



6 Grandes aliados

Revista de Software Libre ATIX 2009

Reconocimiento-Compartir bajo la misma licencia

Usted es libre de:



copiar, distribuir y comunicar públicamente la obra



hacer obras derivadas

Bajo las condiciones siguientes:



Reconocimiento. Debe reconocer los créditos de la obra de la manera especificada por el autor o el licenciador (pero no de una manera que sugiera que tiene su apoyo o apoyan el uso que hace de su obra).



Compartir bajo la misma licencia. Si altera o transforma esta obra, o genera una obra derivada, sólo puede distribuir la obra generada bajo una licencia idéntica a ésta.

- Al reutilizar o distribuir la obra, tiene que dejar bien claro los términos de la licencia de esta obra.
- alguna de estas condiciones puede no aplicarse si se obtiene el permiso del titular de los derechos de autor
- Nada en esta licencia menoscaba o restringe los derechos morales del autor.

Colaboradores

Dirección y Coordinación General

Esteban Saavedra López (jesaavedra@opentelematics.org)

Diseño y Maquetación

Jenny Saavedra López (jennysaavedra@gmail.com)

Esteban Saavedra López (jesaavedra@opentelematics.org)

Revisiones

Esteban Saavedra López

Jenny Saavedra López

Noticias

Ivonne Menacho

Jenny Saavedra López

Marcia Velasquez

Autores Frecuentes

Arnold Guzmán

Carlos Ramos

Ernesto Rico Smith

Esteban Saavedra López

Joseph Sandoval

Rocio Figueroa

Rafael Rendón

Herramientas

La edición de esta revista fue realizada de forma íntegra haciendo uso de Software Libre





**Palabra quechua,
con un sentimiento profundo
y con gran significado filosófico**

El que lo sabe

El que lo intenta

El que lo puede

El que lo logra

Todo proyecto o persona al concebirse, inicialmente esta solo frente al mundo, pero al pasar el tiempo surge la necesidad de interacción y acrecentar la cobertura de acción, algunos proyectos o personas se atreverán a intentarlo solos, algunos lo lograrán, pero en su mayoría fracasarán; mientras que los proyectos o personas que decidan tener aliados serán los que tengan más posibilidades de triunfar y perdurar en el tiempo.

La vida nos enseña que cada día es una continua lucha por la sobrevivencia; lucha que si la enfrentamos solos será más difícil de salir airoso, a lo mejor no imposible pero si muy difícil de conseguir triunfar en la misma, por esto es recomendable que tengamos aliados con nuestros mismos ideales, con nuestros mismos objetivos, para que entre todos nos demos el apoyo que requerimos, el aliento que precisamos rumbo a la victoria que perseguimos.

Grandes aliados, un título que refleja la importancia y el valor de tener grandes aliados; las comunidades de software libre en el mundo son un claro ejemplo de grandes alianzas, donde participan muchísimas organizaciones y personas entre usuarios y desarrolladores, todos ellos con una misma meta, lograr la libertad del software y promover los valores éticos y morales que tanta falta hacen en nuestro mundo.

En éste décimo segundo número ponderamos la realización del **Primer encuentro de la Fundación de Software Libre América Latina** realizada en Venezuela y la Visita de **Richard Stallman**, a **Bolivia**.

Grandes Aliados, conviertete en uno de ellos.

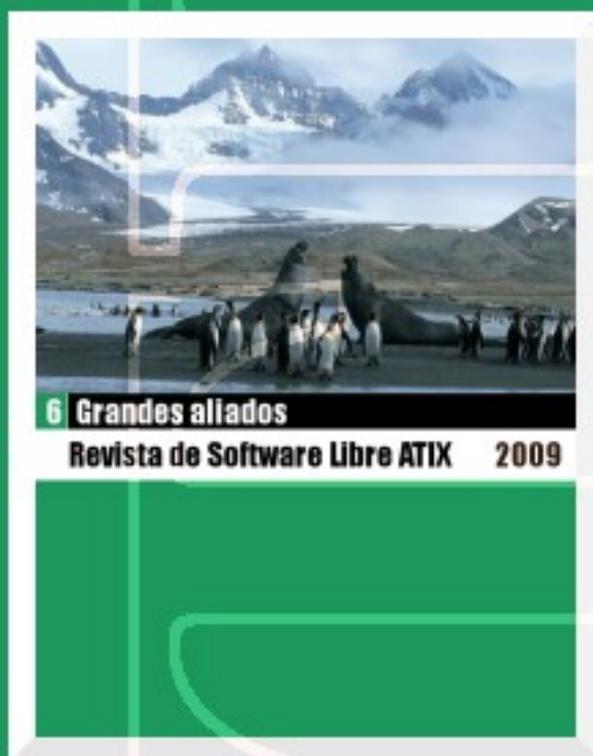
Bienvenidos a nuestro décimo segundo número

Esteban Saavedra López
Director y Coordinador General

Contenido

Liberado el 24 de agosto de 2009

- 7 Plataforma móvil Android
- 15 Escribiendo una aplicación en MonoDevelop (1ra parte)
- 20 Grails: Framework para el desarrollo de aplicaciones Web (5ta Parte)
- 26 Administrando GNU/Linux con Webmin
- 36 Primer Encuentro Fundación Software Libre América Latina
- 42 Declaración de Caracas
- 45 Visita de Richard Stallman a Bolivia
- 52 Willay news
- 60 Comics
- 61 Conociendo lo nuestro - Turismo y Libertad
- 64 Arte Libre
- 66 Información de contacto
- 67 Números anteriores



Plataforma móvil Android

En esta segunda parte trataremos el tema de instalación y configuración del SDK de Android y un pequeño ejemplo práctico.



Introducción

En la entrega anterior tratamos el tema de Android como SO para plataformas móviles. Una vez abordada la teoría, se antoja algo más tangible, un ejemplo práctico me parece ideal para apreciar las posibilidades que ofrece esta plataforma para los desarrolladores.

Nos enfocaremos a la instalación y configuración del Android SDK (en GNU/Linux). Posteriormente se desarrollará un ejemplo práctico sencillo, no es algo grande y mucho menos complicado ya que el objetivo es únicamente mostrar la forma de trabajar con las herramientas de desarrollo. Sin más, manos a la obra.

Instalación General

Requerimientos de sistema

El SDK soporta a los 3 sistemas operativos más conocidos:

- ✓ Windows XP (32-bit) o Vista (32- o 64-bit)
- ✓ Mac OS X 10.4.8 o posterior

(únicamente x86)

- ✓ Linux, Se recomienda ubuntu, sin embargo he comprobado su funcionalidad también en OpenSuse y en Debian Lenny (el que utilizo para este documento) .

Las opciones de entorno necesarios son:

- ✓ Eclipse 3.3 (Europa), 3.4 (Ganymede) , este elemento es opcional.
- ✓ El JAVA JDK en su versión 5 o 6, es necesario instalar el kit de desarrollo completo.
- ✓ Plugin ADT(Android Development Tool), opcional.

Nota: El compilador gcj de GNU no es compatible con el Android SDK , por tanto, no funcionará. En si los requerimientos son razonables, en cuanto a la capacidad de computo del sistema está claro que cuanto más recursos existan más eficiente será el desempeño.

Opciones

Android proporciona un SDK para desarrollar aplicaciones. Además de esto, también han creado un plugin para el IDE eclipse con el que el desarrollo de aplicaciones se facilita considerablemente. Esto último es opcional y podemos prescindir de ello.

En otras palabras, tenemos 2 opciones para crear nuestras aplicaciones, Trabajando directamente con el SDK o mediante el plugin para eclipse que puede facilitarnos mucho las cosas. Depende de cada uno la opción a

seguir, en este documento describiremos la instalación del plugin para eclipse, y el ejemplo práctico lo desarrollaremos usando esta opción.

La instalación paso a paso

Antes que nada debemos checar si contamos con lo requerimientos mínimos, en el caso de JAVA, apoyense de su administrador de paquetes para instalarlo. Una vez que tenemos todo listo vamos a realizar la descarga del Android SDK para lo cual nos dirigimos a la página oficial del proyecto en <http://developer.android.com/> y descargamos la versión 1.5 del SDK, son unos 170 mb aproximadamente. Esperamos unos cuantos minutos para que descargue y continuamos.

Una vez descargado el paquete, los guardamos en un directorio especial en el que queremos instalar el software, puede ser el que ustedes gusten, después extraemos el paquete. Supongamos que el paquete se ha descargado en el escritorio y que el directorio de instalación será `/usr/local/Android`.

Nota: Los comando iniciados con `#` indican usuario root y las que inician con `$` indican a un usuario normal.

```
# cp $HOME/Desktop/android...x86-
x.y.zip \
/usr/local/
# cd /usr/local
# unzip android-sdk-linux_x86-1.5_r3.zip
# mv android-sdk-linux_x86-1.5_r3 Android
# chown -R usuario Android/
```

En los comandos, `$HOME` es la variable de entorno que apunta al directorio personal de cada usuario. Las comandos anteriores han creado el directorio `/usr/local/Android` en el cual se encuentran todas las herramientas y archivos del SDK, en el subdirectorio `tools` se encuentran los ejecutables tales como el

emulador, el compilador, etc. En la última línea(comando `chown`) cambiamos el propietario del directorio base de Android, donde usuario es la cuenta de usuario de ustedes. Esto es solo en caso de haber usado un directorio propiedad de usuario root o cualquier otro usuario donde no tengamos derechos de lectura y escritura, si su directorio se encuentra en en su directorio personal lo más probable es que no sea necesaria la última línea.

El siguiente paso a realizar será hacer que el sistema reconozca la herramientas del SDK, lo cual lograremos registrando el subdirectorio `tools` a la variable de entorno `PATH`.

```
$ vim $HOME/.bashrc
.
.
#Al final del archivo agregamos
#la siguiente linea

export PATH=$
{PATH}:/usr/local/Android/tools
```

Después de agregar la línea de comando, guardamos y salimos del VIM. Si queremos que la configuración se actualice de inmediato utilizamos el comando `source` con el archivo `.bashrc` y listo.

```
$ source $HOME/.bashrc
```

Con ésto ya podemos ejecutar los programas contenidos en el directorio `tools` desde cualquier otro lugar donde tengamos guardado nuestro proyecto. También desde cualquier otro directorio pueden ejecutar. por ejemplo, el comando emulador y podrán ver que corresponde a un programa del Android SDK. Una vez realizado lo anterior y si no ha surgido otro inconveniente el Android SDK ya se encuentra instalado y listo para usarse.

Instalación del plugin para Eclipse

Si bien ya está instalado el SDK, éste nos proporciona únicamente las herramientas básicas, como el compilador, el emulador, documentación, algunos ejemplos, etc. Pero no trae consigo herramientas de edición de código, por ejemplo, un IDE; en vista de eso vamos a configurar el IDE Eclipse para que cumpla ese objetivo. Existe un plugin para eclipse que permite trabajar con el Android SDK, y de esta forma contar con un ambiente más amigable de cara al desarrollo de aplicaciones sobre Android.

La versión de eclipse que utilizo para este documento es eclipse 3.4 Ganymede, que pueden ustedes conseguir en <http://eclipse.org>, doy por hecho que ya cuentan con el programa instalado en su sistema.

Una vez que abran eclipse nos vamos rápidamente al menú help->software Updates donde aparece una ventana como la que se muestra en la imagen, en la pestaña Available Software, nos da la opción de agregar nuevas fuentes, sitios.

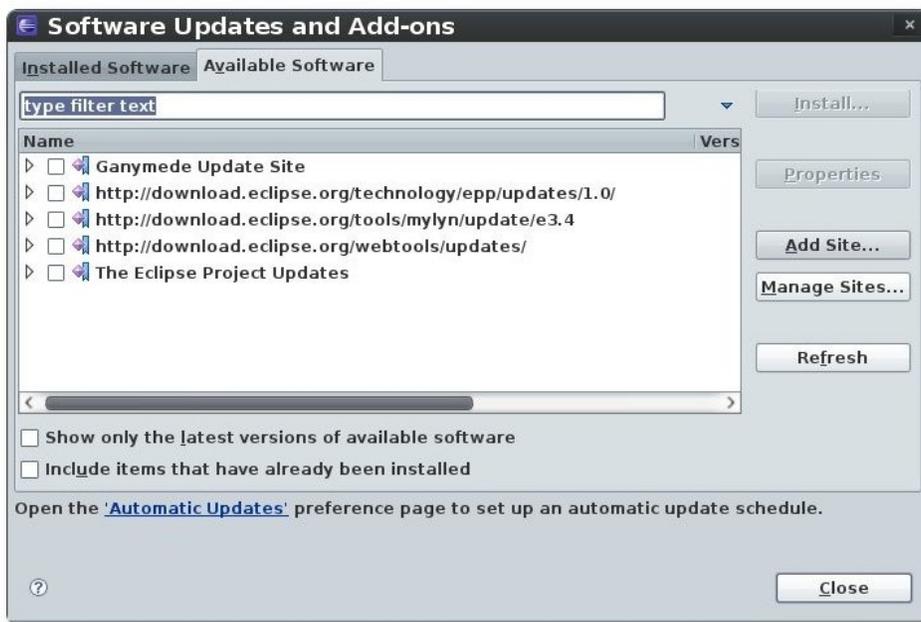


Gráfico 1. Agregar fuentes de software a eclipse.

Seleccionamos pues la opción Add Site e introducimos esta dirección <https://dl-ssl.google.com/android/eclipse/>, si les marca algún error pueden cambiar https por http simplemente, aceptamos para cerrar la ventana.

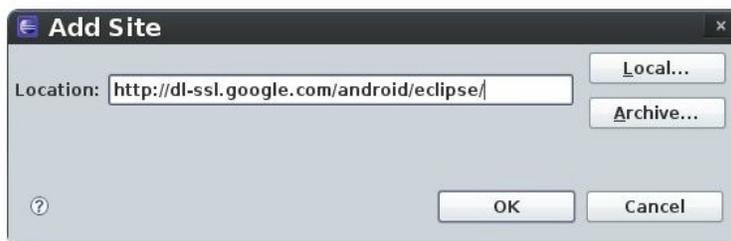


Gráfico 2. Sitio donde se encuentra el Plugin.

Posteriormente nos aparece una nueva fuente de software en la lista de software disponible que corresponde con la dirección que agregamos, seleccionamos todo el contenido de esa fuente y procedemos a instalar.



Gráfico 3. Herramientas necesarias para instalar.

Después de iniciar la instalación esperamos un momento a que descargue los componentes necesarios, después habrá que aceptar algunas licencias y listo, a instalar...Una vez que termine la instalación el programa nos pedirá reiniciar el programa, lo cual aceptamos y eclipse se reiniciará.

Ya se encuentra instalado el plugin, ahora hay que ajustar algunas configuraciones. Tenemos que indicarle a eclipse donde se encuentra el Android SDK para que pueda realizar su trabajo. Nos vamos Window->Preferences y en la parte izquierda de la ventana seleccionamos lógicamente la opción "Android", nos saltarán algunas ventanitas, una de agradecimiento por parte de Android, otra que nos indica que falta señalar la dirección del Android SDK, acepten y listo, si la ventana de agradecimiento no se cierra no se preocupen, arrastrenla donde no estorbe y a continuación eligen la ruta donde tenemos nuestro SDK, en nuestro caso en /usr/local/Android.

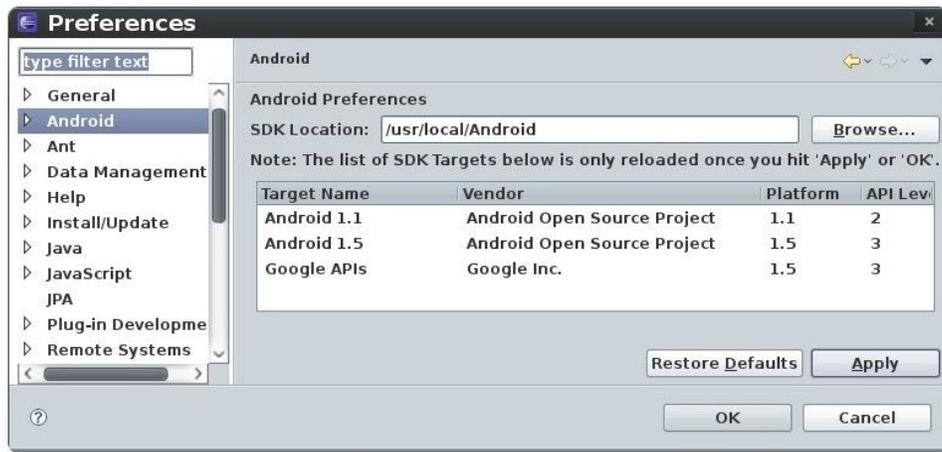


Gráfico 4. Configuración del SDK con Eclipse.

Damos clic en Aplicar, si la ruta está correcta nos desplegará información de los componentes con que se cuenta. Posteriormente aceptamos y salimos de la ventana de configuración. YA ESTA TODO LISTO!!.

Ejemplo práctico (Hola Mundo)

Los proyectos de aplicaciones son por lo general muy extensos, realizar una aplicación lo más sencilla posible nos abarcaría un buen de espacio tanto en la explicación como en el código. Es por eso que para este ejemplo práctico he decidido únicamente realizar el famoso “Hola Mundo!” para la plataforma Android. Es cierto que es muy trivial, pero por lo anteriormente comentado y porque en mi opinión es suficiente para empezar, “Hola Mundo!” y “Hola Mundo!!!!”, ni hablar XD.

Como es de pensar, las aplicaciones que desarrollemos para Android están pensadas para ser corridas en dispositivos móviles. Pero está claro que no vamos a probar nuestra aplicación a cada instante en un dispositivo con el cual es posible no contemos, para ésto dentro del SDK se incluye un emulador el cual podemos utilizar tal cual un dispositivo móvil para probar las aplicaciones que desarrollemos. Antes de poder utilizar el emulador tenemos que crear un Android Virtual Device (AVD), una unidad virtual que utiliza un emulador como para anclar su sistema, parecido a las máquinas virtuales.

Crear un AVD

Abrimos una terminal y desde cualquier directorio ejecutamos el siguiente comando:

```
$ android create avd --target 3 \  
  --name mi_avd
```

La opción `--target` con el valor 3 especifica que para el nivel de API se crea la AVD, éste se puede ver en el gráfico 6. La opción `--name` obviamente indica el nombre de la AVD, pueden usar algún otro nombre.

Crear un nuevo proyecto para Android

Para crear un nuevo proyecto para Android en eclipse, nos dirigimos a file->new->project en la ventana que nos aparezca seleccionamos la opción correspondiente y elegimos continuar.

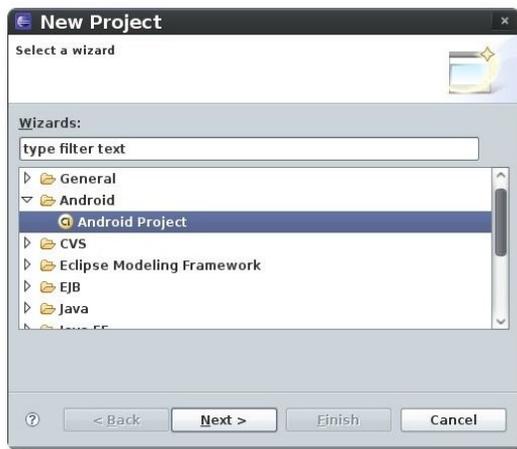


Gráfico 5. Tipo de proyecto a realizar.

A continuación tenemos que llenar algunos campos correspondientes al proyecto.

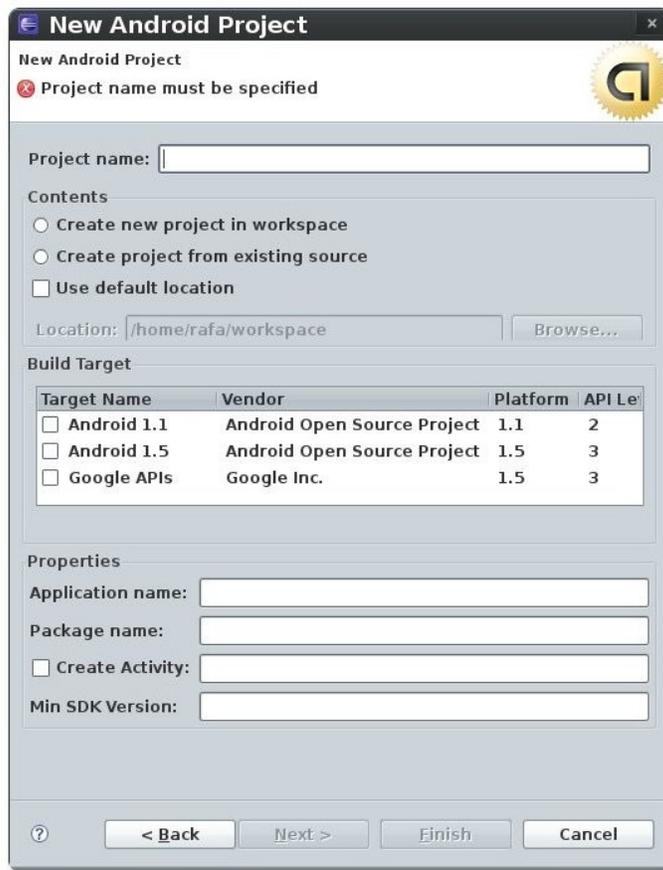


Gráfico 6. Datos del proyecto.

Para nuestro ejemplo utilizaremos los siguientes datos para el proyecto:

```
Project name: HolaMundo
Application name:  Hola Mundo de Android
Package name: com.ejemplo.holamundo
Create activity:  HolaMundo
Min SDK Version:  3
```

A continuación algo de explicación sobre los datos con que llenamos los campos del proyecto:

- ✓ **Project name:** Claramente es el nombre del proyecto.
- ✓ **Application name:** Es el título que muestra la aplicación cuando se ejecuta.
- ✓ **Package name:** El paquete donde se empaquetará la clase creada.
- ✓ **Create activity:** Nombre de la clase que el plugin generará de default.
- ✓ **Min SDK Version:** Nivel de la API, 3 para el SDK 1.5

Después de llenar los campos del proyecto elegimos la opción finalizar y listo. En unos instantes se nos presenta en la ventana principal de eclipse el proyecto recién creado con un archivo de código que contiene el esqueleto de nuestra aplicación. En el lado izquierdo de la ventana

principal se desglosan los componentes del proyecto.



Gráfico 7. Estructura del proyecto y sus componentes.

El código que se genera por defecto es el siguiente:

```
package com.ejemplo.holamundo;

import android.app.Activity;
import android.os.Bundle;

public class HolaMundo extends Activity {
    /** Called when the activity is first created. */
    @Override
    public void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.main);
    }
}
```

Código 1. Código fuente generado automáticamente por eclipse.

Bueno, algo de explicación. Para todo lector con conocimientos mínimos sobre JAVA (lo cual doy por hecho) no será difícil entender estas líneas de código, es una simple clase, que de momento no hace gran cosa. Al código anterior hagamos las siguientes modificaciones y posteriormente explicamos ya el código resultante.

```
package com.ejemplo.holamundo;

import android.app.Activity;
import android.os.Bundle;
import android.widget.TextView;

public class HolaMundo extends Activity {
    /** Called when the activity is first created. */
    @Override
    public void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        TextView tv = new TextView(this);
        tv.setText("Bienvenido al Mundo de Android!
        \n\t\tSaludos ATIX!");
        setContentView(tv);
    }
}
```

Código 2. Código después de la modificación.

El código subrayado es lo que se ha modificado del código inicial. Lo que hemos hecho es importar una clase más, la clase `TextView`, la cual nos permite mostrar texto en la aplicación. Después ya en la clase, creamos un objeto de la clase `TextView` pasándole como parámetro la clase actual, posteriormente agregamos el texto que será mostrado en la aplicación, y por último aplicamos los cambios en la línea 18. No hay mucho más que explicar, esto es todo para este ejemplo.

Pasemos a ejecutar el programa, nos dirigimos a **Run->Run configurations**, ahí elegimos la AVD a utilizar(en caso de que no esté seleccionada) , elegimos la opción Aplicar y posteriormente en Ejecutar(Run). Se iniciará el emulador, tengan paciencia esto puede tardar un buen rato. Si todo ha salido bien la siguiente imagen muestra el resultado.

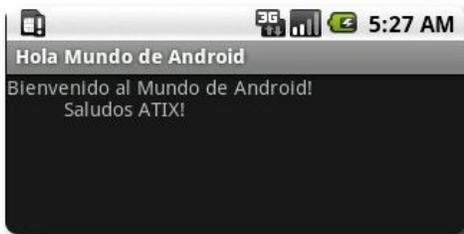


Gráfico 8. El resultado.

Con esto hemos terminado con nuestra primera aplicación para la plataforma Android, con la cual hemos aprendido como elaborar proyectos de una forma fácil y rápida con la ayuda de Eclipse. Disfruten su creación y continúen practicando :D.

Conclusión

En este artículo hemos aprendido como iniciarse en la programación de aplicaciones para la nueva plataforma de dispositivos móviles, Android. Con lo explicado aquí es suficiente para entender el funcionamiento básico del SDK que proporciona Android. Espero que les haya gustado el tema y nos vemos la próxima.

Referencias

[1] <http://developer.android.com/>

Autor



Rafael Rendón Pablo

Estudiante de Ingeniería en Sistemas Computacionales

Usuario de software libre

smart.dyr@gmail.com

www.inforscience.260mb.com



Escribiendo una aplicación en MonoDevelop (1ra Parte)

Mono es el nombre de un proyecto de código abierto iniciado por Ximian y actualmente impulsado por Novell, para crear un grupo de herramientas libres, basadas en GNU/Linux y compatibles con .NET



¿Qué es MonoDevelop?

MonoDevelop un IDE libre de GNOME primariamente diseñado para C# y otros lenguajes .NET, aunque permite abrir cualquier otro tipo de lenguaje. MonoDevelop busca algo más que sólo ser un IDE: intenta ser una plataforma extensible sobre la cual cualquier tipo de herramienta de desarrollo pueda ser construida.

MonoDevelop fue originalmente una migración del IDE SharpDevelop, escrito en Windows Forms para Microsoft Window, a Gtk#, pero ha evolucionado mucho desde esa migración inicial. Hay numerosos objetivos que MonoDevelop busca lograr. Algunos de esos son:

Crear el mejor ambiente de desarrollo para sistemas Unix en Mono.

Debido a que está escrito en Gtk#, y que existe un buen soporte para Gtk#, busca

agregar funcionalidad para mejorar la experiencia en Gtk#.

Integrar herramientas que han sido construidas hasta ahora, cosas como MonoDoc, NUnit-Gtk y el depurador deben apuntar a MonoDevelop.

Escribiendo una aplicación en MonoDevelop

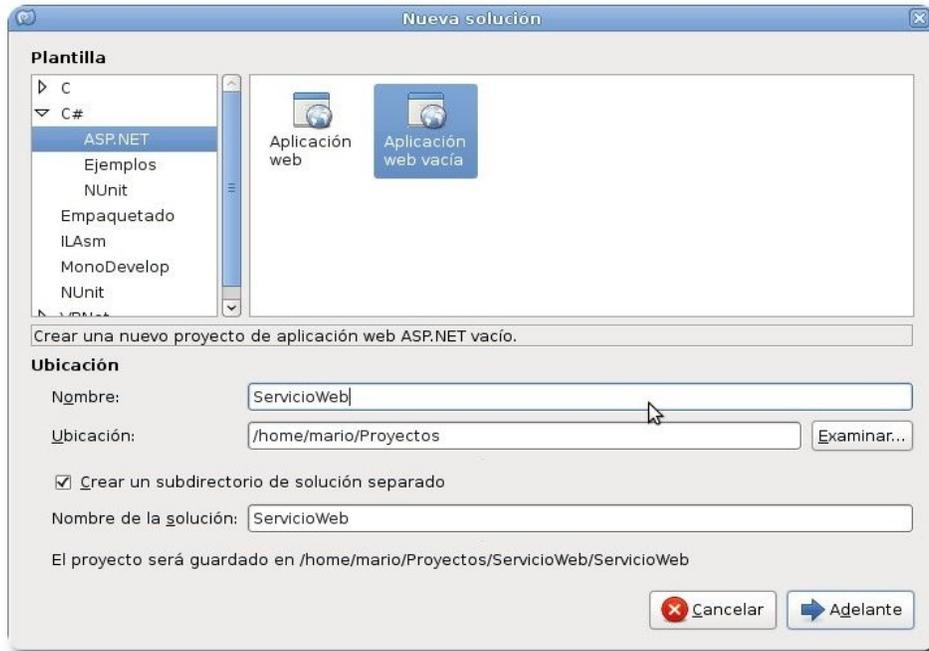
Esta es la primera de dos entregas para crear una aplicación utilizando MonoDevelop. Escribir una aplicación en MonoDevelop es sencillo, el único requerimiento inicial es saber programar en C#, no es necesario tener experiencia previa con IDEs, pero si se tiene experiencia utilizando Visual Studio .NET es una ayuda extra pues MonoDevelop funciona de una forma semejante.

Este tutorial está dividido en dos entregas, en la primera vamos a diseñar un servicio web que permitirá sumar dos números enteros, y en la segunda vamos a diseñar una interface gráfica utilizando Gtk# en la cual llamaremos a nuestro servicio web para sumar dos números enteros, de esta forma tendremos una aplicación completa que utiliza un servicio web a través de una aplicación Gtk#.

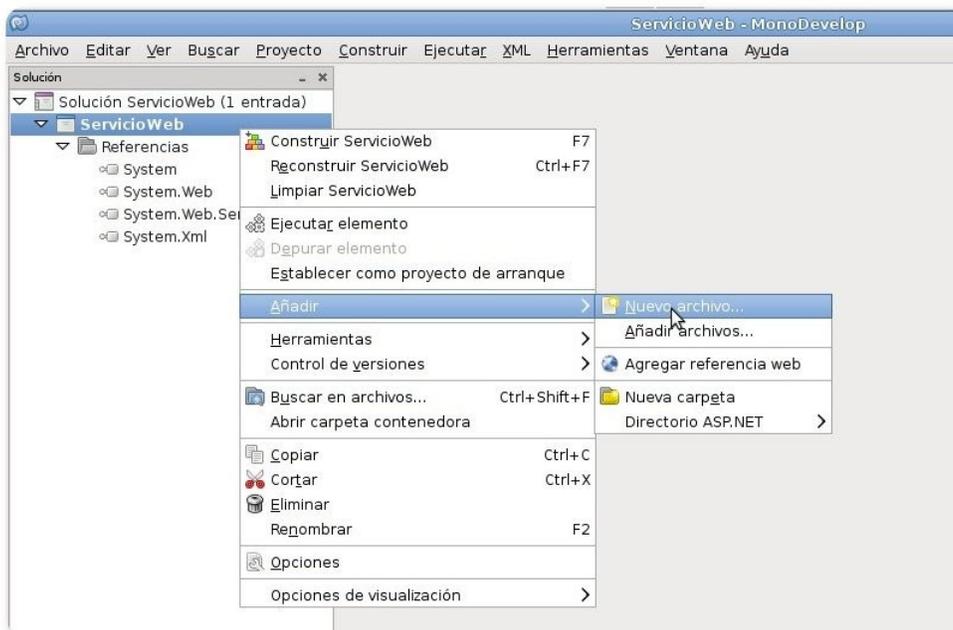
Todos estos ejemplos fueron escritos en OpenSUSE 11.1 utilizando MonoDevelop 2.0.

Escribiendo un servicio web

Comencemos por abrir MonoDevelop. Vamos a crear una solución nueva que será la encargada de contener nuestro servicio web, para esto seleccionaremos Archivo, Nuevo, Solución:



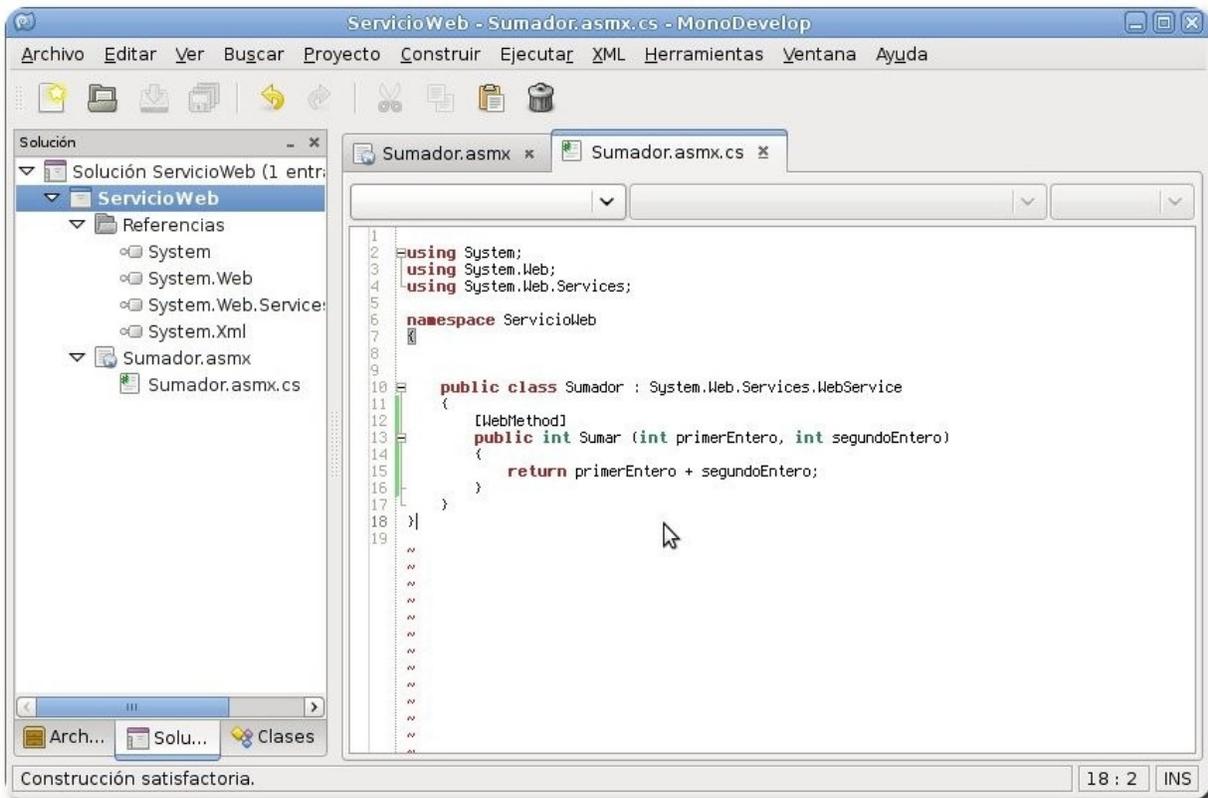
y seleccionaremos Aplicación web vacía, nombremos este proyecto como ServicioWeb, a continuación apretaremos el botón Adelante, en la siguiente ventana no seleccionamos opción alguna, dejamos todo con los valores por defecto, pues no requerimos, por el momento, ninguna de las opciones disponibles. Después de apretar Aceptar, tendremos una solución ASP.NET, ésta nos servirá para crear nuestro servicio web. No existe contenido en nuestra solución, lo que debemos hacer ahora es agregar un archivo de servicio web, para esto debemos seleccionar nuestro proyecto ServicioWeb, justo debajo de la solución ServicioWeb y apretar el botón derecho para mostrar un menú desplegable



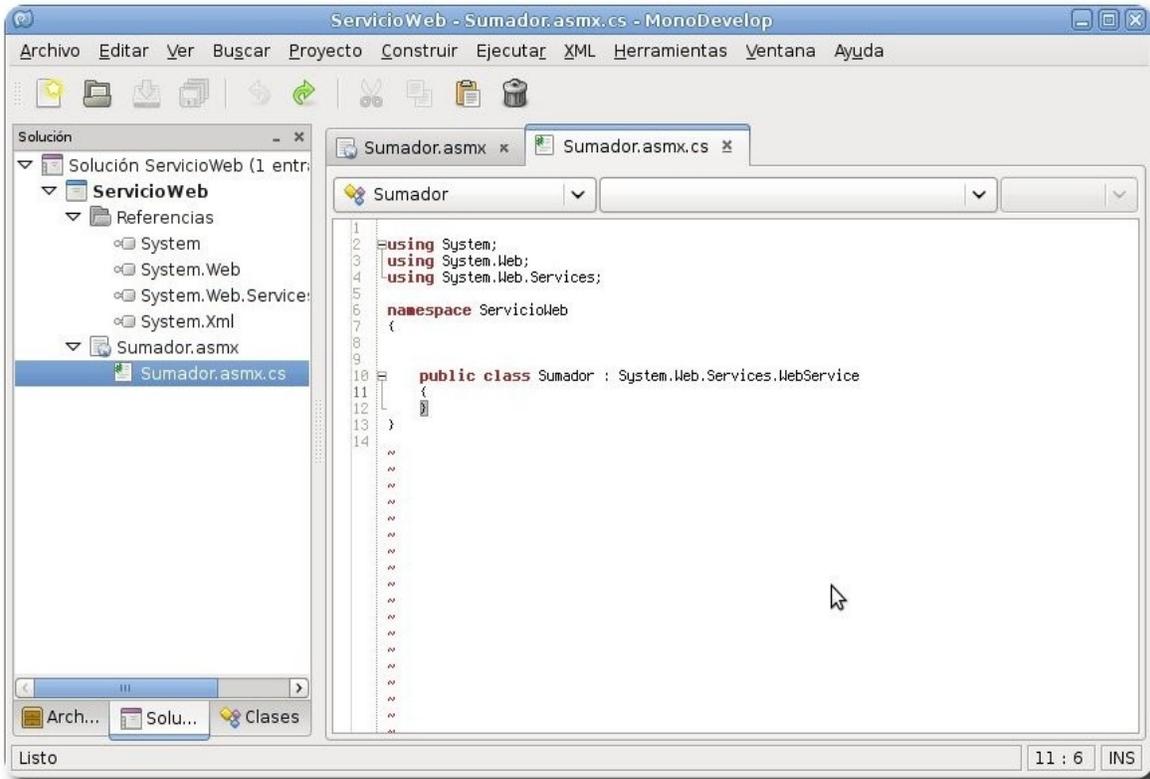
seleccionar Nuevo Archivo, lo cual abrirá una ventana nueva, en ésta seleccionaremos ASP.NET de la lista que se encuentra a la izquierda, moveremos la barra desplegable de la derecha y seleccionaremos Servicio web con CodeBehind, utilicemos Sumador como nombre del servicio web y apretemos Nuevo:



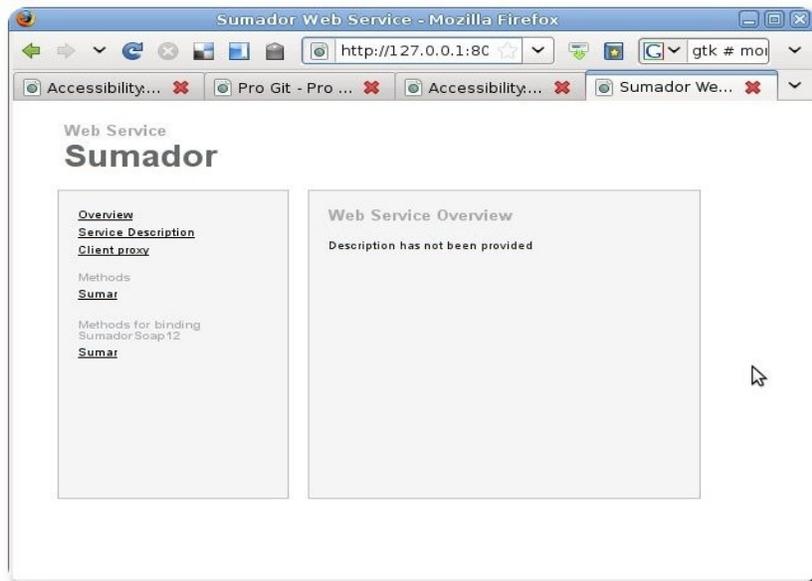
Esto creará nuestro servicio web, y agregará dos archivos (**Sumador.aspx** y **Sumador.aspx.cs**), nuestra lógica la escribiremos en el archivo **Sumador.aspx.cs**:



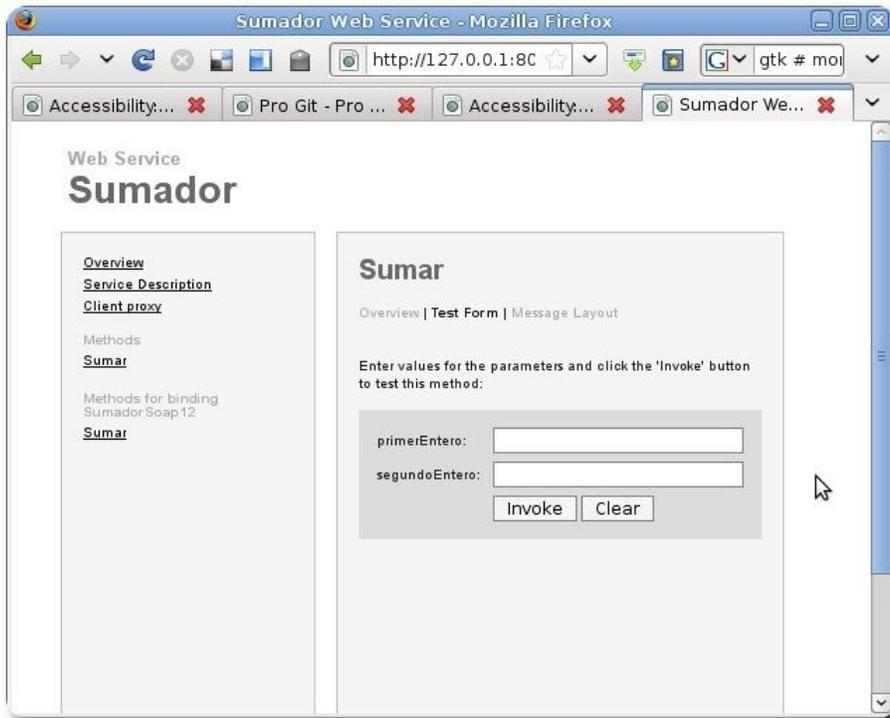
En este archivo agregaremos un método llamado: Sumar, el cual sumará dos enteros y regresará el resultado:



Dos cosas son importantes al definir este método: Primero, que el método debe estar decorado con el atributo `WebMethod`, de esta forma se sabe que es un método web, y segundo que debe ser un método público, pues si no lo es a pesar de ser un método web éste no será accesible por el exterior. A continuación debemos probar que todo funciona correctamente, para esto seleccionaremos la opción `Ejecutar` desde la barra de herramientas `Ejecutar` o simplemente podemos apretar `F5`. Esto abrirá nuestro navegador cargando la dirección `http://127.0.0.1:8080/`, la cual mostrará un error, esto es válido porque nuestro servicio no se encuentra en la raíz del proyecto, sino que nuestro servicio web se llama `Sumador.asmx`, el mismo nombre que se generó al inicio, bastará con cambiar nuestra dirección web a `http://127.0.0.1:8080/Sumador.asmx` y probar de nuevo, lo cual mostrará lo siguiente:



A la izquierda de esta página web de prueba existe nuestro método web, el que definimos dentro de MonoDevelop, llamado Sumar, para probarlo hay que apretar el enlace del método y luego seleccionar la opción Test Form, la cual mostrará un formulario con entradas de texto con los nombres de los argumentos que utilizamos cuando definimos nuestro método:



Prueba escribiendo algunos valores enteros dentro de las cajas de texto y aprieta el botón Invoke para que veas la respuesta del servicio web. ¡Felicidades has escrito un servicio web utilizando MonoDevelop!

Referencias

- [1] <http://www.mono-project.com/>
- [2] <http://projects.gnome.org/tomboy/>
- [3] <http://banshee-project.org/>
- [4] http://beagle-project.org/Main_Page
- [5] http://f-spot.org/Main_Page
- [6] <http://do.davebsd.com/>
- [7] <http://www.ecma-international.org/>
- [8] <http://www.mono-project.com/Contributing>
- [9] <http://www.mono-project.com/StudentProjects>
- [10] http://mono-project.com/User_Groups

Autor



Mario Carrion
 Software Engineer
 mario.carrion@gmail.com
<http://www.mariocarrion.com/>



Grails: Framework para el desarrollo de aplicaciones Web (5ta Parte)

Grails es un framework orientado al desarrollo de aplicaciones web de forma sencilla, rápida y divertida.



Introducción

Dentro el proceso de desarrollo de aplicaciones web, la autenticación es uno de los más importantes, ya que éste permitirá, realizar los controles de acceso por parte de los usuarios a las distintas partes del sistema.

Autenticación

por lo general este proceso consiste, en desplegar un formulario donde el usuario deberá acceder su nombre de usuario y su contraseña, estos datos son enviados al servidor y contrastados con alguna entrada en una base de datos, para luego permitir o denegar el acceso en caso de que correspondan o no.

Hashing

Por lo general las contraseñas son sometidas a un hash antes de ser almacenadas como una forma de precautelar la seguridad de las mismas.

En Grails una técnica común para realizar esta tarea es usar `DigestUtils` en el paquete `org.apache.commons.codec.digest.DigestUtils`. Esta clase contiene una variedad de

métodos para producir diferentes tipos de hashes, entre los más usados se encuentran:

- ✓ **`DigestUtils.md5(java.lang.String data)`**: Este método produce un elemento de 16 bytes usando el mecanismo MD5
- ✓ **`DigestUtils.md5Hex(java.lang.String data)`**: Este método crea una cadena hexadecimal de 32 caracteres usando el mecanismo MD5
- ✓ **`DigestUtils.shaHex(java.lang.String data)`**: Este método crea una cadena hexadecimal de 32 caracteres pero usa el mecanismo SHA-1.

Control de acceso

Dentro de este proceso, se consideran los siguientes elementos:

- ✓ Validación de la sesión
- ✓ Usuarios
- ✓ Roles
- ✓ Permisos
- ✓ Reglas

Captcha



Son representaciones gráficas de una cadena alfanumérica generada aleatoriamente, y que sirve para validar los datos introducidos dentro un formulario, así

mismo también previenen el spam en los formularios.

Proceso de autenticación

Para crear una solución de seguridad para nuestras aplicaciones web, debemos considerar los siguientes aspectos dentro su diseño e implementación

- ✓ Registro de usuarios
- ✓ Proceso de login
- ✓ Registros de acceso y salida de usuarios
- ✓ Asegurar los controles

Registro de usuarios

Para ésto debemos generar el formulario de registro (**registro.gsp**), una acción para el

Para crear el controlador de login, debemos ejecutar:

```
grails create-controller login
```

Dentro el archivo **grails-app/controllers/LoginController.groovy** debemos incluir el siguiente código:

```
class LoginController {
    def index = { redirect(action:login,params:params) }

    def login = {}

    def manejaLogin = {
        def user = User.findByUsuarioAndPasword(params.usuario, params.password)
        if (!user) {
            flash.message = "Usuario no encontrado: ${params.usuario}"
            redirect(action:'login')
            return
        }
        else
        {
            session.user = user
            redirect(controller:'evento')
        }
    }

    def logout = {
        if(session.user) {
            session.user = null
            redirect(action:'login')
        }
    }
}
```

registro dentro el UserController. Adicionalmente haremos uso de un CAPTCHA, para lo cual debemos instalar el plugin correspondiente (<http://www.boomchucka.com/grails/grails-Captcha-0.5.zip>)

```
grails install-plugin
http://www.boomchucka.com/grails/grails-
Captcha-0.5.zip
```

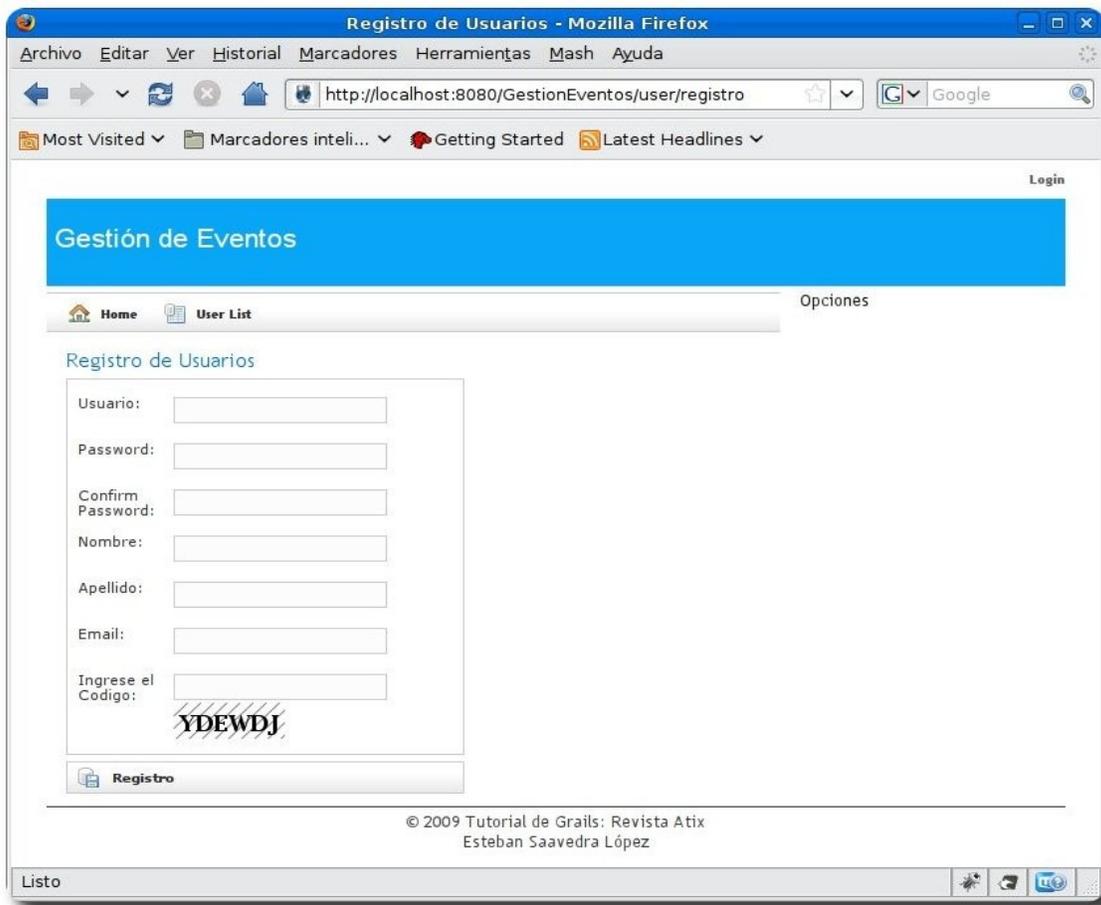
Proceso de Login

Para establecer un proceso de login dentro de nuestra aplicación, debemos realizar las siguientes actividades:

- ✓ Crear el controlador de login
- ✓ Crear el formulario de login
- ✓ Registrar la sesión del usuario

Este código básicamente permite manejar las acciones de login y logout, registrando o anulando el registro de la sesión del usuario respectivamente.

Para que este controlador pueda funcionar correctamente precisamos crear una vista con el siguiente contenido:



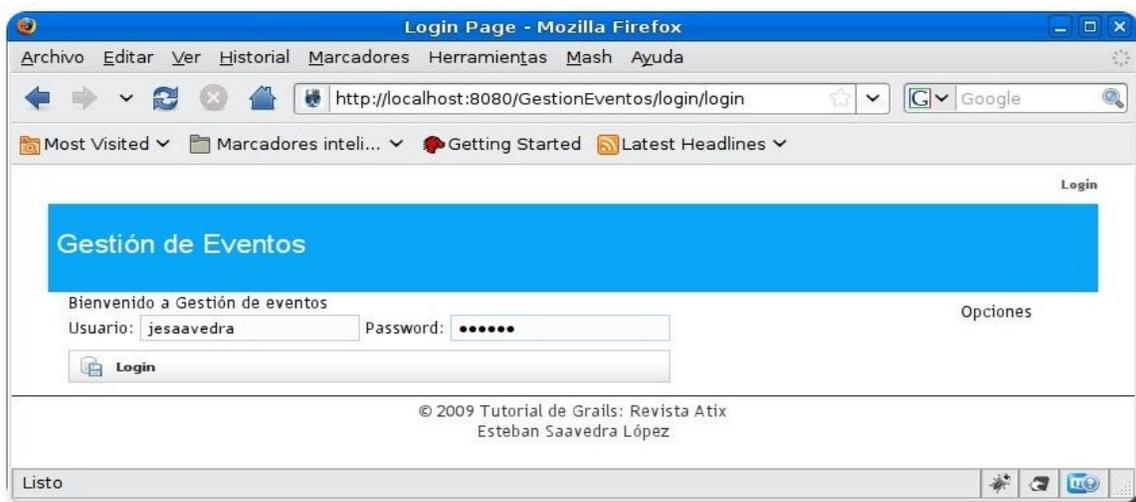
```
<html>
<head>
<title>Login Page</title>
<meta name="layout" content="main" />
</head>
<body>
<div class="body">
<g:if test="${flash.message}">
<div class="message">
${flash.message}
</div>
</g:if>
<p>
Bienvenido a Gestión de eventos
</p>
<form action="manejaLogin">
<span class='nameClear'><label for="usuario">Usuario:</label>
</span>
```

```

        <td valign="top" class="value $
{hasErrors(bean:userInstance,field:'usuario','errors')}>
        <input type="text" id="usuario" name="usuario" value="$
{fieldValue(bean:userInstance,field:'usuario')}"/>
        </td>
        <span class='nameClear'><label for="password">Password:</label>
        <td valign="top" class="value $
{hasErrors(bean:userInstance,field:'password','errors')}>
        <input type="password" id="password" name="password" value="$
{fieldValue(bean:userInstance,field:'password')}"/>
        </td>
        <div class="buttons">
        <span class="button"><input class="save" type="submit" value="Login"
/></span>
        </div>
    </div>
</form>
</div>
</body>
</html>

```

El resultado de esta vista la podemos observar en el siguiente gráfico:



Una vez que el usuario haya introducido los datos correctos de usuario y contraseña, estos deben ser registrados en la sesión, adicionalmente podemos desplegar los datos del usuario en la sección TopBar de la pantalla y claro está desplegar la opción de logout; para esto debemos realizar algunas modificaciones en la vista del TopBar, cuyo contenido deberá ser el siguiente:

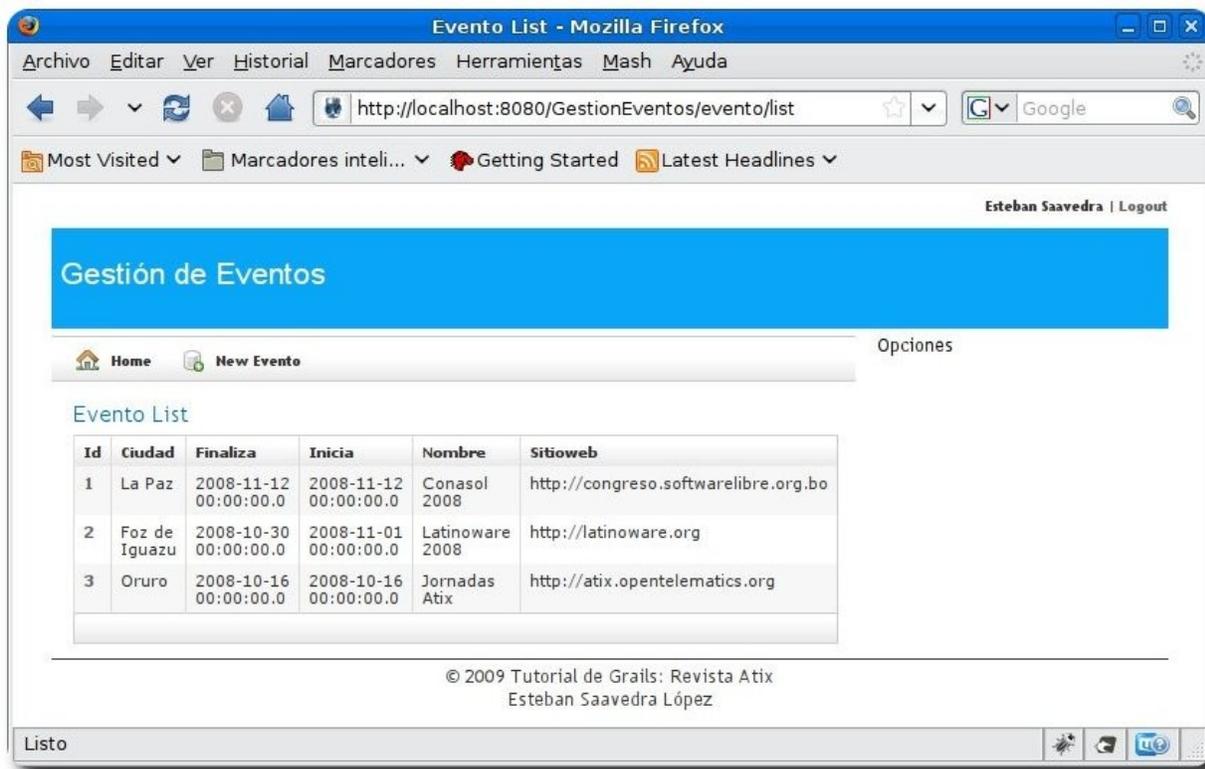
grails-app/views/common/_topbar.gsp

```

<div id="menu">
    <nobr>
        <g:if test="\${session.user}">
            <b>\${session.user?.nombre} &nbsp; \${session.user?.apellido}</b> |
            <g:link controller="login" action="logout">Logout</g:link>
        </g:if>
        <g:else>
            <g:link controller="login" action="login">Login</g:link>
        </g:else>
    </nobr>
</div>

```

El resultado del proceso de login se muestra en el siguiente gráfico:



Asegurar los controles

Una práctica común dentro del desarrollo de aplicaciones web, es asegurar los controles de acceso, referidos principalmente a controlar mediante un proceso de login (autenticación de usuarios) el acceso a distintas partes de la aplicación. Para esto Grails posee distintos mecanismos, algunos de ellos basados en plugins y otros basados en el uso de filtros.

Para nuestro caso haremos uso de filtros, que es el mecanismo más sencillo y rápido de implementar. Para esto debemos crear una clase filters, cuyo contenido es el siguiente:

grails-app/conf/SecurityFilters.groovy

```
class SecurityFilters {
    def filters = {
        GestionEventosFilter(controller: '*', action: '*') {
            before = {
                if(!session.user
                    && !controllerName.equals('login')
                    && !controllerName.equals('captcha')
                    && ( !controllerName.equals('user')
                    && !actionName.equals("registro"))
                ) {
                    redirect(controller:'login')
                    return false
                }
            }
        }
    }
}
```

El anterior código hará que los usuarios para acceder a cualquier parte de la aplicación necesariamente tengan que autenticarse, dicho de otra forma si el usuario trata de acceder a una URL de un módulo en específico de la aplicación, será redireccionado al módulo de login.

Referencias

- [1] <http://www.grails.org>
- [2] Getting Started with Grails
- [3] The Definitive Guide to Grails

Autor



Esteban Saavedra López

Líder de la Comunidad ATIX (Oruro – Bolivia)

Activista de Software Libre en Bolivia

jesaavedra@opentelematics.org

<http://jesaavedra.opentelematics.org>



Administrando GNU/Linux con Webmin

Esta herramienta basada en TECNOLOGÍA WEB nos facilita la administración local y/o remota de equipos con Sistema Operativo UNIX y Unix-like.



Introducción

Webmin es una interface basada en tecnología web para la administración sencilla de Sistemas Operativos Unix y Unix-like (Linux). Utilizando cualquier Web Browser usted puede realizar localmente o remotamente varias tareas administrativas, como ser:

- ✓ Administración de cuentas de usuarios y grupos.
- ✓ Administración de Apache Web Server.
- ✓ Administración de de DNS Server.
- ✓ Administración de MySQL Server.

- ✓ Administración de SAMBA.
- ✓ File Sharing.
- ✓ Cuotas de espacio.
- ✓ Administración de Servicios.
- ✓ Apagado y reinicio del equipo.
- ✓ Etc, etc, etc, etc.
- ✓

El Webadmin está escrito en el lenguaje Perl, incluye su propio servicio y su propio Web Server, es decir que no es necesario que esté corriendo una Web Server adicional como APACHE para su funcionamiento.

Si se tienen muchas máquinas que administrar es mucho más fácil hacerlo remotamente con esta herramienta (administración centralizada) que administrarlas localmente una por una.

Y en el caso de tener equipos con Sistema Operativo en Modo Texto, esta herramienta nos facilita aun más la administración.

Sistemas Operativos soportados

Actualmente la versión 1.480 del Webmin soporta varios Sistemas Operativos Unix y Unix-like:

AlphaCore Linux	Endian Firewall Linux	openmamba Linux	SuSE OpenExchange Linux
APLINUX	FreeBSD	OpenNA Linux	SuSE SLES Linux
Asianux	Generic Linux	Oracle Enterprise Linux	Tao Linux
Asianux Server	Gentoo Linux	Oracle VM	Tawie Server Linux
BigBlock	Gralinux	pclinuxos Linux	ThizLinux Desktop
BSDI	Haansoft Linux	Playstation Linux	ThizServer
Caixa Magica	HP/UX	Redhat Enterprise Linux	TinySofa Linux

Caldera OpenLinux	IBM AIX	Redhat Linux	Trustix
Caldera OpenLinux eServer	Immunix Linux	Redhat Linux Desktop	Trustix SE
cAos Linux	Lanthen Linux	SCI Linux	TurboLinux
Cendio LBS Linux	LinuxPPC	Scientific Linux	Ubuntu Linux
CentOS Linux	Lycoris Desktop/LX	SCO OpenServer	United Linux
Cobalt Linux	Mac OS X	SCO UnixWare	Ute Linux
Coherent Technology Linux	Mandrake Linux	Secure Linux	White Dwarf Linux
Conectiva Linux	Mandrake Linux Corporate Server	SGI Irix	Whitebox Linux
Corel Linux	Mandriva Linux	Slackware Linux	Windows
Corvus Latinux	Mepis Linux	Slamd64 Linux	X/OS Linux
Cygwin	MSC Linux	SoL Linux	Xandros Linux
Darwin	NeoShine Linux	StartCom Linux	Yellow Dog Linux
Debian Linux	NetBSD	Sun Java Desktop System	Yoper Linux
DEC/Compaq OSF/1	OpenBSD	Sun Solaris	
DragonFly BSD	OpenDarwin	SuSE Linux	

Instalación de Webmin

Primero se debe instalar las dependencias de Webmin, para lo cual ejecutaremos el siguiente comando en Debian:

```
# apt-get install libnet-ssleay-perl libauthen-pam-perl libio-pty-perl libmd5-perl
```

Podemos bajar los instaladores del Webadmin de <http://www.webmin.com/download.html>

Para la instalación de Webmin en Debian ejecutaremos el siguiente comando:

```
# dpkg -i webmin_1.480_all.deb
```

Configuración de Webmin

Si deseamos cambiar la configuración básica del Webmin, debemos cambiar el contenido del archivo de configuración de su Web Server, el MINISERV es su servidor web programado en Perl el cual es muy sencillo y básico. Este archivo de configuración es el **/etc/webmin/miniserv.conf**

```
port=10000
root=/usr/share/webmin
mimetypes=/usr/share/webmin/mime.types
addtype_cgi=internal/cgi
realm=Webmin Server
logfile=/var/webmin/miniserv.log
errorlog=/var/webmin/miniserv.error
pidfile=/var/webmin/miniserv.pid
logtime=168
ppath=
ssl=1
```

```
env WEBMIN_CONFIG=/etc/webmin
env WEBMIN_VAR=/var/webmin
atboot=1
logout=/etc/webmin/logout-flag
listen=10000
denyfile=\.pl$
log=1
blockhost_failures=5
blockhost_time=60
syslog=1
session=1
premodules=WebminCore
userfile=/etc/webmin/miniserv.users
keyfile=/etc/webmin/miniserv.pem
passwd_file=/etc/shadow
passwd_uindex=0
passwd_pindex=1
passwd_cindex=2
passwd_mindex=4
passwd_mode=0
preroot=blue-theme
passdelay=1
sudo=1
debian01:/etc/webmin#
```

Como se puede observar, en este archivo se puede configurar desde el puerto de escucha del Web Server, la ubicación de los archivos logs, si se utiliza https (ssl), etc.

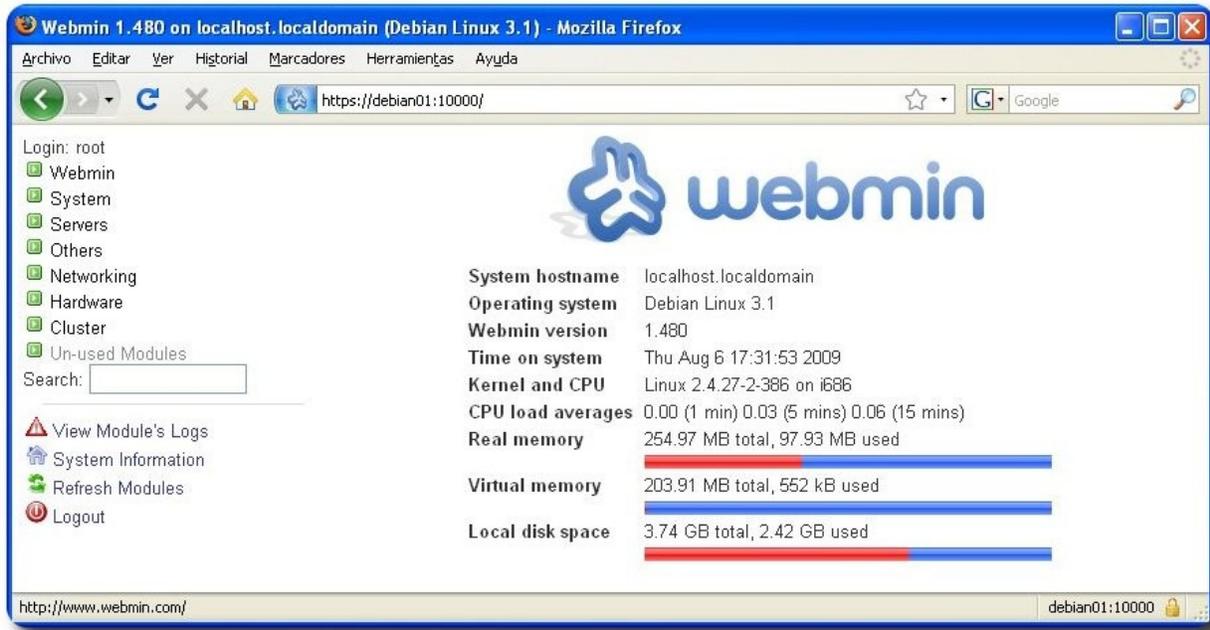
Fron-End del Webmin

Para ingresar a la página web del front-end del Webmin debemos ingresar a la siguiente URL: **https://servidor:10000/**

Donde se nos pedirá el usuario administrador:



Después de logearnos, se nos presentará la página principal del Webmin:



Donde se puede apreciar en la parte derecha el resumen del equipo que se está administrando y en la parte izquierda se muestra el menú donde se puede apreciar las siguientes módulos de administración:

Webmin: Desde este sub-menú se pueden realizar tareas de administración del servicio Webmin, como:

Backup Configuration Files	Webmin Configuration
Change Language and Theme	Webmin Servers Index
Webmin Actions Log	Webmin Users

System: Desde este sub-menú se pueden realizar tareas de administración del equipo, como:

Bootup and Shutdown	PAM Authentication
Change Passwords	Running Processes
Disk and Network Filesystems	Scheduled Commands
Filesystem Backup	Scheduled Cron Jobs
Initial System Bootup	Software Packages
Log File Rotation	System Documentation
MIME Type Programs	System Logs

Users and Groups

Servers: Desde este sub-menú se pueden realizar tareas de administración de servicios como:

Apache Webserver	Procmail Mail Filter
CVS Server	Read User Mail
PostgreSQL Database Server	SSH Server

Others: Desde este sub-menú se pueden realizar otras tareas de administración de equipo, como:

Command Shell	Protected Web Directories
Custom Commands	SSH Login
File Manager	System and Server Status
HTTP Tunnel	Upload and Download
Perl Modules	

Networking: Desde este sub-menú se pueden realizar tareas de administración del servicio de red, como:

Bandwidth Monitoring	NIS Client and Server
Internet Services and Protocols	Network Configuration
Linux Firewall	TCP Wrappers

Hardware: Desde este sub-menú se pueden realizar tareas de administración de:

CD Burner	Partitions on Local Disks
GRUB Boot Loader	Printer Administration
Linux RAID	System Time

Cluster: Desde este sub-menú se pueden realizar tareas de administración del cluster de linux:

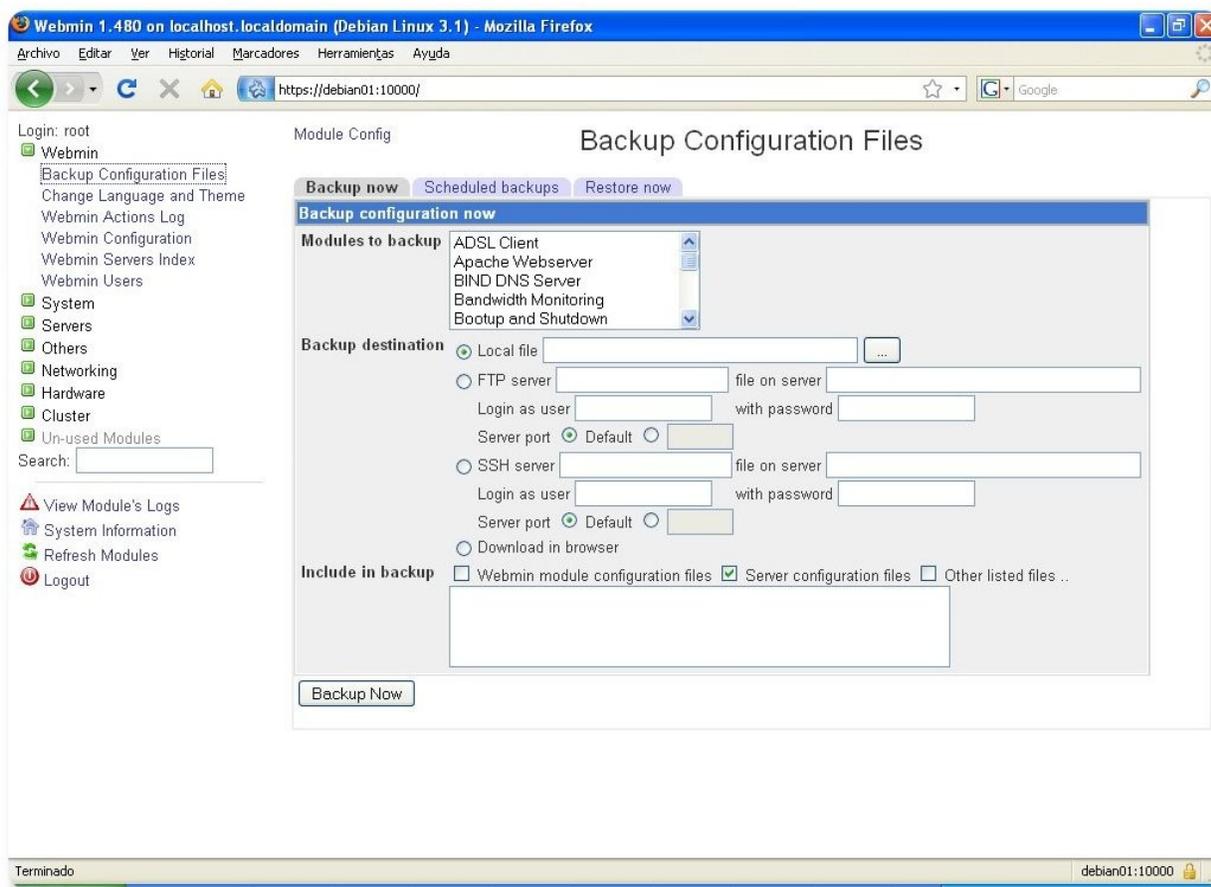
Cluster Change Passwords	Cluster Software Packages
Cluster Copy Files	Cluster Usermin Servers
Cluster Cron Jobs	Cluster Users and Groups
Cluster Shell Commands	Cluster Webmin Servers

Un-used Modules: Desde este sub-menú se pueden ver los servicios que no se están utilizando al momento (deshabilitados ó desinstalados) pero que si podrían ser administrados por el Webmin.

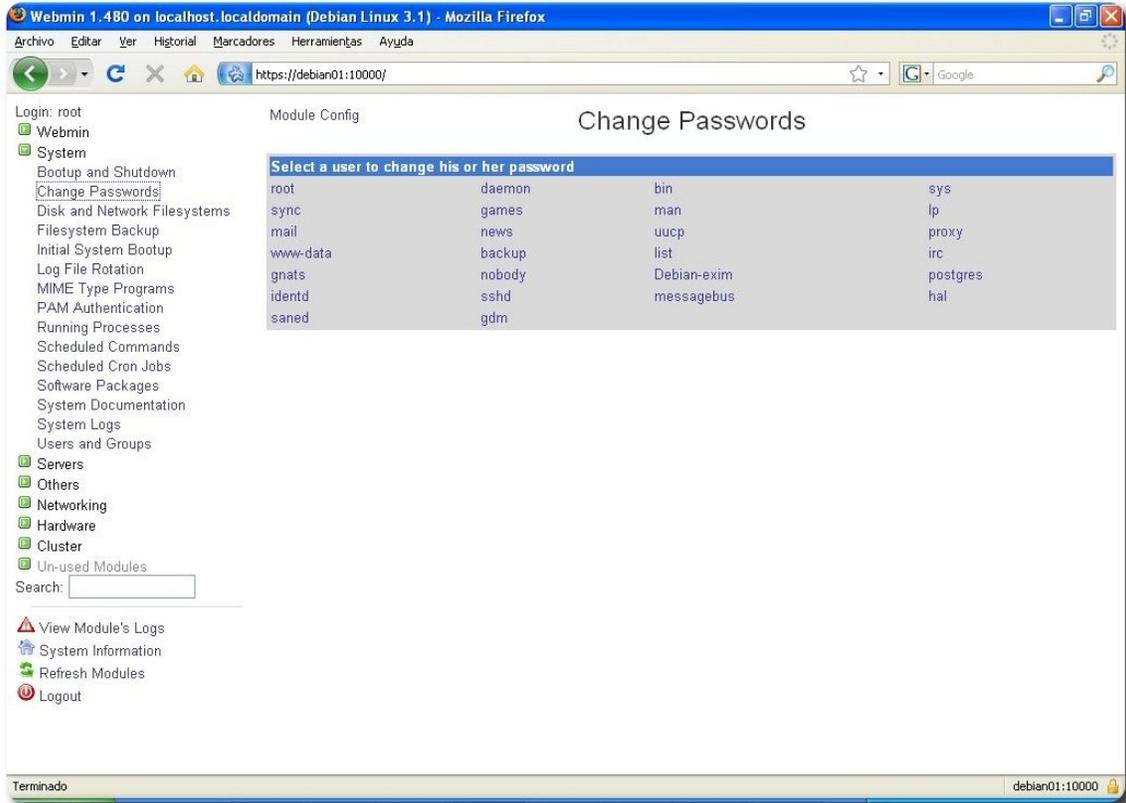
Como se puede observar, desde el Webmin se puede administrar casi todo el Sistema Operativo, cabe recalcar que muchas nuevas aplicaciones ya vienen integradas al Webmin para su administración a través de esta simple y útil herramienta.

A continuación mostraremos ejemplos de algunos módulos de administración del Webadmin:

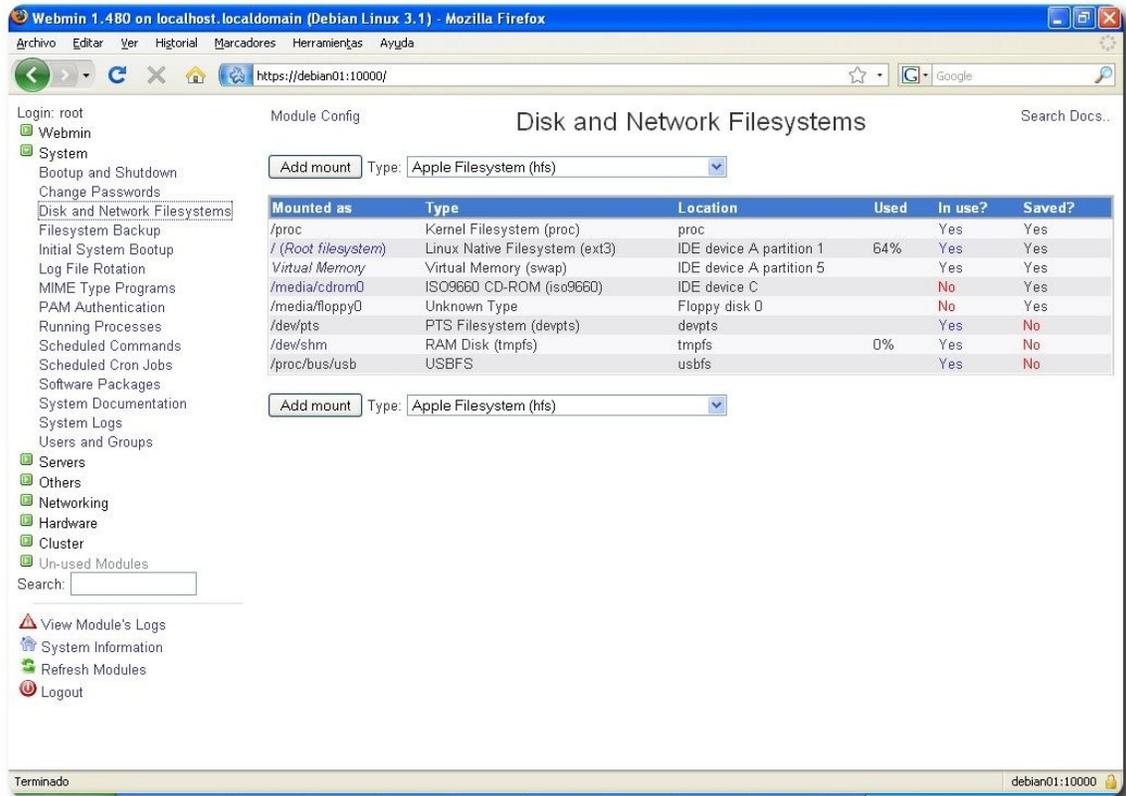
Backup y restore de archivos de configuración:



Cambio de claves de usuarios



Discos y nfs



Monitoreo de procesos

The screenshot shows the 'Running Processes' page in Webmin. It displays a table with columns for ID, Owner, Started, and Command. The table lists various system processes such as init, klogd, and sshd.

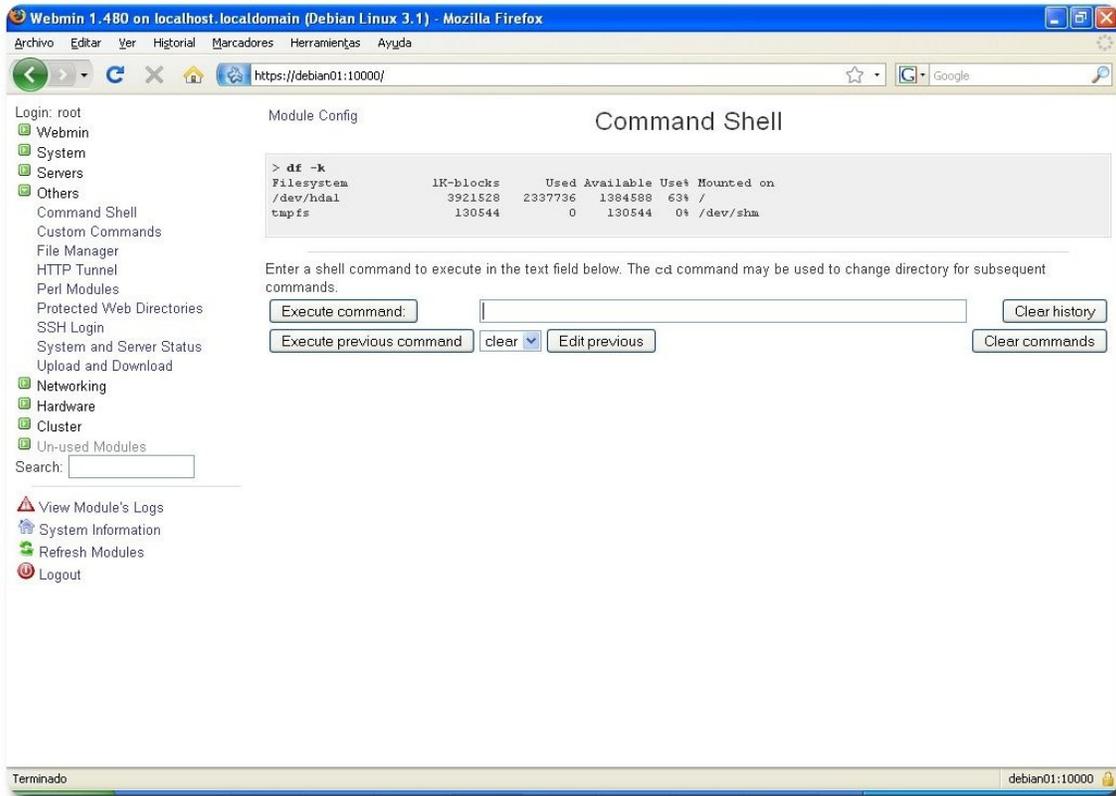
ID	Owner	Started	Command
1	root	16:57	init [2]
2	root	16:57	[klogd]
3	root	16:57	[ksoftirqd_CPU0]
4	root	16:57	[kswapd]
5	root	16:57	[bdflush]
6	root	16:57	[kupdated]
99	root	16:57	[kjournald]
229	root	16:58	[mdrecoveryd]
362	root	16:58	udev
477	root	16:58	[khubb]
911	daemon	16:58	/sbin/portmap
1076	root	16:58	/sbin/syslogd
1079	root	16:58	/sbin/klogd
1089	messagebus	16:58	/usr/bin/dbus-daemon-1 --system
1095	root	16:58	/usr/bin/dirmngr --daemon --sh
1224	Debian-exim	16:58	/usr/sbin/exim4 -bd -q30m
1230	root	16:58	/usr/sbin/inetd
1234	lp	16:58	/usr/sbin/lpd -s
1293	postgres	16:58	/usr/lib/postgresql/bin/postmaster -D /var/lib/postgresql/data
1298	postgres	16:58	postgres: stats buffer process
1299	postgres	16:58	postgres: stats collector process
1304	postgres	16:58	/usr/lib/postgresql/bin/pg_autovacuum -D -p 5432 -L /var/log/postgresql/autovac...
1310	root	16:58	/usr/sbin/snmpd -Lsd -Lf /dev/null -p /var/run/snmpd.pid
1315	root	16:58	/usr/sbin/sshd
1796	root	17:04	sshd: root@pts/0
1798	root	17:04	sshd: root@pts/0
1800	root	17:04	-bash
1320	root	16:58	/usr/sbin/famd -T 0
1324	root	16:58	/sbin/rpc.statd

Administración de usuarios y grupos

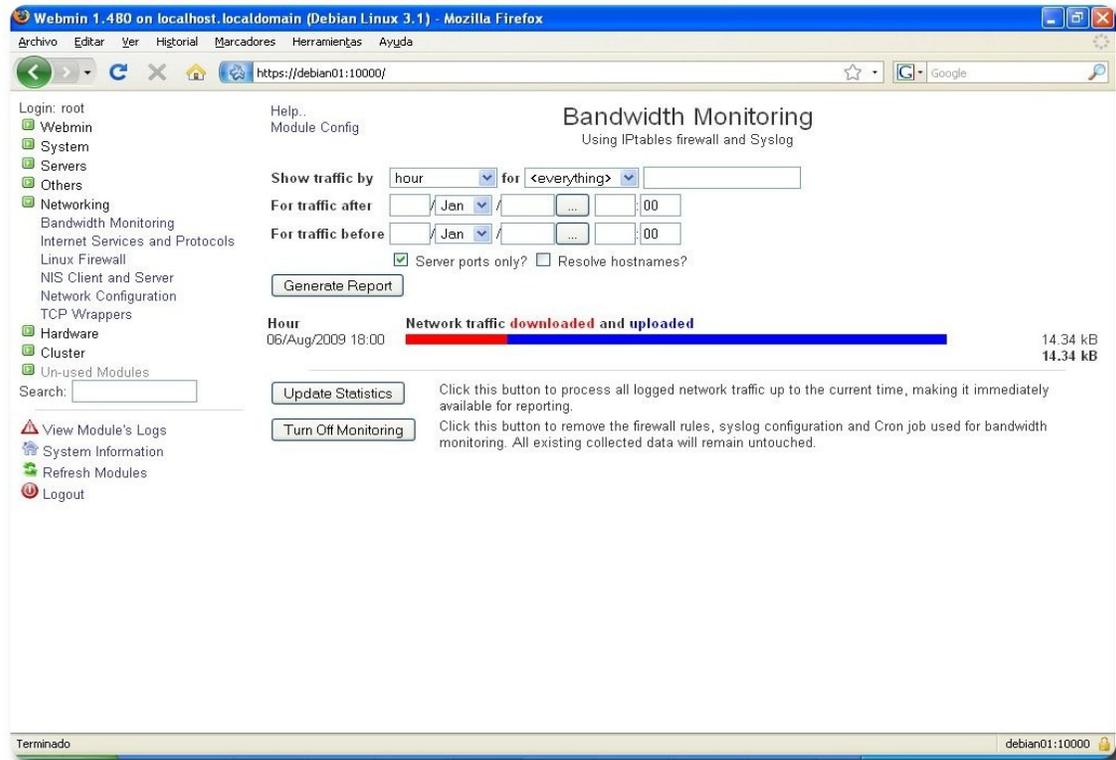
The screenshot shows the 'Users and Groups' page in Webmin. It displays a table of local users with columns for Username, User ID, Group, Real name, Home directory, and Shell. The table lists users such as root, daemon, bin, sys, and postgres.

Username	User ID	Group	Real name	Home directory	Shell
<input type="checkbox"/> root	0	root	root	/root	/bin/bash
<input type="checkbox"/> daemon	1	daemon	daemon	/usr/sbin	/bin/sh
<input type="checkbox"/> bin	2	bin	bin	/bin	/bin/sh
<input type="checkbox"/> sys	3	sys	sys	/dev	/bin/sh
<input type="checkbox"/> sync	4	nogroup	sync	/bin	/bin/sync
<input type="checkbox"/> games	5	games	games	/usr/games	/bin/sh
<input type="checkbox"/> man	6	man	man	/var/cache/man	/bin/sh
<input type="checkbox"/> lp	7	lp	lp	/var/spool/lpd	/bin/sh
<input type="checkbox"/> mail	8	mail	mail	/var/mail	/bin/sh
<input type="checkbox"/> news	9	news	news	/var/spool/news	/bin/sh
<input type="checkbox"/> uucp	10	uucp	uucp	/var/spool/uucp	/bin/sh
<input type="checkbox"/> proxy	13	proxy	proxy	/bin	/bin/sh
<input type="checkbox"/> www-data	33	www-data	www-data	/var/www	/bin/sh
<input type="checkbox"/> backup	34	backup	backup	/var/backups	/bin/sh
<input type="checkbox"/> list	38	list	Mailing List Manager	/var/list	/bin/sh
<input type="checkbox"/> irc	39	irc	ircd	/var/run/ircd	/bin/sh
<input type="checkbox"/> gnats	41	gnats	Gnats Bug-Reporting System (admin)	/var/lib/gnats	/bin/sh
<input type="checkbox"/> nobody	65534	nogroup	nobody	/nonexistent	/bin/sh
<input type="checkbox"/> Debian-exim	102	Debian-exim		/var/spool/exim4	/bin/false
<input type="checkbox"/> postgres	100	postgres	PostgreSQL administrator	/var/lib/postgres	/bin/bash
<input type="checkbox"/> identd	101	nogroup		/var/run/identd	/bin/false
<input type="checkbox"/> sshd	103	nogroup		/var/run/sshd	/bin/false
<input type="checkbox"/> messagebus	104	messagebus		/var/run/dbus	/bin/false
<input type="checkbox"/> hal	107	hal	Hardware abstraction layer	/var/run/hal	/bin/false

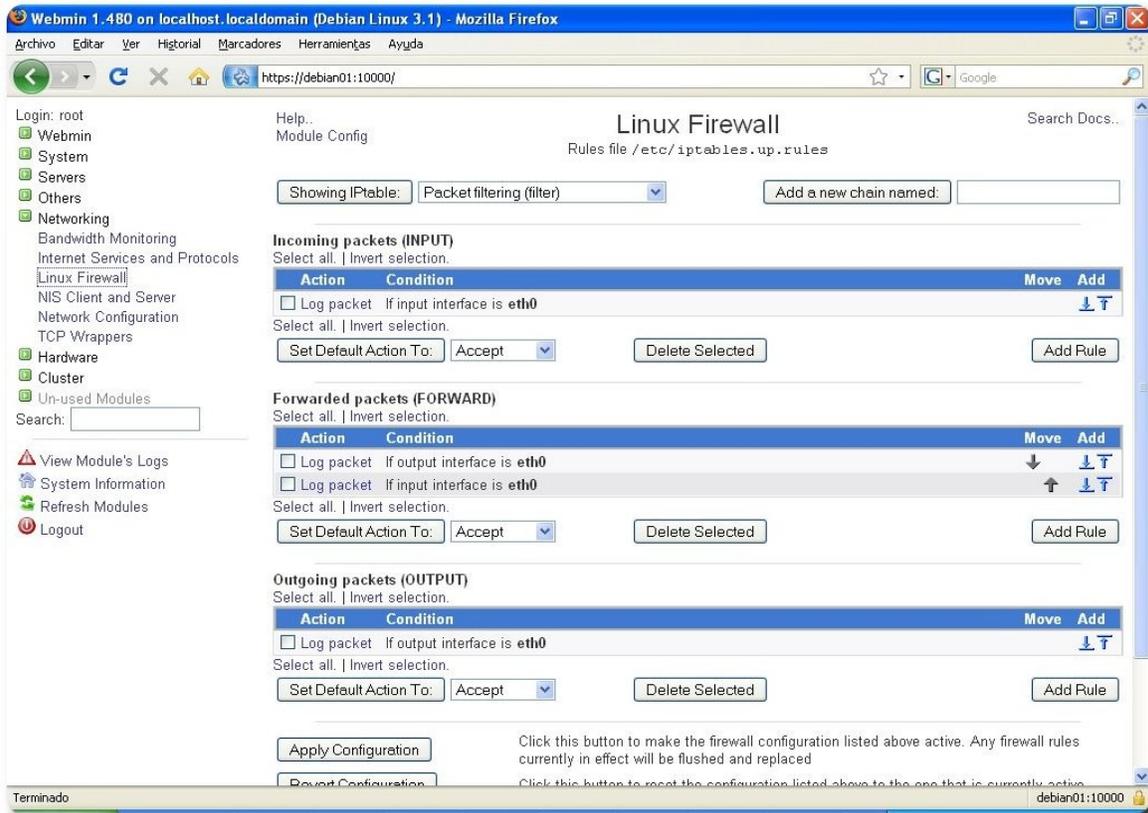
Ejecución de la línea de comando



Monitoreo de red



Firewall



Conclusiones

Este tipo de herramienta basada en Tecnología WEB nos facilita bastante las tareas administrativas del Sistema Operativo, al poder realizarlas desde un solo Front-End.

Cuando debemos administrar muchos equipos, esta herramienta nos permite centralizar la administración.

Referencias

<http://www.webmin.com>

<http://doxfer.com/Webmin/Modules>

Autor



Renzo Martinez Pardo

Ingeniero Sistemas Electrónicos
renzomp@lycos.com



OFFSFIO

Fundación Software Libre
América Latina



Primer Encuentro
Fundación Software Libre
América Latina



En el marco del quinto congreso nacional de software libre en Venezuela, se llevo a cabo el “**Primer encuentro de la Fundación Software Libre America Latina**”.

Justificación

En vista de la cantidad de proyectos, propuestas, acciones y eventos en los cuales están activamente participando los Miembros de la Fundación del Software Libre para la América Latina (FSFLA), se plantea la necesidad de un encuentro de sus principales activistas y líderes para complementar el trabajo de comunicación a través de los medios electrónicos.

Objetivos:

Reunir a los líderes del movimiento del Software Libre de Latinoamérica en un evento donde se logren las siguientes metas:

- ✓ Lograr la interacción personal entre los miembros del Consejo para estrechar lazos de amistad y fraternidad en función del mejoramiento de las relaciones humanas de la organización
- ✓ Comprender la realidad de cada uno de los países representados por los miembros del consejo para así tener una visión mucho mas amplia del universo que nos rodea, generando un mapa geopolítico y situacional del Movimiento del Software Libre en la América Latina
- ✓ Proponer estrategias de mediano y largo plazo en miras a fortalecer coherentemente y mancomunadamente las acciones que cada uno de los líderes de este movimiento permanentemente llevan a cabo en cada una de las regiones de América.

Actividades

Son varias las actividades que realizaron los miembros del consejo de la FSFLA, entre las que destacan:

- ✓ Visita a distintos organismos nacionales (Venezuela) e internacionales con el objetivo de hacer conocer los beneficios que representa adopción del software libre en contextos sociales y académicos.
- ✓ Ruedas de prensa dando a conocer la situación del software libre en los países de América Latina.
- ✓ Brindar conferencias sobre las experiencias técnicas y sociales entorno a la difusión de los valores éticos y morales promovidos por la FSF y FSFLA
- ✓ Establecer lazos de colaboración entre la FSFLA, comunidades de usuarios GNU/Linux, y otras instituciones a nivel latinoamericano, de tal forma de contribuir al desarrollo de una sociedad mas justa y libre por medio del uso del software libre y los valores éticos y morales que promueve este.



Miembros del Consejo de la Fundación Software Libre América Latina (Esteban Saavedra, Daniel Yucra, Andres Casteblanco, Quiliro Ordoñez, Octavio Rosell, Oscar Valenzuela, Alexandre Oliva)



Miembros del Consejo junto a Richard Stallman y Carlos Figueira (Presidente del CNTI)



FSFLA frente a la Casa de Simón Bolívar



Almuerzo con los organizadores



Compartiendo con los organizadores



FSFLA junto a RMS en una rueda de prensa



FSFLA presente en el Parlamento Latinoamericano



FSFLA en rueda de prensa



FSFLA en rueda de prensa



Compartiendo con los organizadores



Exposición de Daniel Yucra



Exposición de Quiliro Ordoñez



Exposición de Oscar Valenzuela



Exposición de Esteban Saavedra



Exposición de Andres Castebianco



FSFLA recibiendo la Lemote Yeeloong por parte de Gustavo Fonseca (Presidente de Poder Digital)



El ultimo almuerzo de confraternidad FSFLA - Glove

Declaración de Caracas

Necesidad de cooperación internacional y comunitaria en Latinoamérica en favor del Software Libre

Caracas, Venezuela. 20 de julio del 2009

Primer Encuentro Fundación Software Libre América Latina

Preámbulo

En Caracas, Venezuela a los 20 días del mes de Julio del 2009, de conformidad con los valores de Libertad consagrados en el manifiesto GNU y en la definición de Software Libre,

Considerando que el conocimiento científico y tecnológico constituye una necesidad y un derecho de los Pueblos de América Latina, como política prioritaria para el desarrollo cultural, económico, social y político de sus naciones.

Considerando que es irrenunciable el compromiso de defender los derechos de los usuarios, desarrolladores, gobiernos y empresas para usar, adaptar, compartir y mejorar su software y resistir al manejo de información personal de forma no autorizada por parte de terceros, para así poder tener control de su informática.

Considerando que el Software Libre es un modelo ético de desarrollo tecnológico y de carácter colaborativo, basado o soportado en un tejido social conformado por equipos multidisciplinarios que luchan y participan por un fin común: la Libertad de Software y los valores que la misma conlleva.

Considerando que el Software Libre representa, para los Pueblos y Gobiernos de América Latina, una oportunidad para la adopción de Estándares Abiertos Libres en sus procesos administrativos, que se ajusten a sus necesidades de implementación de sistemas de información para el Gobierno Electrónico.

Considerando que la adopción del Software Libre desarrollado con Estándares Abiertos Libres en los gobiernos de América Latina facilitará la interoperabilidad de los sistemas de información de los Estados, contribuyendo a dar respuestas rápidas y oportunas a los ciudadanos, mejorando la gobernabilidad, a la par de una mayor participación de los usuarios en el mantenimiento de los niveles de seguridad de su software.

Considerando que el Software Libre representa una oportunidad única para la consolidación de la Soberanía e Integración Tecnológica de los pueblos de América Latina, y la eliminación del bloqueo tecnológico causado por monopolios de Software Privativo.

Considerando que una concepción común de estos derechos y libertades es de la mayor importancia para el pleno cumplimiento de dicho compromiso.

Como Fundación Software Libre América Latina, hemos resuelto publicar el siguiente mensaje a través de este documento, el cual proclama a la "Libertad de Software" como un ideal común, por el cual todas las naciones de América Latina deben esforzarse, con el fin de generar un trabajo comunitario que promueva y exija valores éticos, mediante la enseñanza y el respeto a los derechos y Libertades de usar, estudiar, modificar y distribuir Software Libre. Así es como hemos desarrollado la siguiente declaración titulada "Declaración de Caracas" que incluye recomendaciones para cada uno de los ejes de acción que consideramos prioritarios.

Sobre las Comunidades Locales y el Software Libre

Invitamos a las comunidades Latinoamericanas y sus integrantes a difundir todas sus actividades y por sobre todo sus casos de éxito, puesto que el conocimiento de todos los logros locales a nivel internacional servirá para ejemplificar con hechos el beneficio de la Libertad, incentivando que otras comunidades lo imiten.

Así también, para el éxito de nuestra misión es importante dejar de lado las diferencias y problemas que han sido ya antecedentes históricos, tomando iniciativas de trabajo en comunidad, de modo que proliferen los muchos símiles por encima de las pocas diferencias de las comunidades locales, para lograr más y mejores resultados.

Los activistas para la Libertad de Software tienen la responsabilidad de entregar valores, defendiendo y difundiendo las Libertades esenciales que definen el Software Libre, y es en este sentido que solicitamos dar a conocer a los usuarios sobre el daño que genera el Software Privativo que incluyen la mayoría de las distribuciones de GNU/Linux populares actualmente, e invitamos a promover el uso de distribuciones totalmente Libres, educando a la sociedad para la Libertad y sus valores por encima de la tecnología.

La Fundación Software Libre América Latina, consciente de las necesidades y requerimientos para enfrentar las graves implicaciones que el Software Privativo impone, renueva su compromiso para apoyar la difusión y los procesos comunitarios que generen sinergia entre las comunidades locales a nivel internacional, en favor de la Libertad y los valores que el Software Libre promueve.

Sobre Software Libre y los Estados Latinoamericanos

Los gobiernos deben representar y promover los intereses de su pueblo, siendo su deber el asegurar el control de los bienes que administran y regulan, razón por la cual deben mantenerlos en su dominio a través de Software que traiga consigo la libertad de ejecutarlo para cualquier propósito, de estudiar su código para conocer su funcionamiento y adaptarlo a sus necesidades, garantizando así la soberanía del estado en el ámbito tecnológico y la continuidad e integridad del acceso a la información.

Es por esto que invitamos a los gobiernos a utilizar y promover Software Libre (incluyendo controladores libres y tecnologías asociadas) para que puedan cumplir con su deber de mantener el control propio, la auditabilidad y la soberanía.

Todo lo que los estados producen en cuanto a Software es de propiedad de los ciudadanos y por ende, un bien público, que debe estar a disposición del pueblo, respetando las Libertades esenciales del Software Libre. Además, estos bienes públicos deben mantener su función de servir a los ciudadanos y deben ser publicados bajo términos que promuevan los intereses de las naciones y la sociedad. Hacemos un llamado a los gobiernos para que publiquen el software que desarrollen y usen, bajo licencias que no solo respeten, sino también defiendan y promuevan los valores apropiados para todos sus usuarios, es decir, licencias de Software Libre y Izquierdo de Autor (Copyleft), que hacen que las libertades sean inseparables del software.

Gobiernos de América Latina: promuevan una cultura de respeto a la Libertad de Software, rompiendo la inercia social que induce a los gobiernos y personas a ceder sus libertades, permitiéndoles generar una sociedad más libre, equitativa y justa.

Software Libre en la Educación Latinoamericana

En el ámbito de la educación, con la enseñanza del Software Libre se inculcarán los valores éticos y morales como instrumento dinámico de integración entre los individuos, sus contextos sociales y por ende en todas las naciones.

Hacemos un llamado a promover valores en los estudiantes para con su sociedad, fomentando en ellos la cooperación y el ánimo para compartir con su prójimo a través del uso de Software Libre, ya que el uso del Software Privativo convierte el compartir y colaborar en un delito, y coarta la Libertad de aprender al no permitir el acceso al conocimiento sobre como el Software está construido.

Otro punto, es que el Software Libre permite un mejor aprovechamiento y redistribución de los recursos económicos y estos ahorros permiten mejores plataformas educativas en los centros de formación.

Nuestro Compromiso

La Fundación Software Libre América Latina, actuando como una red internacional de organizaciones y personas que promueven la Libertad de Software, servirá como un agente facilitador para las comunicaciones y la difusión de las actividades locales, junto a apoyar las iniciativas en forma de representación y puente internacional.

Acerca de este documento

El presente documento nace producto de la primera reunión de miembros de la Fundación Software Libre América Latina, quienes en Caracas, Venezuela, en el marco del Quinto Congreso Nacional de Software Libre se reúnen y resuelven generar ésta declaración. Esta contiene un conjunto de impresiones y posiciones sobre los aspectos comunitarios, educativos y políticos, donde el enfoque principal que se le ha dado al documento es promover los valores de la Libertad por encima de la tecnología y los valores éticos por encima de los prácticos.

Acerca de FSFLA

Fundación Software Libre América Latina se ha sumado desde el 2005 a la red internacional de FSFLAs, anteriormente formada por las Free Software Foundations de los Estados Unidos, de Europa y de la India. Estas organizaciones hermanas actúan en sus respectivas áreas geográficas con el sentido de promover los mismos ideales de Software Libre y defender las mismas Libertades para usuarios y desarrolladores de software, trabajando localmente, pero cooperando globalmente.

Copyright 2009 FSFLA

Se permite la distribución y la copia literal de este artículo en su totalidad por cualquier medio en todo el mundo sin paga de derechos, siempre y cuando se conserve la nota de copyright, el URL oficial del artículo y esta nota de permiso.

<http://www.fsfla.org/anuncio/2009-07-declaración-de-caracas>

Firman en unidad de acto los consejeros de la Fundación Software Libre América Latina, en la ciudad de Caracas, Venezuela el 20 de Julio del 2009, con la cooperación del observador del consejo Eduardo Saavedra.

Richard Stallman

Bolivia



Software libre & descolonización

Sábado 15 de agosto, 10h30:
Taller "El Peligro de las Patentes del Software"
 Vicepresidencia del Estado Plurinacional
 Calle Ayacucho y Mercado, La Paz

Lunes 17 de agosto 18:00:
"Software Libre en la Ética y en la Práctica"
El Movimiento del Software Libre y el Sistema Operativo GNU/Linux"
 Paraninfo de la UMSA
 Avenida Villazón, La Paz

Martes 18 de agosto, 19h:
"Copyright vs Comunidad en la Era de las Redes Informáticas"
 Auditorio del Banco Central de Bolivia
 Calle Ayacucho y Mercado, La Paz

por **Richard M. Stallman**
 fundador del movimiento de Software Libre

www.softwarelibre.org.bo
 contacto@softwarelibre.org.bo

Logos: GNU/Linux, UMSA, Ministerio de Planificación del Desarrollo, ADSIB, www.cienciaytecnologia.gob.bo

La **Comunidad Nacional de Software Libre (www.softwarelibre.org.bo)**, organizó la llegada de **Richard Stallman** a Bolivia, oportunidad donde se realizaron varias actividades en las cuales participó el fundador del movimiento del Software Libre en el mundo y presidente de la Free Software Foundation (**www.fsf.org**)

Charlas preparativas: lunes 10 a jueves 13 de agosto

Lunes 10 de Agosto Lugar: Biblioteca Municipal

- ✓ **Una Computadora por Niñ@ herramientas para una educación descolonizada,** Daniel Viñar Ulriksen - 19h30

Jueves 13 de Agosto Lugar: Vicepresidencia del Estado Plurinacional de Bolivia

- ✓ **Licencias de Software Libre,** Hardy Beltran Monasterios - 16:00
- ✓ **Presentación "Notebook Lemote HW 100% Libre",** Esteban Saavedra 17:00
- ✓ **Aprendizaje Tecnológico e Innovación, Experiencias de ProcessMaker,** Patricia Cabero - 18:00
- ✓ **Software Libre y Medio Ambiente,** Amos Batto - 19:00

Entrevistas y Visitas

Adicionalmente a sus conferencias magistrales, Richard Stallman, fue entrevistado por algunos medios de comunicación como: Erbol (entrevista con Amalia Pando), Red Uno (Programa: Que no me pierda), sin olvidar a muchos otros que cubrieron sus conferencias y presentaciones.

Richard Stallman tuvo la oportunidad de reunirse con miembros de la comunidad nacional de software libre y miembros de proyectos de software libre en Bolivia.

Gracias Richard por visitar **BOLIVIA** y compartir con nosotros....

El peligro de las Patentes de Software

Hall de la Vicepresidencia





Palabras de agradecimiento por parte de Alejandro Patiño (Director de la Agencia para el desarrollo de la sociedad de la información en Bolivia)



Cena con algunos de los miembros de la Comunidad Nacional de Software Libre

Software Libre en la Ética y en la Práctica

Paraninfo de la Universidad Mayor de San Andrés



Visita de Richard Stallman en BOLIVIA

RMS-Bolivia

Copyright vs Comunidad en la Era de las Redes Informáticas

Auditorio Banco Central de Bolivia



Miembros de la Comunidad de Software Libre junto a Richard Stallman e invitados especiales



Visita de Richard Stallman en BOLIVIA

RMS-Bolivia

Copyright vs Comunidad en la Era de las Redes Informáticas

Auditorio Banco Central de Bolivia



Visita de Richard Stallman en BOLIVIA

RMS-Bolivia

A large green circle containing the word "Willay" in white, bold, sans-serif font.

Willay

news

Información Actual

Willay news

Uso del software libre en la educación universitaria



Una vez más el uso del software libre específicamente orientado al aprendizaje de la programación, es un ejemplo de éxito la Universidad Anahuac de México.

Cada año se incrementa a nivel mundial las universidades que van apostando por el uso del software libre en el proceso enseñanza aprendizaje, empezando por la migración al Sistema Operativo GNU/Linux y utilizando algunas aplicaciones libres en las distintas asignaturas conforme a la malla curricular especialmente en las carreras de Ingeniería Informática, Sistemas y Telecomunicaciones.

Una de las materias básicas y elementales en un primer curso en Ingeniería es Programación, el uso del software libre es una buena alternativa y a la vez económica en el proceso enseñanza aprendizaje. El Ing. Virgilio De la Cruz Jardón docente de la Universidad Anahuac de México (<http://ingenieria.anahuac.mx>), nos habla sobre su experiencia de éxito: “Desde mediados del año 2007, la Facultad de Ingeniería de la Universidad Anahuac México Norte ubicada en la Ciudad de México, comenzó a usar el entorno de desarrollo GAMBAS (<http://gambas.sourceforge.net/>) para la enseñanza de la programación a los alumnos de primer semestre.

Gambas

<http://gambas.sourceforge.net>

Paquetes generados por el proyecto gnuLinEx (<http://www.linex.org>)

Este programa está publicado bajo la GNU General Public Licence



Gambas (Gambas Almost Means Basic) es lenguaje orientado a objetos derivado de BASIC, incluye un entorno de desarrollo sencillo de usar, el cual corre principalmente en GNU/ Linux. Este lenguaje tiene diferentes ventajas, entre las cuales destacan su sintaxis sencilla, que permite a los alumnos concentrarse en la lógica de programación, y la capacidad de desarrollar aplicaciones tanto de línea de comando como de ambiente gráfico. Los alumnos tienen diferentes formas de realizar la practicas en GAMBAS: pueden conectarse a un servidor con sistema operativo GNU/Linux dentro del campus, usar las computadoras que tienen instalado localmente GNU/Linux en el laboratorio o usar UANLinux en su casa el cual consiste en una versión modificada de Knoppix con este entorno de desarrollo. Al final del curso los alumnos entregan un proyecto final aprovechando los elementos gráficos de este lenguaje, que consiste en un juego de mesa totalmente funcional, ya sea “Serpientes y Escaleras”, un juego de Memoria o un Dominó entre otros”.

El uso del software libre en las Universidades es una buena alternativa que ayuda a mejorar la calidad de enseñanza y contribuye la formación de profesionales de mayor calidad.

Posible amenaza para Google, Buscador Chino supera a Yahoo y Bing

En Asia el Buscador chino Baidu especializado en la búsqueda de música y vídeos supera a Yahoo y Bing.



Pensabas que Google tenía un dominio total a nivel mundial?, pues no. El gigante Google domina en todos los países del mundo excepto en China, uno de los países con mayor población a nivel mundial. Baidu creció un 2.6% en el segundo trimestre de este año, frente a Google que baja 1.5% en comparación con el primer trimestre. Pero conozcamos un poco de la historia del "Google Chino", Baidu fue fundado en Beijing en el 2000, cuyo nombre proviene de un poema de la dinastía Song de 800 años de antigüedad y significa "cientos de veces". La misión de Baidu es proporcionar a las personas las mejores alternativas para encontrar información, escuchando detenidamente las preguntas de sus usuarios sobre todo lo que necesitan y desean, algunas de las preguntas comunes son:

¿Coleccionan todas las páginas web chinas que los usuarios desean ver?, ¿Las páginas se actualizan cada día?, ¿Los resultados de las búsquedas se relacionan con las consultas realizadas?, ¿Devuelven los resultados de las búsquedas inmediatamente?, la gente de Baidu toman las experiencias de los usuarios para mejorar sus productos y servicios.

Baidu podría ser una amenaza para Google quizás a largo plazo, por el momento el Buscador está presente en la red en el idioma Chino, frente a Google que tiene versiones en cientos de idiomas, pero ello no deja de ser una amenaza, por lo que Google intenta ingresar al mercado Chino y aceptar algunas condiciones de su Gobierno. Puedes probar Baidu ingresando a la página www.baidu.com.

Autor



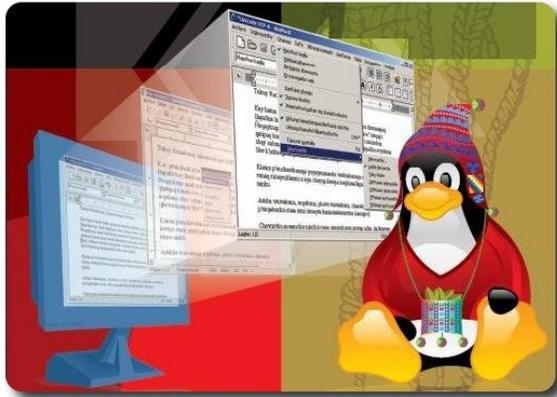
Ivonne Karina Menacho Mollo

Titulada de la carrera de Ingeniería de Sistemas e Informática (F.N.I.)

Conductora programa radial "Doble Clic"
ivonnekarina2003@hotmail.com

Willay news

Runasimipi.org un desafiante proyecto



Bien sabemos que las lenguas originarias se consideran del pasado, del campo, sólo de los abuelos, y hasta muchas veces las consideramos inútiles sin futuro cuando se trata de algo moderno o tecnológico. Promover una alternativa tecnológica que apoye culturas y lenguas originarias como el Quechua Aymara o Guaraní es el objetivo de Runasimipi.org.

Runasimipi.org pretende cambiar esta percepción, adaptando la manera de usar la computadora en cuanto a software, de modo que se empiece a valorar las lenguas originarias.

Por que hacer el cambio en una computadora? Según ellos: "Porque, es configurable, el usuario define como quiere ver los diferentes programas en su computadora y transformar el contenido de la misma y el Software libre permite todo esto".

Con una gran experiencia al traducir AbiWord (un excelente procesador de texto) al Quechua, en la cual se trabajo arduamente, buscando términos adecuados, con los que el usuario se pueda familiarizar, por ejemplo:

carpeta -> q'ipi (bulto, un objeto para llevar y guardar las cosas), celda -> uyu (corral, un lugar donde son guardados los animales).

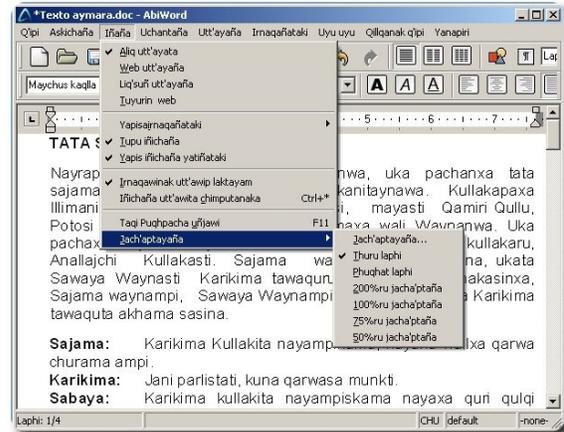


tabla -> uyu uyu (varios corrales, similar a una tabla física)

cuyo resultado fue: esta aplicación de forma bilingüe, sus futuros proyectos son bastante ambiciosos:

- ✓ Crear vocabulario nuevo con 5000 palabras en Aymara
- ✓ Crear tres correctores ortográficos
- ✓ Crear un glosario de informática
- ✓ Traducir al Quechua, Aymara y Guaraní
- ✓ OpenOffice (conjunto de ofimática)
- ✓ Mozilla Firefox (navegador Web)
- ✓ GCompros (conjunto educativo para niños)
- ✓ Tux Paint (diseñador gráfico para niños)

Esperando que después de la culminación de estos proyectos las lenguas originarias no sean tan absorbidas por las lenguas dominantes, no nos queda mas que, elogiar los desafíos tanto técnicos como de adecuación de términos, de esta organización cuyo sueño es el de que cualquier niño andino que vaya a una cabina pública de internet, en el futuro tenga la opción de ver todo en su lengua materna.

Google Voice pudo ser revolucionario en el AppStore



AppStore el nuevo iPhone de Apple no contará, con la aplicación que tenía el objetivo de unificar las comunicaciones de voz, denominada Google Voice (GV), la nueva versión de GrandCentral.

Las características de esta aplicación son brillantes permitiendo al usuario elegir su nuevo número de teléfono en Google Voice y cuando otros le llamen, Google puede llamar a todos los teléfonos reales de dicho usuario (puede unificar los teléfonos bajo un número exclusivo) y dejarle un mensaje de voz, de hecho con la tecnología "speech-to-text" aquellos mensajes que quedan en nuestro buzón de voz pasan a texto para ser leídos como si fuera un correo electrónico, o si el usuario lo desea puede escuchar directamente el mensaje. Los contactos de Gmail le indican a GV como identificar a diferentes interlocutores.

Quizá las siguientes características fueron el problema para la fatal expulsión de GV: esta aplicación también puede enviar mensajes de texto SMS y crear conferencias telefónicas, es decir mensajes SMS gratuitos y llamadas de larga distancia muy baratas. Justo en este momento todos pensamos: "genial!!! esto tendrá un impacto positivo en la vida de los usuarios", pero uno de los gigantes en telecomunicaciones como AT&T, no piensa lo mismo, y tal vez evitará que Google Voice se popularice, y ahora quizá pensemos "será que Apple reprime la innovación para dar gusto a uno de sus socios."



Pero sabemos bien que este pequeño inconveniente no cortará las grandes ideas de Google, con este y otros proyectos como Latitude, la gran empresa de Google nos demuestra que esta apostando fuertemente sobre las comunicaciones móviles de hecho ya esta trabajando en una versión de dicha aplicación, a la cual podremos acceder desde el navegador de la propia iPhone.

Autor



Marcia Estrella Velasquez Aguilar

Egresada de la carrera de Ingeniería de Sistemas (F.N.I.)
 mevaguerradelaluz@gmail.com

Willay news

Caffeine: Google prueba un nuevo motor de búsqueda

¿Puede mejorarse Google? Pues parece que sí, Google puede mejorarse, y la prueba es "Caffeine". Si algo saben en el buscador más importante del mundo es que el eje de su empresa tiene un punto débil: la gente cambia de buscador con facilidad si aparece otro mejor. Y si alguien duda de esto, que recuerde cuando sólo buscaba en Yahoo, cuando se pasó a Lycos o Altavista, y qué rápido se olvidó de todos esos cuando le presentaron a Google.



Desde hace ya varios días tenemos todos a disposición Google Caffeine, la nueva y mejorada versión del buscador "Google". La misma al parecer surge en contraposición al surgimiento y buena aceptación que ha tenido Bing (buscador de Microsoft) en el mercado desde su reciente lanzamiento.

Google Caffeine es una actualización de nivel mayor de la infraestructura de Google que sigue en fase de pruebas, pero que sin embargo puede usarse desde ya al ingresar a <http://www2.sandbox.google.com/>



Entre las mejoras que hemos notado, están resultados más exactos y además un despacho a casi el doble de velocidad o más de casi cualquier resultado. Atentos webmasters y especialistas en SEO, si bien los cambios en los resultados no son drásticos, hay algunas diferencias y puede que hayan más luego del feedback que reciba Google por parte de los usuarios.

Cámara que puede ver el aura de las personas.

La tecnología cada vez va avanzando más y más, hasta lograr inventos que cada vez nos sorprenden más, y nos ponen a pensar en aquellos tiempos en que jamás hubiésemos imaginado tener dispositivos con tantas y variadas capacidades.

Konstantin Korotkov es un inventor ruso que afirma haber creado una máquina que puede ver el aura de los seres humanos, ha mostrado imágenes de una mujer tomadas a diferentes temperaturas en su "Cámara de descarga de Gas", aparentemente a través de esta cámara especial la persona que observa puede ver el aura de las personas. El experimento ha sido publicado en un medio de comunicación popular de Rusia (life.ru) donde se puede leer que aparte del aura humana, el dispositivo puede registrar circunstancias de muerte como por ejemplo diferenciar fácilmente una persona que murió por muerte natural a una que fué torturada y asesinada violentamente.



Autor

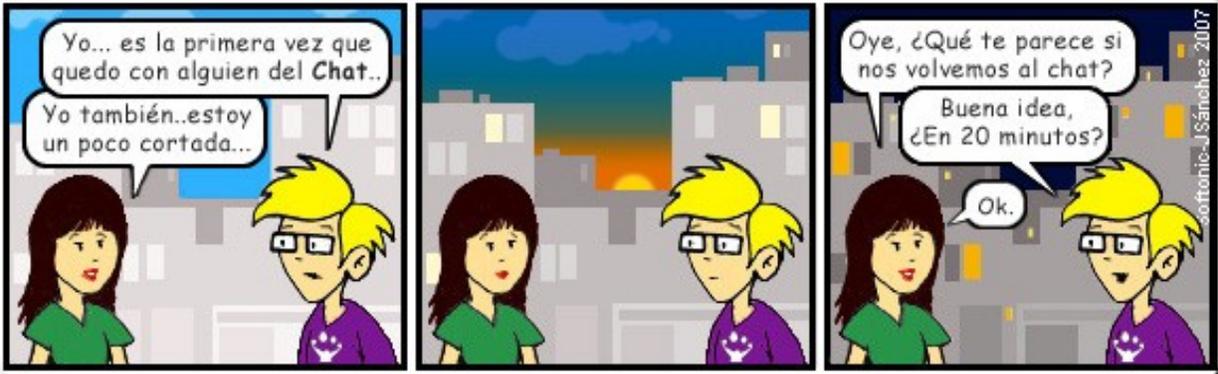


Jenny Saavedra López
Diseño y Edición Revista Atix
jennysaavedra@gmail.com

**El Software Libre
promueve los valores
éticos y morales**



Comics





Conociendo lo Nuestro

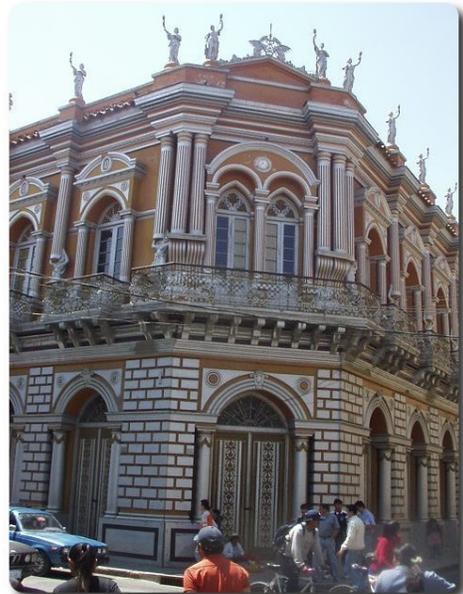
Tarija



Viñedos Tarijeños



Catedral de Tarija



Casa Dorada



Vista de la ciudad de Tarija



Danza de la Chacarera, chaco tarijeño



Plaza de Tarija

Libres para pensar, libres para decidir, libres para crear

 **Arte** **Libre** 

Te ofrecemos este espacio para mostrar tu Creatividad



Envíanos tus diseños y creaciones para publicarlos



Contacto

Para solicitar cualquier información, puedes contactar a:

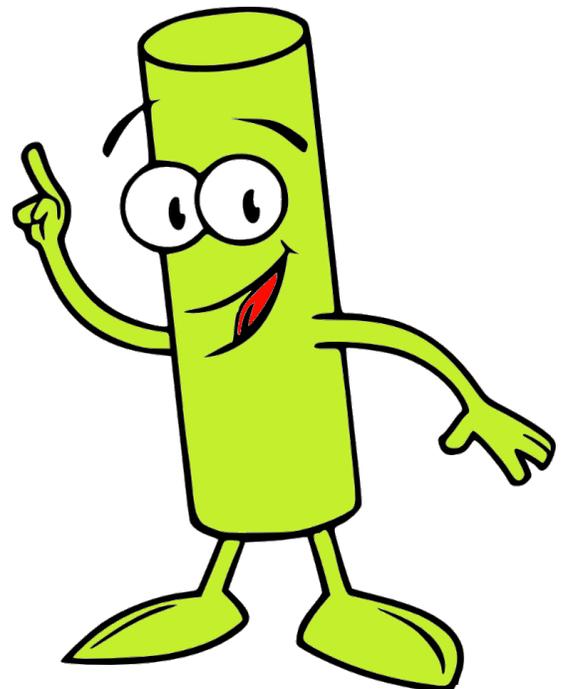
- ✓ Esteban Saavedra López (jesaavedra@opentelematics.org)
- ✓ Jenny Saavedra (jennysaavedra@gmail.com)

Publicación

Te invitamos a ser parte de la **Revista ATIX**. La forma de participar puede ser enviándonos:

- ✓ Artículos referidos a áreas como:
 - ✓ Instalación y personalización de Aplicaciones
 - ✓ Scripting
 - ✓ Diseño gráfico
 - ✓ Programación y desarrollo de aplicaciones
 - ✓ Administración de servidores
 - ✓ Seguridad
 - ✓ y cualquier tema enmarcado dentro del uso de Software Libre
- ✓ Trucos y recetas.
- ✓ Noticias.
- ✓ Comics.
- ✓ Links de interés.

Usa siempre
Software Libre



Empezamos a registrar nuestra historia



Nuestros números anteriores



Marcamos Huella



<http://atix.opentelematics.org>