generación de Claves:

- function ***generateKeys()***, genera el par de claves de criptografía asimétrica en el directorio *components/com\_joomgapps/keys*:
  - la clave privada que queda almacenada en dicho directorio, con acceso restringido desde la web
  - la clave pública que debe descargarse el usuario para subirla al panel de control de Google Apps.

El principal problema en el proceso de generación fue el tipo de clave a generar. En un primer lugar utilicé el algoritmo DSA y la extensión OpenSSL de php ( <http://www.php.net/manual/es/book.openssl.php> )

que incorpora numerosas funciones que emulan el comportamiento de la librería OpenSSL ( <http://www.openssl.org/> ).

También probé la generación de claves mediante el algoritmo de cifrado RSA. Posteriormente y tras el problema con la librería xmlseclibs que no tenía implementados métodos para firmar con claves RSA, decidí generar un certificado x.509 para aprovechar la funcionalidad primitiva de la librería. Pronto me percaté que el módulo OpenSSL de php no daba soporte directo para la creación de este tipo de certificados. En consecuencia opté por buscar alguna librería php que sí que lo hiciera. Así pues encontré la librería de clases Crypt\_OpenSSL en el repositorio de clases (<http://www.phpclasses.org/package/4509-PHP-Class-created-for-manipulation-with-ssl-certs-.html> ) PHPClasses.org. Dicha librería también tenía licencia BSD de tres cláusulas, por tanto era compatible con mi componente.

El proceso de generación del certificado, pasaba por el auto-firmado del mismo ya que el certificado creado no estaba reconocido por ninguna autoridad de certificación. Así que también se intentó emular a una CA generando un par de claves y firmando el certificado mediante dichas claves.

- function **publicKeyForceDownload()**, implementa la acción de descarga de la clave pública, para que el usuario pueda enviarla al panel de control de Google Apps
- function **goToGoogleAppsCP()**, redirige al usuario al Panel de Control del

dominio definido en los parámetros Globales. De esa manera incita al usuario a enviar la clave pública descargada.

## **Proceso de Sincronización de Usuarios**

---

El proceso de sincronización de usuarios esta incluido en la vista users del backend del componente. En este proceso están implicados los siguientes scripts:

- ***administrator/components/com\_joomgapps/controllers/users.php***  
implementa las acciones de los diferentes botones del formulario. Sirve de interfaz y los procesos se realizan en el siguiente fichero
- ***administrator/components/com\_joomgapps/helpers/content.php***,  
implementa los procesos de creación, borrado, y migración de usuarios. Para ello necesita contactar con el API de administración de Google Apps. Debido a que la interacción con dicha API es algo compleja, nos apoyamos en el framework php Zend, con licencia new-BSD (compatible con la licencia GPL). En principio tan solo vamos a utilizar la parte del framework que se dedica a interactuar con el API de Google Apps . Actualmente Zend implementa la versión 2.0 del API, ([http://code.google.com/intl/es-ES/googleapps/domain/gdata\\_provisioning\\_api\\_v2.0\\_reference.html](http://code.google.com/intl/es-ES/googleapps/domain/gdata_provisioning_api_v2.0_reference.html)).  
En este fichero se definen funciones para obtener la lista de usuarios en función del filtro establecido:
  - Usuarios de Joomla
  - Usuarios de Google Apps
  - Usuarios de JoomGApps (Usuarios que pertenecen a Joomla y Google Apps)
  - No Usuarios de JoomGApps (Usuarios que pertenecen a Joomla o no pertenecen a Google Apps)

- Filtro contiene ...
- y del número de resultados que se desea obtener (paginación)

Desde dichas funciones se invoca a los siguientes métodos de la librería Zend

- ***Zend\_Gdata\_ClientLogin::getHttpClient***, crea un objeto de autorización de tipo ClientLogin
- ***Zend\_Gdata\_Gapps(\$client, \$domainName)*** crea un objeto de servicio Gapps que nos permite realizar consultas y administrar usuarios en el dominio especificado
- ***Zend\_Gdata\_Gapps::retrieveUser***: método del objeto de servicio que recupera una entrada de usuario
- ***Zend\_Gdata\_Gapps::retrieveAllUsers***: método del objeto de servicio que recupera todas las entradas de usuario
- ***Zend\_Gdata\_Gapps::delete***: método del objeto de servicio que elimina una entrada de usuario
- ***Zend\_Gdata\_Gapps::insertUser***: método del objeto de servicio que crea una entrada de usuario

En el sitio web de Zend podemos encontrar la referencia completa de las funciones a utilizar en la interacción con el API de Administración (<http://framework.zend.com/manual/en/zend.gdata.html>). La principal ventaja que aporta el framework Zend es la abstracción de la interfaz del protocolo de datos de Google, el cual se basa en el Protocolo de Publicación ATOM (<http://tools.ietf.org/html/rfc5023>). Este protocolo permite a las aplicaciones cliente realizar consultas, añadir, borrar y actualizar información web utilizando el estándar HTTP y el formato de

sindicación ATOM. Aunque también podemos destacar la lentitud del protocolo para realizar operaciones extensas. Desconocemos si se debe a la implementación del protocolo por parte de Zend o de Google Apps.

### **Proceso de Comprobación de Requerimientos**

---

Otra vista administrativa del componente es la comprobación de requisitos mínimos para su funcionamiento. Dicha funcionalidad se apoya principalmente en la función `check_joomgapps_system()` perteneciente al fichero `/administrator/components/com_joomgapps/helpers/content.php`. Dicha subrutina devuelve una matriz en la que cada fila representa una comprobación, y las columnas representan:

- resultado de la comprobación
- descripción de la comprobación
- descripción del error

Las comprobaciones realizadas corresponden a los requisitos mínimos para el funcionamiento de las librerías y extensiones php utilizadas:

- Versión PHP 4.0.0 o superior
- Instalación del módulo LIBXML para PHP
- Instalación del módulo CTYPE para PHP
- Instalación del módulo DOM para PHP

Aunque la idea es que puedan extenderse a medida que evolucione la extensión.

### **Creación estructura de soporte Multilenguaje**

---

La extensión soporta dos idiomas:

- `es_ES`: Español (España)
- `en_GB`: Inglés (Reino Unido)

Para posibilitar la capacidad multi-lenguaje del componente todos los mensajes



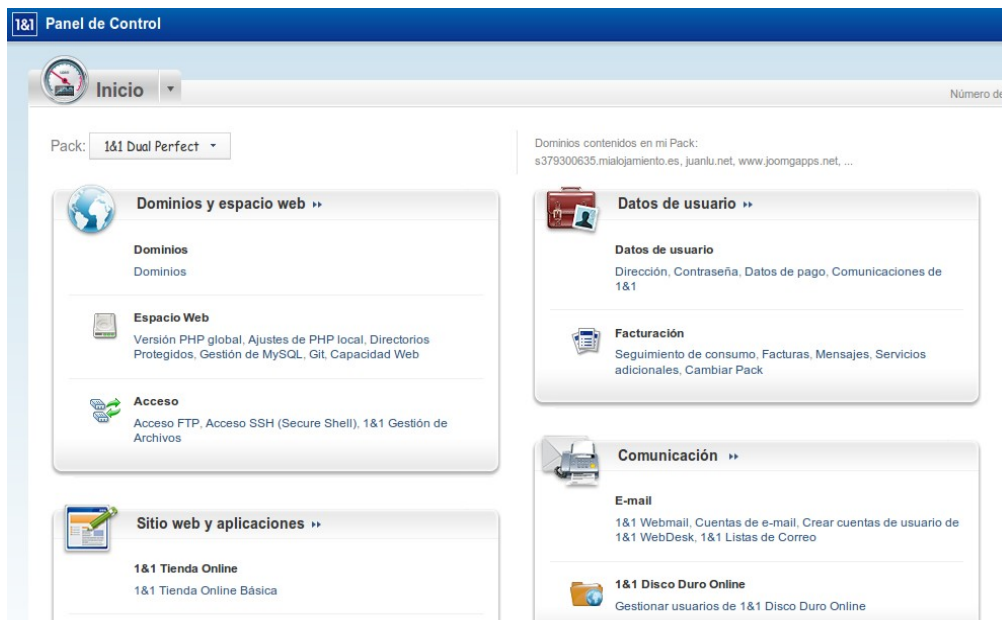
de usuario y cadenas de caracteres visibles están codificadas mediante constantes cuya traducción se almacenan en los ficheros .ini de las carpetas

- language/es-ES
- language/en-GB
- administrator/language/es-ES
- administrator/language/en-GB

## Publicación del componente

Tras implementar el componente y una vez probado en un alojamiento local sobre el dominio Google Apps de mi centro de trabajo, me dispuse a publicar mi trabajo en Internet. Así pues requería de:

- Un alojamiento en Internet que cumpliera los requisitos mínimos. Tras analizar diversas ofertas de Hosting me decante por el producto Dual Perfect del proveedor 1&1. Este me ofrecía dos dominios y 100GB de almacenamiento a un precio realmente increíble.



Captura 32: Panel de Control de ISP 1&1

- Un dominio propietario asociado al componente. Escogí el dominio JoomGApps.net ya que hacía referencia directa al componente, y el dominio de primer nivel .net se relaciona con la red.
- Una cuenta Google Apps versión Bussiness or Education para realizar las demos. Decidí dar de alta una cuenta básica de Google Apps, y aceptar el periodo de pruebas de Google Apps Bussiness Edition para 30 días. Es una solución temporal, pero un tarifa de 10 euros mensuales (para dos usuarios) no puedo permitírmela, al menos de momento. Me planteo la posibilidad de aceptar donaciones vía PayPal que ayuden a mantener esta cuenta premium, o incluso de negociar con Google la gratuidad, ya que mi componente al fin y al cabo promociona y facilita la implementación de sus sistemas
- Un website en Joomla como sitio principal de la extensión. He instalado la versión 1.5.23 descargada de JoomlaSpanish.org. Adicionalmente he añadido las siguientes extensiones
  - Un sistema de traducción multi-lenguaje: JoomFish. Permite crear traducciones de todos los artículos, menús, módulos. De ese modo el sitio web se convierte el multi-lenguaje. En principio existen dos idiomas habilitados y actualizados, de acuerdo con el componente
    - es\_ES: Español (España)
    - en\_GB: Ingles (Reino Unido)

Para poder extender las traducciones al foro de soporte, he tenido que instalar los ficheros de correspondencia xml que permiten que JoomFish traduzca los contenidos de Kunena.



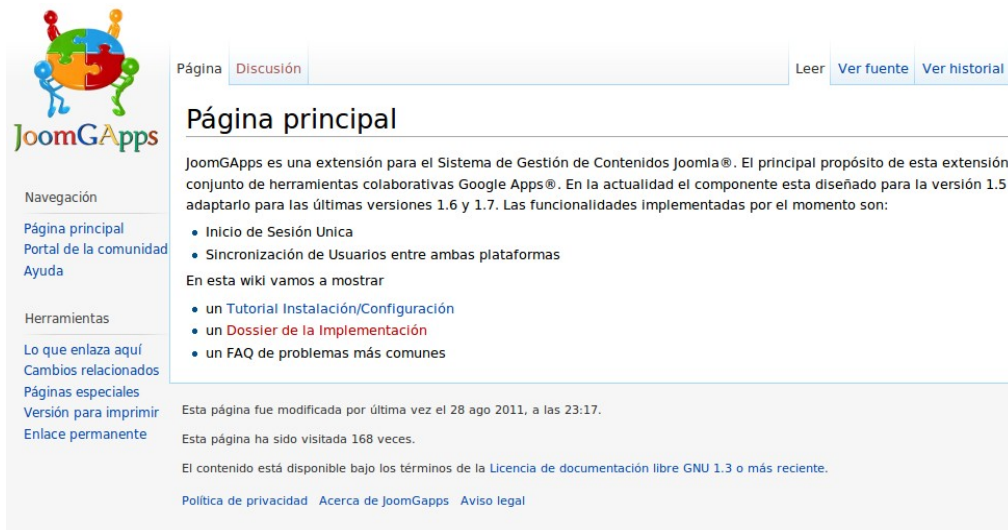
Captura 33: JoomFish Panel de Control

- Adicionalmente diseñe un plantilla específica para el componente que se adaptara al estilo del mismo. Escogí un plantilla con licencia GPL del sitio web Joomla24 (<http://www.joomla24.com/>) y la adapte modificando el código HTML y CSS
- Una cuenta en un repositorio de Proyectos OpenSource para almacenar el código fuente. Para poder mantener de el código de manera colaborativa los proyecto de código abierto utilizan repositorios de código fuente, donde se van publicando las diferentes revisiones, modificaciones, bugs. Para ello se utilizan sistemas de control de versiones tales como Subversión (SVN). Por ser un proyecto de Joomla, me he decantado por el repositorio JoomlaCode. Actualmente solo utilizamos el repositorio para colgar la versiones publicadas y estables, sin utilizar las capacidades de control de modificaciones que nos ofrece subversión.

Nombre del Paquete	Publicación más Reciente	Madurez	Ficheros	Tamaño de
version_1.0	JoomGAppsComponent	5 - Production/Stable	com_joomgapps_1.0.zip	1.38 Mb

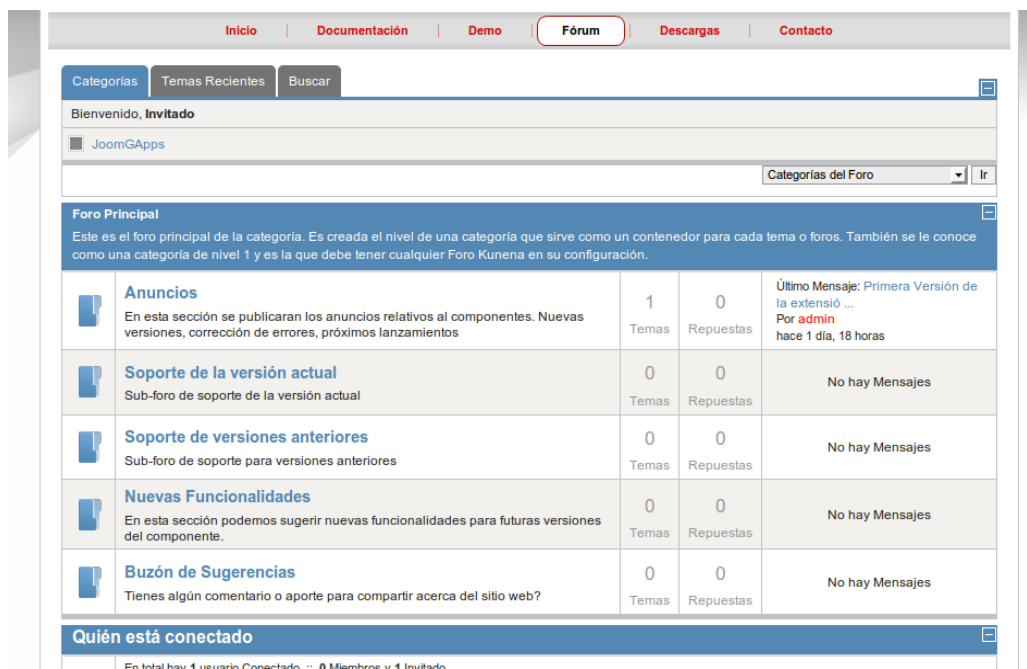
Captura 34: JoomGApps en JoomlaCode

- Un wiki para almacenar la documentación de manera colaborativa, de modo que los futuros miembros de la comunidad puedan aportar información. He optado por instalar el sistema MediaWiki (<http://www.mediawiki.org/>). Para facilitar el acceso a los miembros de la comunidad se ha instalado un sistema de Inicio de Sesión Único entre Joomla y MediaWiki. Esta disponible en el JED a modo de plug-in con el nombre MediaWiki Auto-Login(<http://extensions.joomla.org/extensions/news-production/wiki-integration/8489>). Este sistema aún no está correctamente configurado, quedando a la espera de que el autor nos asesore en los problemas existente. En el wiki en principio hemos planteado tres secciones principales.
  - un Tutorial Instalación/Configuración
  - un Dossier de la Implementación
  - un FAQ de problemas más comunes



Captura 35: Wiki de JoomGApps

- Un foro para resolver dudas y cuestiones acerca del componente. Hemos elegido Kunena por ser uno de los sistemas de foros más populares dentro de la comunidad Joomla.



Captura 36: Foro JoomGApps

- Logotipos que representarán al componente. En principio utilizaremos dos logotipos:
  - Logotipo grande, forma rectangular de dimensiones 880x175px, que servirá como cabecera para el website del componente. Diseñado mediante el programa de retoque fotográfico GIMP, utiliza la fuente y los colores de Google para identificar la funcionalidad de la extensión.

The image shows the large logo for JoomGApps. The text 'JoomGApps' is rendered in a multi-colored, rounded font. The 'J' is green, the first 'o' is orange, the 'm' is blue, the 'G' is red, the 'A' is yellow, and the 'pps' are green. The letters have a slight 3D effect with shadows.

*Captura 37: Logotipo Grande JoomGApps*

- Logotipo pequeño, forma cuadrada de dimensiones 200x200 px, que servirá como simbolo del componente en el JED, el wiki, menu del back-end, etc



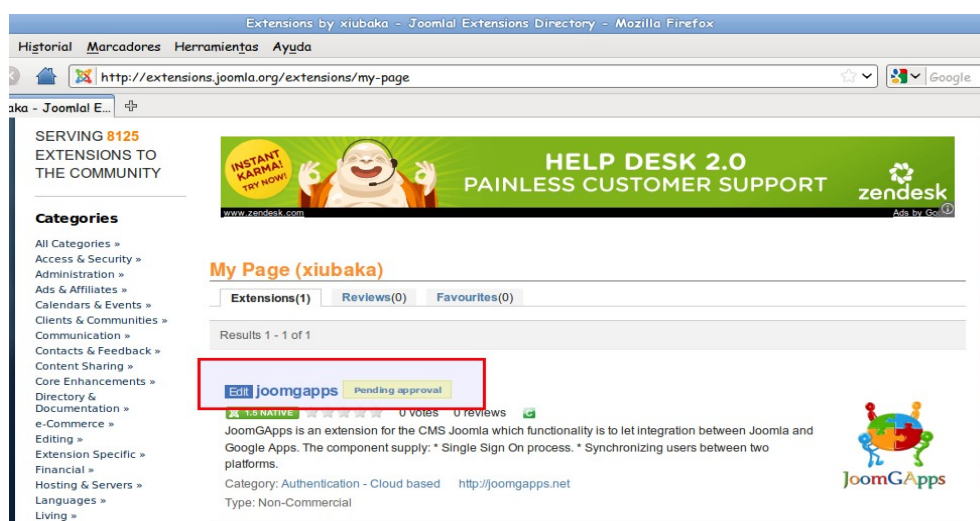
*Captura 38: Logotipo pequeño JoomGApps*

- Solicitar publicación de la extensión en el Joomla Extensions Directory (JED). Esta parte que en principio puede parecer trivial tiene más peculiaridades de las esperadas. El equipo de Joomla tiene muy en cuenta la calidad de las extensiones que anuncia en su repositorio, por ello exige

unos criterios muy estrictos en cuanto a integridad, fiabilidad, usabilidad, portabilidad, distribución, marcas etc. Para ello publica una lista de comprobación con los pasos que debemos verificar antes de solicitar la inclusión del componente en el JED

- Comprobación de requerimientos de publicación [http://docs.joomla.org/JED\\_Entries\\_Submission\\_Checklist](http://docs.joomla.org/JED_Entries_Submission_Checklist)
- Problema con la marca y derivados
  - [http://docs.joomla.org/JED\\_Entries\\_Trademark\\_Checklist](http://docs.joomla.org/JED_Entries_Trademark_Checklist)

En este punto no esperaba tener problemas con el nombre de la extensión, sin embargo OpenSourceMatters la organización que tiene registrada la marca de Joomla (y se encarga de proporcionar soporte legal, financiero y organizativo) restringe el uso de dominios y extensiones que contengan parte de la palabra Joomla. Así que mi extensión al comenzar por Joom necesita la aprobación de esta organización. En el momento de escribir esta memoria estoy a la espera de su aprobación



Captura 39: JoomGApps en JED

## **Promoción de la extensión**

---

El desarrollo de esta extensión tiene su origen en la necesidad de integrar los sistemas Joomla y Google Apps en mi centro de trabajo. Debido a que muchos centros educativos utilizan estos sistemas, considero que un buen modo de promocionar mi componente es contactar mediante correo-electrónico con los coordinadores TIC de algunos institutos. Además antes de desarrollar el componente me encontré en algunos foros de discusión personas que requerían una extensión con esas funcionalidades. Mi tarea de promoción consistirá por tanto en publicitar el componente en dichos foros. También he dado de alta el sitio en herramientas para webmasters de google con la finalidad que los robots de google rastreen de manera más óptima el contenido del sitio y establezcan como palabras clave Joomla, Google Apps e Integration. No obstante de momento el sitio no aparece en las primeras posiciones cuando realizamos búsquedas con dichas palabras.

---

## **Conclusiones**

En la actualidad nadie discute la necesidad de extender los procesos de comunicación corporativa a través de las nuevas tecnologías. Por ello pienso que este componente tendrá gran aceptación, al menos dentro de la comunidad de usuarios de Joomla. En definitiva, quedo satisfecho aprendiendo a desarrollar extensiones para dicho CMS, ya que ello posibilita el desarrollo de sitios web con funcionalidades específicas y utilizando herramientas modulares.



## Bibliografía

- [Learning Joomla! 1.5 Extension Development](#), **Joseph Leblanc**
- [Mastering Joomla! 1.5 Extension and Framework Development: The Professional Guide to Programming Joomla!](#), **James Kennard**
- [Professional Joomla!\(Programmer to Programmer\)](#), **Dan Rahmel**
- [Zend Framework Programmers Reference Guide](#)
- [Google Apps Provisioning API Developer's Guide: Protocol](#)
- [Developers, Joomla Documentation](#)

## Futuras ampliaciones de funcionalidad

Se plantean como posibles extensiones de la funcionalidad del componente las siguientes características:

- Sincronización y mantenimiento de Usuarios
  - Soporte para sub-organizaciones (Google Apps) y Roles (Joomla)
  - Cambio del protocolo de autenticación a OAuth
- Integración de correo-electrónico Gmail
  - Aplicación de correo embebida en el marco principal
  - Vista de listad de distribución disponibles
- Conexión con la Agenda de Contactos
  - Directorio de contactos de la organización en Joomla
- Integración con el Calendario de Eventos
  - Mostrar eventos recientes
  - Añadir/Modificar/Suprimir eventos
  - Lista de calendarios compartidos en la organización

- Documentos compartidos
  - Lista de documentos de la organización
  - Compartición de nuevos documentos
  - Utilización de formularios web de Google para obtención de datos a los usuarios de la organización