

## 6. Fase de Transición

### 6.1. Ambiente

BawaMx es finalmente un sistema web el cual se ejecuta en un servidor. A continuación se describen los requerimientos de hardware y software para instalar y poner en operación el sistema.

#### 6.1.1 Requerimientos mínimos de hardware

Estas especificaciones están calculadas para que la arquitectura BawaMx pueda ejecutarse sin ningún problema una vez que el sistema desarrollado esté completo e instalado.

| Requerimiento            | Descripción                                                                    |
|--------------------------|--------------------------------------------------------------------------------|
| Procesador               | 1 o más procesadores Intel® XEON™ a 1,8 GHz, 2,0 GHz o superior o equivalente. |
| Memoria principal mínima | 2 GB                                                                           |
| Unidad de disco duro     | 1 GB o más. Dependiendo del sistema a desarrollar.                             |

Tabla 6.1.1.Descripción de requerimientos de Hardware

#### 6.1.2 Requerimientos mínimos de software para ejecución.

Estas especificaciones están diseñadas para que la arquitectura BawaMx pueda funcionar de forma óptima, sin embargo es posible encontrar muchas equivalencias compatibles dependiendo de las bondades que tenga el sistema operativo sobre el que se instala.

### 6.1.2.1 Sistema operativo

El sistema operativo ideal para trabajar con BawaMx es uno basado en Linux, se ha probado su rendimiento en Ubuntu server y Debian, sin embargo puede utilizarse cualquier sistema operativo en el que sea posible instalar Apache Tomcat.

**Sistema operativo recomendado: Ubuntu server 10.1 o superior**

### 6.1.2.2 Máquina Virtual de Java

Se necesita tener instalada la máquina virtual de java (Java Runtime Environment) JRE en la versión 1.6. o superior.

**Java Runtime Environment  
recomendado: 1.6.0\_21 o superior**

### 6.1.2.3 Servidor (contenedor de Servlets)

BawaMx es una arquitectura que utiliza archivos jsp, por ello es necesario contar un servidor web HTTP adecuado que implemente tanto los java servlets como las JSP, BawaMx ha sido únicamente desarrollado en Apache Tomcat, aunque podría utilizarse otro haciendo las adecuaciones necesarias.

**Servidor recomendado: Apache Tomcat 6.0 o superior.**

### 6.1.2.4 Framework Struts

La arquitectura BawaMx está basada en un framework llamado Struts. Este software depende directamente del servidor Apache Tomcat y funciona con la versión Struts 1.3.8 o superior.

**Framework Struts recomendado: Apache Struts 1.3.8 o superior**

### 6.1.2.5 Manejador de Base de Datos PostgreSQL

La arquitectura BawaMx ya soporta totalmente las consultas para este manejador de base de datos y es soportado para la versión 8.3.0 o superior.

**Manejador de bases de datos recomendado:**

**PostgreSQL 8.3.11 o superior**

### 6.1.2.6 Servidor de correos

La arquitectura BawaMx en su plug-in de mensajes utiliza el servidor de correos Postfix en su versión 2.5.1, sin embargo puede soportar otros como Sendmail en su versión 8.14.3 o superior.

**Servidor de correos recomendado:**

**Postfix 2.5.1 o superior**

## 6.2 Distribución del proyecto.

### 6.2.1 Distribución para desarrollo.

#### 6.2.1.1 Proyecto primitivo BawaMx.

La arquitectura BawaMx se distribuye como un proyecto Java-Web de NetBeans 6.9.1. Como BawaMx está desarrollado sobre el Framework Struts, el IDE NetBeans debe contar con dicho Framework en su versión 1.3.8 o superior, dicho proyecto contiene todo lo necesario para ser extendido y publicado incluyendo el contenido de YUI en su primera versión.

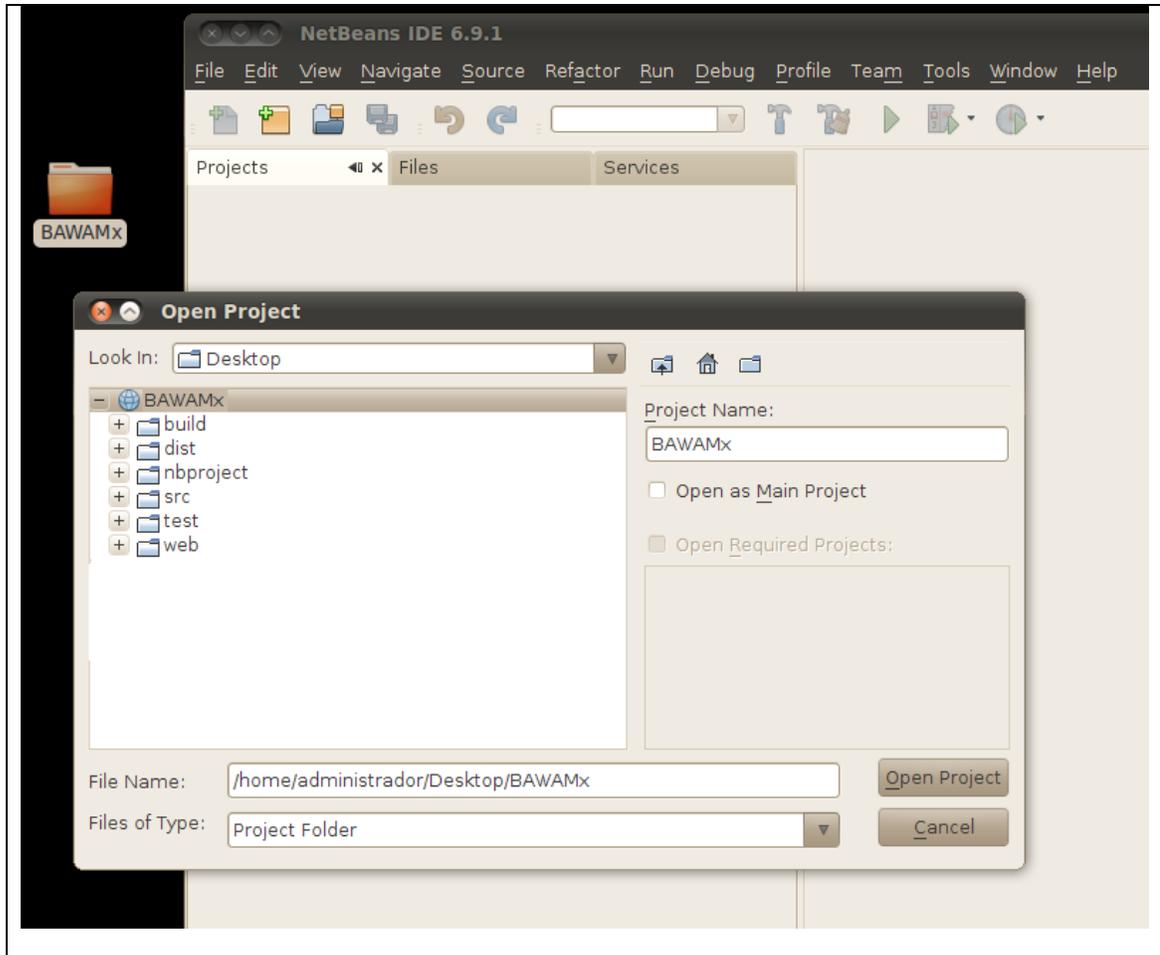


Figura 6.2.1 Vista del proyecto primitivo BawaMx en NetBeans.

### 6.2.1.2 Base de datos de BawaMx

BawaMx cuenta con una base de datos inicial para PostgreSQL. Dicha base es parte esencial de la arquitectura y se distribuye como un archivo de lenguaje de definición de datos SQL. Este archivo se encuentra dentro del proyecto de BawaMx en la carpeta de dist/DataBase.

**Ruta del archivo SQL: BAWAMx1.0/dist/DataBase/DB\_bawamx.sql**

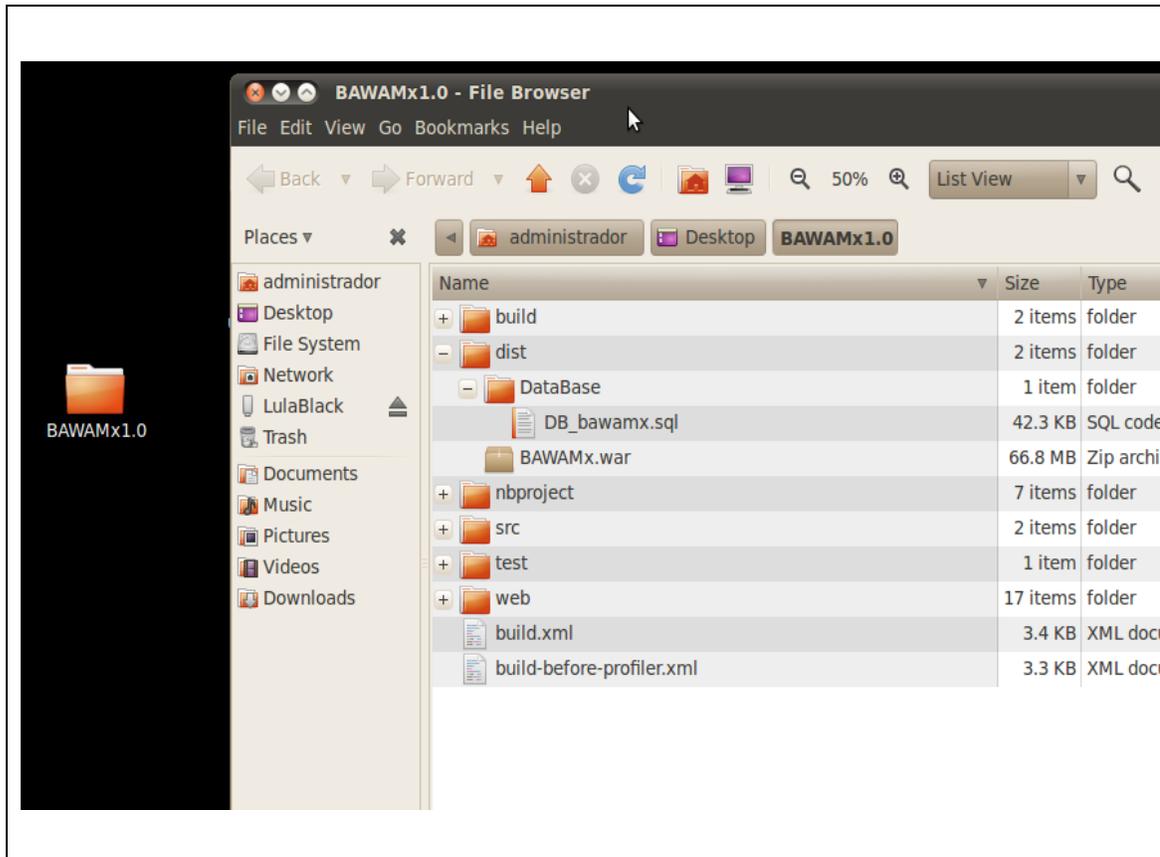


Figura 6.2.2 Localización del archivo SQL para generar la base de BawaMx.

### 6.2.2. Desplegando el proyecto.

A continuación se describe el procedimiento para instalar BawaMx con todos sus valores default.

| Paso | Descripción                                                                                                                                                                   |
|------|-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| 1.   | Descomprimir el archivo BawaMx.ZIP.                                                                                                                                           |
| 2    | Generar la base de datos con el archivo BawaMx.sql valor default :<br><a href="http://localhost:2345/bawamx">http://localhost:2345/bawamx</a><br>user: bawamx<br>Pass: bawamx |

|    |                                                                                                                                                                                                                             |
|----|-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| 3. | Abrir el TomCatManager localizar el archivo BawaMx.war<br>Cuya ruta es : BawaMx/dist/BawaMx.war y presionar deploy                                                                                                          |
| 4. | Entrar a ip/BawaMx y verificar que la página de inicio que es login.jsp.                                                                                                                                                    |
| 5. | Colocar en nombre de usuario: admin y en contraseña:admin, si se ve la página welcome_bawa.jsp entonces el proyecto a sido desplegado correctamente, con los valores default:<br>Nivel de seguridad 2 y único usuario admin |

### 6.3 Configurando la aplicación para desarrollo y despliegue.

A continuación se describen los procesos requeridos para configurar toda la arquitectura BawaMx, cuando no se desean tener los valores default.

| Paso                              | Descripción                                                                                                                                                                                                                                                                                   | Comando ejemplo                                                  |
|-----------------------------------|-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|------------------------------------------------------------------|
| 1. Generar un nuevo usuario       | Crear un nuevo usuario con permisos de creación de bases en postgres, debe de tener contraseña.<br><br>Nota: Este paso es opcional, ya que la base de BawaMx puede ser asignada a un existente de postgres, la única restricción es que este usuario debe de contar con permisos de creación. | CREATE USER<br>nuevousuario<br>WITH<br>PASSWORD<br>'contraseña'; |
| 2. Crear una base de datos vacía. | Se debe de crear una base de datos, regularmente es la base que usara el sistema web extendiendo todas las tablas y funciones que ofrece BawaMx.                                                                                                                                              | CREATE<br>DATABASE<br>mibase WITH<br>OWNER<br>nuevousuario       |
| 3.- Configurar archivo.           | Se debe de modificar el archivo DB_bawamx.sql Cambiando el dueño de todos los elementos de por el usuario al que pertenecerá la base.                                                                                                                                                         | Replace bawamx<br>With nuevousuario                              |
| 4.- Ejecutar archivo.             | Se ejecuta el archivo DB_bawamx.sql o bien si se le cambio el nombre ejecútalo y este generará todas las tablas.                                                                                                                                                                              | \i DB_bawamx.sql                                                 |
| 5.-                               | Verificar que la creación haya sido correcta.                                                                                                                                                                                                                                                 |                                                                  |

Tabla 6.2.1 Pasos para configurar la base inicial de BawaMx.

Una vez que la base de datos esté lista se le pueden agregar todas las tablas, secuencias, y funciones necesarias para el proyecto específico que se desea realizar.



Se debe de tener mucho cuidado en no eliminar o modificar ninguna función o tabla de la base inicial de BawaMx. Únicamente se permiten adiciones de tablas o funciones.

### 6.3.1 Configurando la base de datos en netbeans

Para poder utilizar la base de datos en el ambiente de desarrollo es necesario configurar el archivo bawa-config.xml en el cual se encuentran todos los parámetros para realizar la conexión a la base de datos de BawaMx, del mismo modo se pueden colocar en este archivo n conexiones a diferentes bases de datos que utilicen el módulo DataBase manager.

#### **Archivo de conexiones a bases de datos.**

Parte del archivo de bawa-config.xml que se encuentra en la carpeta WEB-inf del proyecto de Netbeans sirve para la configuración de bases de datos. BawaMx1.0 posee la siguiente estructura:

```
<xml>
  <DB_connections>
    <database name='bawamx'>
      <host value='localhost'>
        <user name='bawamx' />
        <password value='bawamx' />
        <encoding value='utf-8' />
        <driver value='postgres'>
      </database >
    .
  </DB_connections>
</xml>
```

Figura 6.2.3 Estructura del archivo DBConnections.xml

Las propiedades para la configuración de las bases de datos son las siguientes:

### **database**

La etiqueta database indica el inicio y término de todos los parámetros necesarios para la conexión a una base de datos. Los parámetros para realizar la conexión son indicados a través de hijos de la etiqueta database.

- Etiqueta: <database></database>
- Elementos anidados:
  - <user >
  - <password >
  - <encoding >
  - <host >
  - <driver >

- **Propiedades:**

**name:** Indica el nombre con que está creada la base de datos en el manejador, dicho nombre debe ser escrito tal y como está registrado respetando mayúsculas y minúsculas.

### **User**

La etiqueta user indica el usuario con el que nos vamos a conectar a la base de datos. Este usuario de preferencia debe ser el dueño de la base de datos, en caso contrario debe de ser un usuario con los permisos necesarios para realizar lectura de datos.

Etiqueta: <user/>

- Elementos anidados:
  - ninguno

- **Propiedades:**

**name:** Indica el nombre del usuario con que vamos a conectarnos a la base de datos, dicho nombre debe ser escrito tal y como está registrado respetando mayúsculas y minúsculas o símbolos.

### **password**

La etiqueta password indica la palabra clave de conexión que utiliza el usuario de la etiqueta <user>.

Etiqueta: <password/>

- Elementos anidados:
  - ninguno

- **Propiedades:**

**value:** Indica el valor de la cadena que se utilizará como contraseña, dicha cadena debe ser exactamente la que utiliza el usuario para conectarse, deben respetarse mayúsculas, minúsculas o símbolos que incluya dicha contraseña.

### **encoding**

La etiqueta encoding como su nombre lo indica contiene el valor de codificación de texto en la que la base es codificada por el momento soporta todas las codificaciones que soporta postgres:

(<http://www.postgresql.org/docs/8.1/static/multibyte.html>, dic 2010).

Etiqueta: <encoding />

- Elementos anidados:
  - ninguno

- **Propiedades:**

**value:** Indica el valor específico de la codificación de la base de datos, debe colocarse el nombre de la codificación exactamente como lo indica postgres.

### host

En la etiqueta host se indica la dirección IP de la máquina en donde se encuentra almacenada la base de datos.

Etiqueta: <host />

- Elementos anidados:
  - ninguno

- **Propiedades:**

**value:** Indica el valor específico del huésped de la base de datos, regularmente dado por una dirección y un puerto. En esta propiedad se puede utilizar "localhost" sin ningún problema.

### driver

En la etiqueta driver se indica el manejador sobre el cual está desarrollada la base de datos, este debe ser soportado por java sql connection.

Etiqueta: <driver />

- Elementos anidados:

- ninguno
- **Propiedades:**
  - value:** Indica el valor específico del driver por ejemplo: com.mysql.jdbc.Driver, org.postgresql.Driver, etc.

### 6.3.2 Librerías

Algunas veces el proyecto de netbeans pierde las referencias a las librerías que utiliza BawaMx. Para resolver el problema de referencias hay que indicar la posición de las librerías, todas ellas se encuentran en la carpeta de web del proyecto de netbeans \BAWAMx1.0\web\lib. A continuación se listan y se describe para que se utilizan:

- Gjson.-Se utiliza para transformar objetos a formato json
- Mail.- Facilita el envío de los mensajes electrónicos.

### 6.3.3 Manejo del Plug-in de seguridad

En general es posible proveer a cualquier aplicación de un nivel de seguridad que activa diferentes filtros en el sistema haciendo más robusto y seguro el inicio de sesión y navegación de un usuario dentro del sistema.

BawaMx contará con un plug-in de seguridad, dicho **plug-in** deberá estar indicado en el archivo de configuración de struts conocido como struts-config.xml, el plug-in contiene las propiedades que sirven para configurar su compartimento. Dichas propiedades se describen a continuación.

### 6.3.3.1 Propiedades del Plug-in de Seguridad

#### Security\_Level

#### Nivel de seguridad

Descripción: Indica el número de filtros utilizados por el sistema para establecer los accesos permitidos de cada usuario.

#### Valores

Valor '0'.- Indica que el sistema se ejecutará sin ningún filtro de seguridad e inmediatamente entrará a `welcome_page` indicada en el archivo `web.xml`. Este nivel no utiliza las acciones ni los permisos.

Valor '1'.- Este valor hace que en cualquier sistema se active el filtro de autenticación. Éste debe verificar si el usuario que desea usar el sistema está registrado, si es así lo deja utilizar todos los recursos del sistema. Por otra parte, se llena la base de datos con la que BawaMx trabaja incluyendo sus usuarios y permisos.

También se activan las siguientes propiedades con sus valores por defecto:

| Propiedad                           |
|-------------------------------------|
| <code>initial_forward</code>        |
| <code>enable_initial_forward</code> |
| <code>welcome_page_template</code>  |
| <code>welcome_page_path</code>      |
| <code>welcome_page</code>           |
| <code>action_config_path</code>     |
| <code>action_login</code>           |
| <code>action_logout</code>          |

|                        |
|------------------------|
| <b>encoding_system</b> |
| <b>enable_logs</b>     |
| <b>enable_md5</b>      |

Valor '2'.- Este valor hace que en cualquier sistema se activen dos filtros de seguridad el primero es la autenticación y el segundo es la autorización, el primer filtro verifica si el usuario que desea usar el sistema está registrado. Si es así, el sistema lo reconoce, una vez que ha sido reconocido por el sistema se debe verificar si dicho usuario tiene permiso de visitar o consultar los recursos que solicita.

Este nivel de seguridad al igual que el anterior, llena la base de datos con la que BawaMx trabaja incluyendo sus usuarios y sus permisos.

Se activan las siguientes propiedades con sus valores por defecto:

| Propiedad                     |
|-------------------------------|
| <b>initial_forward</b>        |
| <b>enable_initial_forward</b> |
| <b>welcome_page_template</b>  |
| <b>welcome_page_path</b>      |
| <b>welcome_page</b>           |
| <b>action_config_path</b>     |
| <b>action_login</b>           |
| <b>action_logout</b>          |
| <b>encoding_system</b>        |
| <b>enable_logs</b>            |
| <b>enable_md5</b>             |
| <b>action_fail</b>            |

- **Valor por defecto:** '1'.
- **Valores posibles:** '0','1','2'

- Nota: si se escribe otro valor diferente de '1','2' o '0' en la propiedad de `security_level` se cambiará por default a nivel '1'.

### **Initial\_forward.**

Indica cual va ser el primer redireccionamiento (forward) del sistema. Regularmente en el archivo inicial contiene únicamente una etiqueta `<forward>` de struts la cual indica la primera página a mostrar. Dicha página puede ser un action o bien un jsp. Si se desea manipular este valor basta indicar el path o action en la configuración del plug-in.

- **Valor default:** 'login.jsp'.
- **Valores posibles:** cualquier ruta a un jsp valido, o un action.

### **Enable\_initial\_forward.**

Activa o desactiva el redireccionamiento inicial del sistema. Esta bandera existe porque no siempre se desea que haya un forward inicial aunque esté activado uno de los niveles de seguridad, ni siquiera el default. Por ello si deseamos que el index no contenga concatenada la etiqueta inicial `<forward>` inicial se modificará esta propiedad.

- **Valores posibles:**
  - 'true'.-Activa el `initial_forward` en la página inicial indicada en el `struts_config.xml`.
  - 'false'.-Desactiva el `initial_forward` no importando que nivel de

seguridad esté activado.

- **Valor default:** 'true'.

BawaMx ofrece tres formas para cambiar la página inicial :

- a) Modificar el web.xml e indicar en el plug-in la propiedad de welcome\_page y asegurarse de poner la propiedad en enable\_forward en false.
- b) Hacer un action y colocarle el forward deseado en su propiedad, e indicar el action en el initial\_forward. Nota: si este no se incluye en la lista de <unfiltered actions> del action-config.xml los filtros se activarán y harán que su aplicación funcione anormalmente.
- c) Simplemente cambiar el valor del initial\_forward a la primera página jsp deseada y en esta los filtros no se activaran si dichas páginas no tienen un <forward page= <action>>

### **Welcome\_page\_template**

Es el nombre de una plantilla jsp, en la cual es posible colocar cabeceras y código a la primera página que se mostrará en el sistema como por ejemplo:

```
<% @ Page import = "text / html"%>  
<% @ Page pageEncoding = "UTF-8"%>
```

welcome\_page\_template al igual que initial\_forward, enable\_initial\_forward, welcome\_page y welcome\_page\_template se utilizan para poder construir la página de inicio del sistema.

- **Valores posibles:** Cualquier página jsp que se desee de plantilla.
- **Valor por default:** 'index\_template.jsp'

### welcome\_page

Es exactamente la misma página de bienvenida del web.xml. Ésta se utiliza para reconstruir la página de inicio como se desea.

- **Valores posibles:** Cualquier página jsp. válida y existente
- **Valor por default:** 'index.jsp'

### Welcome\_page\_path

La ruta de acceso de la página inicial. Es útil porque se construye cada vez que se arranque la aplicación.

- **Valores posibles:** path válido donde va estar almacenada la welcome\_page de la aplicación.
- **Valor default:** '././ web /'
-

### Action\_config\_path

Lugar en todo el sistema donde se albergará el archivo de configuración de las acciones. Dicho archivo es esencial para el funcionamiento correcto del plug-in de seguridad y regularmente esta propiedad es indicada respecto la dirección absoluta del equipo donde este almacenado la aplicación web.

- **Valores posibles:** <cualquier dirección valida donde se encuentre el action-config.xml>
- **Valor default:** './web/WEB-INF/action-config.xml'

Nota: En esta versión de BawaMx todas las rutas deben ser absolutas.

### Action\_login

El action que se activa cuando el sistema tiene en la propiedad security\_level '1' y '2', es decir filtros activados, si un usuario trata de entrar a un recurso no permitido el sistema lo detecta como usuario no autenticado mandándolo a la acción de login o algún otro que realice autenticación.

- **Valores posibles** :<acción válida para que un usuario haga login>
- **Valor default:** '/Login'

### Action\_logout

El action que debe activarse cuando un usuario ha dejado el sistema si hay algún filtro activado.

- **Valores posibles** : <acción valida cuando un usuario cerro sesión>

- **Valor default:** '/Login'

Nota: Es recomendable utilizar el action que provee BawaMx para el cierre de sesión, ya que este action realiza algunas funciones de limpieza de sesión específicas de BawaMx.

### Action\_fail

El action\* que debe activarse cuando un usuario no tiene permisos de acceder a un action esto es cuando el security\_level es '2'.

- **Valores posibles:** <acción válida cuando un usuario no tiene permisos para entrar >
- **Valor default:** '/Login'

\*Esta acción debe estar indicada como unfiltered siempre.

### Enable\_logs

Esta propiedad hace que se registre en la base de datos de BawaMx cada vez que un usuario mande a llamar un action en nivel '0' solo se guarda la ip y id de action, mientras que en nivel '1' y '2' se guarda la ip, el usuario y el id de action.

- **Valores posibles:**
  - 'true'.-Activa los logs
  - 'false'.-Desactiva los logs
- **Valor default:** true

## Enable\_md5

Este hace que todas las contraseñas de los usuarios ingresados en el sistema sean leídos como md5, para niveles de seguridad '1' y '2'.

- **Valores posibles:**
  - 'true'.-Activa el md5 en passwords
  - 'false'.-Desactiva md5 en passwords
- **Valor default:** 'true'

## Sincronization\_type

Elige cómo se hace la sincronización de la base de datos. Existen dos formas posibles para la sincronización:

- **Valores posibles:**
  - 'query'.- A través de consultas.
  - 'Function'. -A través de funciones que existen dentro de la base de datos inicial de BawaMx.

Valor default: 'function'

### 6.4.3.1 Archivo XML de configuración de acciones del plug-in de Seguridad

El funcionamiento de los filtros del plug-in de seguridad está basado en un archivo XML con las siguientes propiedades:

## Groups

Los groups son conjuntos formados por n elementos de tipo group donde

$n \geq 1$ . Esta etiqueta debe siempre tener por lo menos un hijo.

- Etiqueta: `<groups></groups>`
- Elementos anidados: `<group>`
- **Propiedades:** ninguna.

## Group

Un group es un conjunto de acciones válidas y existentes en el archivo `struts-config.xml` llamadas actions, este conjunto de acciones deberá tener un nombre único para diferenciarlo de los demás.

- Etiqueta: `<group></group>`
- Elementos anidados:
  - `<action>`
  - `<no-action>`
- **Propiedades**
  - `name`.-Debe contener un nombre único entre todos los grupos. Este nombre acepta valores alfanuméricos y los signos bajos y medios.
  - `set`.- Acepta únicamente dos valores posibles "all" y "none" si no se coloca la etiqueta o bien no se coloca ningún valor de los aceptados se pondrá el valor default que es "none". El valor de la propiedad "all" asigna todas las acciones a un grupo y por lo tanto ya no es necesario poner todas las acciones en dicho grupo.

Regularmente se coloca el `set="all"` cuando se desean quitar algunas

acciones de todo el conjunto definido en el archivo struts-config.xml utilizando la etiqueta <no-action>, o bien cuando se quieren asignar todas las acciones de una sola vez.

“None” no asigna ninguna acción por default pero se pueden agregar por medio de la etiqueta <action>, si no se coloca la propiedad por defecto es “none”.

- unfiltered.-Esta propiedad recibe únicamente dos valores: “true” y “false”, cuando vale “true”, todas las acciones pertenecientes a ese grupo no serán tomadas en cuenta por los filtros y cualquiera podrá acceder a ellas, por default está en ‘false’.

Nota: Hay que tener un poco de cuidado en no repetir acciones dentro de los grupos con la bandera unfiltered ya que éstos tienen prioridad en los filtros.

## Action

Un Action es exactamente la misma que está indicada en el struts-config.xml en el action-config.xml esta siempre debe ser una etiqueta hija de <group>.

- Etiqueta:<action><action/>
- Elementos anidados:
  - ninguno
- **Propiedades:**

- name.- Un nombre que se le puede asignar a una acción para posteriores búsquedas del sistema a realizar.
- path.- Exactamente el indicado en el struts-config.xml. Esta propiedad es OBLIGATORIA.

### no-Accion

Esta etiqueta sirve para quitar una acción (action) de un grupo cuando se ha asignado la propiedad set="all". Ésta debe ser una acción válida en el struts-config.xml. Ésta siempre debe ser una etiqueta hija de <group>.

- **Etiqueta:** <no-action><no-action/>
- Elementos anidados:
  - ninguno
- **Propiedades:**
- path.- Exactamente el indicado en el struts-config.xml. Esta propiedad es OBLIGATORIA.

### Roles

La etiqueta roles indica un conjunto de elementos rol.

- Etiqueta:<roles><roles/>
- Elementos anidados:
  - <rol>
- **Propiedades:**

- name: Es el nombre del rol. Este debe ser único entre todos los roles.

## Rol

Un rol es un tipo de usuario el cual tiene asignado a él varios grupos de acciones (groups), estas acciones son sobre las que se tiene permiso de ejecución, todo usuario deberá tener un rol, un rol debe tener por lo menos un hijo:<set-group>

- **Etiqueta:**<rol><rol/>
- Elementos anidados:
  - <set-group>

- **Propiedades:**

Ninguna

## set-group

Set-group asigna un grupo de acciones a un rol. Estas acciones son las que se tiene permiso de ejecutar.

- Etiqueta:<rol><rol/>
- Elementos anidados: <set-group>
- **Propiedades:**
- name.- El nombre del grupo de acciones descrito en los <groups>

**users**

Es un conjunto de elementos user.

- Etiqueta:<users><users/>
- Elementos anidados:<user>
- **Propiedades**

ninguna

**User**

Los usuarios que utilizarán el sistema serán ingresados a la base de datos inicial de BawaMx. Estos necesariamente necesitan su nombre, login, password y un rol. Estos datos serán esenciales cuando se elija en los niveles de seguridad 1 y 2. Estos usuarios tendrán acceso a todos los lugares indicados en sus acciones y grupos cuando esté activado el nivel 2 de seguridad.

- **Etiqueta:**<user ><user/>
- Elementos anidados:<action>
- **Propiedades:**

ninguna

**Action-authorization**

La etiqueta que indica que se trata del archivo de configuración de acciones de BawaMx. Esta etiqueta debe ser la madre de todas aquellas etiquetas que no necesitan estar contenidas en alguna otra etiqueta.

- **Etiqueta:** <action-authorization></ action-authorization >
- Elementos anidados:

- <roles>
- <groups>
- <users>

- **Propiedades**

ninguna

#### **6.4.3.2 Funcionamiento General del Plug-in de Seguridad.**

El plug-in construye la página de inicio dinámicamente, dependiendo del nivel de seguridad dado. Si éste es '0', la página que se mostrará al inicio de la aplicación es la que está indicada en el archivo web.xml, por otro lado si el nivel de seguridad es diferente de cero, al iniciar la aplicación se mostrará en el navegador lo que se conoce como el initial\_forward.

Existe una serie de procesos que se ejecutan cuando está iniciando la aplicación: la página inicial es construida al igual que los permisos asentados en el archivo de configuración de acciones (action\_config.xml). También se realiza la sincronización con la base de datos inicial de bawaMx.

Es importante recordar que cuando se almacenan los usuarios, un campo esencial es el password. Si el desarrollador coloca en el archivo de configuración md5=true todas las contraseñas se consideran transformadas a md5 inclusive en el archivo de action\_config.xml. Una vez que la aplicación inicia, toda la seguridad de la misma queda habilitada para cualquier usuario que haga uso de ella. Cada vez que una página es visitada el sistema verifica si el usuario tiene permisos de hacerlo como lo muestra el siguiente diagrama:

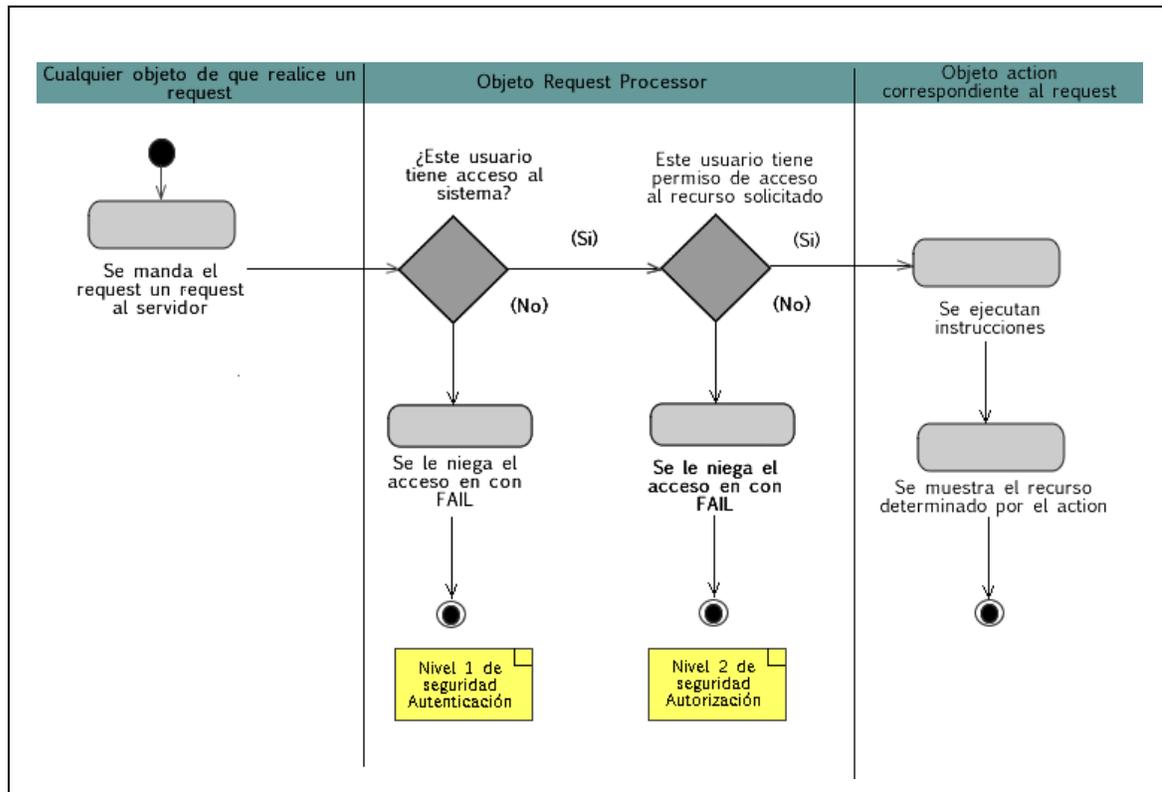


Figura 6.2.3 Diagrama del funcionamiento del plug-in de seguridad

### 6.3.4 Manejo de DataBase Manager

Los desarrolladores podrán hacer uso del Data manager para realizar conexiones y se recomienda crear un paquete para cada una de las bases de datos a las que se desea conectar. Este paquete debe contener por lo menos una clase de conexión. La clase de conexión debe de existir para poder conectarse a una base de datos específica.

#### Class DBconnection

java.lang.Object

└ lmi.bawamx.model.db\_manager.DBconnection

Direct Known Subclasses:

[DBConnection\\_Bawamx](#)

|                                                                                                                                                                                                            |                              |
|------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|------------------------------|
| <pre>public class <b>DBconnection</b> extends java.lang.Object  Makes possible an easy connection with any Data Base.</pre>                                                                                |                              |
| <b>Field Summary</b>                                                                                                                                                                                       |                              |
| protected                                                                                                                                                                                                  | <a href="#">dbmq</a>         |
|                                                                                                                                                                                                            | <a href="#">DBMakerQuery</a> |
| <b>Constructor Summary</b>                                                                                                                                                                                 |                              |
| <a href="#">DBconnection</a> (java.lang.String dbDriver,<br>java.lang.String dblocalhost, java.lang.String database,<br>java.lang.String dbuser, java.lang.String dbpassword,<br>java.lang.String CharSet) |                              |
| <a href="#">DBconnection</a> (java.lang.String dbName)                                                                                                                                                     |                              |

Figura 6.2.4 Documentación de la Clase DBConnection.

El primero de los constructores es recomendado cuando la base de datos tiene bien definido el host y su dirección no cambia después del desarrollo del sistema. El segundo utiliza el archivo de conexiones a bases de datos DBConnections.xml y necesita únicamente el nombre de la base de datos en el constructor. Este nombre de ser anotado exactamente igual que en el archivo DBconnections, de lo contrario sucederá un error.

Una vez que se defina la clase de conexión, se debe definir también una clase por cada tabla de la base de datos y éstas deben siempre heredar del objeto de conexión. Si se tienen dudas sobre cómo se generan estas relaciones se puede consultar la documentación de la conexión a la base inicial de BawaMx.

**mi.bawamx.model.db\_bawamx**

**Class DBConnection\_Bawamx**



Figura 6.2.5 Todas las subclases del DBConection\_Bawamx

### 6.3.5 Manejo de File Manager

En BawaMx existen dos tipos de operaciones que facilitan el manejo y creación de archivos. El primer tipo cubre operaciones básicas que utilizan los desarrolladores casi siempre en la capa de modelo, dichas operaciones son:

| Nombre                            | Operación                              | Descripción                                                                                           |
|-----------------------------------|----------------------------------------|-------------------------------------------------------------------------------------------------------|
| <b>CreategenericTextFile</b>      | Creación de archivo genérico (texto)   | Crea un archivo genérico que contiene texto, este archivo puede ser guardado con cualquier extensión. |
| <b>DeleteFile</b>                 | Eliminación de archivo                 | Elimina cualquier archivo dada su extensión, (tener cuidado con los permisos)                         |
| <b>Create_ifNotExistDirectory</b> | Creación de directorio si no existe    | Crea un directorio dada una ruta                                                                      |
| <b>Delete_althoughContent</b>     | Eliminación de contenido de directorio | Elimina contenido de un directorio                                                                    |
| <b>DeleteDirectory</b>            | Eliminación de todo el directorio      | Elimina el directorio con todos los archivos contenidos en ella.                                      |
| <b>RenameDirectory</b>            | Renombrar de directorio                | Cambia el nombre de un directorio                                                                     |
| <b>Rename file</b>                | Renombra archivo                       | Cambia el nombre de un archivo                                                                        |
| <b>ZipFile</b>                    | Comprimir archivo                      | Crear un archivo comprimido ZIP a partir de un directorio                                             |
| <b>unzipFile</b>                  | Descomprimir archivo                   |                                                                                                       |

Tabla 6.2.2 Operaciones sobre archivos fuera de la vista

El segundo tipo de operaciones son las que casi siempre se utilizan en la capa de vista, ya que son completamente destinadas a interactuar con el usuario, dichas operaciones son:

| Nombre            | Operación                    | Descripción                                                                                                       |
|-------------------|------------------------------|-------------------------------------------------------------------------------------------------------------------|
| <b>UploadFile</b> | migra un archivo al servidor | Copia un archivo de una máquina local al servidor, debe haber un action o formulario relacionado a esta operación |
| <b>getPdfFile</b> | Crea archivo PDF             | Crea un archivo PDF a partir de la vista en el navegador, debe haber un action relacionado a esta operación       |
| <b>getXlsFile</b> | Crea archivo XLS             | Crea un archivo XLS a partir de la vista en el navegador, debe haber un action relacionado a esta operación       |

Tabla 6.2.3 Operaciones sobre archivos fuera de la vista

El uso de las dos últimas operaciones debe estar relacionado con un action porque es el recurso que utilizan los desarrolladores para la generación de los archivos.

### 6.3.5 Manejo de YUI Manager

Yui manager por el momento sólo soporta 3 tipos de componentes YUI con dos importantes características.

- Transporte transparente de datos provenientes de una acción a un componente YUI.
- Renderización automática del componente y sus datos (pasados a Json), evita la escritura de código javascript repetitivo.
- Manejo de acciones básicas de un objeto y posibilidad de realizar actions.

| Componente YUI       | Nombre de la etiqueta |
|----------------------|-----------------------|
| <b>Datatable</b>     | <byui:DataTable>      |
| <b>Drag&amp;Drop</b> | <byui:DragDrop>       |
| <b>Tree</b>          | <byui:Tree>           |

Tabla 6.2.4 Lista de componentes disponibles en BawaMx

### 6.3.5.1 Yui manager en una acción.

Para poder utilizar las tags que ofrece BawaMx los datos deben haber sido trasladados a los objetos dispuestos para la comunicación con YUI. A continuación se muestra la tabla de relación entre objetos y las etiquetas que YUI ofrece.

| Nombre de objeto         | Taglib para YUI     |
|--------------------------|---------------------|
| <b>YUI_Tree</b>          | <bawa-yui:tree>     |
| <b>YUI_Drag&amp;Drop</b> | <bawa-yui-dragdrop> |
| <b>YUI_Table</b>         | <bawa-yui-table>    |

Tabla 6.2.5 Lista de componentes disponibles en BawaMx

Hay que recordar que el objeto a colocar es el que de la propiedad data de la etiqueta correspondiente, la cual básicamente es un objeto de la clase yui-manager, el cual permite la construcción de los objetos YUI en simples pasos.

### 6.3.6 Manejo de Flash Manager

El módulo de flash manager provee de una interfaz simple para tener interacción entre el servidor y cualquier película flash. Para utilizarlo se requieren los siguientes pasos:

- 1) Utilizar el archivo de configuración para indicar las acciones utilizadas en la película flash así como los servidores que se usan en cada caso.
- 2) Crear un jsp en donde se despliega la respuesta sin formato del action.

- 3) Crear un objeto BawaFlashConection en el cual se coloca la etiqueta del action a utilizar. El archivo BawaFlashConection se encuentra en la distribución del proyecto en el directorio WEB.
- 4) Los datos obtenidos, es decir, la respuesta obtenida por el servidor puede desplegarse regularmente en un objeto flash que sea capaz de mostrar texto.

### 6.3.6.1 Archivo de configuración de Flash Manager

El archivo de configuración de flash tiene la siguiente estructura:

```
<?xml version="1.0" encoding="utf-8" ?>
  <flab_action name="" value="" server="" />
</xml>
```

Figura 6.2.6 Estructura del archivo de configuración de las acciones para Flash

La ventaja de que todas las acciones que se utilizarán dentro de la película flash estén integradas a través de una etiqueta, es que una vez que se culminó la asignación de action por etiqueta, si se desea cambiar el action, sólo es necesario cambiar la configuración y no el código.

Otra ventaja es que el flash manager de BawaMx se encarga de pasar las variables deseadas al servidor sin tener que realizarlo manualmente, simplemente se utiliza el método add ( ) del objeto FlashConnection.

Por otra parte es cuando se desean enviar beans a la película flash y obtener los datos, basta con pasarlos como un objeto BeanToFlash en el request del action. Esto asegura que podamos recibir los datos ya sea en forma de un flashArray o un String.

Es necesario que siempre exista el jsp para cargar la respuesta, además de que es un requerimiento obligado para el input del action. Este tipo de modelo permite probar el código sin la necesidad de utilizar forzosamente la película de Flash ayudando a dividir el trabajo entre lógica de negocios y el diseño.