

# SIGES

## Requerimientos Técnicos y Manual de Instalación



# Índice

1	Requerimientos Técnicos.....	5
1.1	Requerimientos de Hardware .....	5
1.2	Red .....	5
1.3	Requerimientos de Software .....	6
2	Procedimiento de instalación .....	7
2.1	Sobre la aplicación .....	7
2.2	Instalación de Java .....	7
2.3	Instalación de JBoss AS.....	7
2.4	Instalación de MySQL Server 5.6.....	8
2.5	Crear la base de datos para la aplicación.....	8
2.6	Crear un módulo para el conector de MySQL en JBoss AS .....	8
2.7	Configurar el origen de datos para la aplicación en JBoss AS .....	9
2.8	Configurar el servidor de correo electrónico .....	9
2.9	Configurar el dominio de autenticación.....	10
2.10	Configurar los niveles de log .....	10
2.11	Autorizar el acceso a la aplicación desde otros equipos.....	11
2.12	Habilitar el acceso seguro (HTTPS).....	11
2.13	Habilitar la compresión de datos .....	12
2.14	Iniciar el JBoss AS .....	13
2.15	Instalar la aplicación en el JBoss AS .....	13
2.16	Acceder a la aplicación.....	13
2.17	Cambio de puertos del JBoss AS .....	13



# 1 Requerimientos Técnicos

El objetivo de este capítulo es detallar las necesidades adecuadas del sistema en hardware y software para desplegar la aplicación en un ambiente de producción o testing. Está dirigido al todo el personal técnico y administradores de sistemas. Se detallarán los requerimientos y especificaciones de hardware sin profundizar en detalles, los cuales quedan a cargo de cada oficina o departamento de informática.

## 1.1 *Requerimientos de Hardware*

Ambiente de producción con un único servidor:

- Requisitos recomendados: Memoria RAM 12 GB, espacio en disco 100 GB
- Requisitos mínimos: Memoria RAM 6 GB, espacio en disco 20 GB

Ambiente de producción para servidores separados:

Servidor de Aplicaciones:

- Requisitos recomendados: Memoria RAM 8 GB, espacio en disco 40 GB
- Requisitos mínimos: Memoria RAM 4 GB, espacio en disco 20 GB

Servidor de Base de Datos:

- Requisitos recomendados: Memoria RAM 4 GB, espacio en disco 60 GB
- Requisitos mínimos: Memoria RAM 2 GB, espacio en disco 20 GB

Se recomienda una conexión de red de 1000 Mbit/s.

Ambiente de testing:

Para el servidor de testing se recomienda un único servidor (Srv. Aplicaciones y Base de Datos) con:

- Memoria RAM: 4 GB.
- Espacio en disco: 20 GB.
- Conexión red: no menor a 100Mbit/s.

En todos los casos es indiferente si estos requerimientos se aplican en un equipo físico o en una máquina virtual.

## 1.2 *Red*

Acceso Usuarios

El sistema de producción debe quedar disponible y al alcance de los usuarios que lo utilicen. Es decisión de cada organismo y personal técnico si la aplicación tiene un alcance de uso interno del organismo y/o externo, teniendo para este último, acceso desde Internet.

Las políticas de seguridad para el uso interno como externo no son detalladas en este documento y dependen de cada organismo.

#### Acceso personal técnico

El personal técnico (Administradores, Desarrolladores, etc.) deberá tener acceso para las siguientes tareas:

- Administración general y configuración del Sistema Operativo y aplicaciones.
- Administración de la Base de Datos.
- Administración del Servidor de Aplicaciones.

### **1.3 *Requerimientos de Software***

A continuación se detalla el software a ejecutar en dicho sistema.

- Sistema Operativo: preferentemente Linux (CentOS, Debian, etc.)
- Servidor de Aplicaciones: jBoss 7.1
- Servidor de Base de Datos: MySQL 5.6
- Java: versión 1.7

## 2 Procedimiento de instalación

En este capítulo se detallan los pasos necesarios para realizar la instalación del sistema.

### 2.1 Sobre la aplicación

La aplicación se ejecuta sobre un servidor de aplicaciones **JBoss AS versión 7.1.1** y almacena sus datos en una base de datos **MySQL Server 5.6**. Es necesario instalar dichos productos antes de proceder a instalar la aplicación en sí misma, como se explica en este documento. Adicionalmente, es necesario contar con un entorno de ejecución **Java (JRE) versión 7** instalado en el sistema.

A continuación se explica la instalación de cada uno de estos componentes.

### 2.2 Instalación de Java

Para funcionar, JBoss AS requiere que exista una instalación de Java 7 en el sistema. Si no lo hay es necesario instalar uno. Se puede obtener un instalador apropiado a partir de la siguiente URL (asegurarse que se trate de Java 7):

<http://www.oracle.com/technetwork/java/javase/downloads/index.html>

Una vez descargado el instalador apropiado se debe ejecutarlo, siguiendo las instrucciones proporcionadas por el mismo.

Tanto en el caso de que ya estuviese instalado el entorno de ejecución de Java, como en el caso de haber tenido que instalarlo, antes de continuar se debe verificar que exista una variable de entorno denominada **JAVA\_HOME**, cuyo valor sea la ruta completa hasta el directorio donde se encuentra instalado el entorno de ejecución de Java 7. La forma de configurar esta variable depende del sistema operativo en uso (por más detalles sobre cómo configurar variables de entorno del sistema operativo consultar el manual del mismo). Como ejemplo, si el directorio de instalación del entorno de ejecución de Java 7 es “C:\Program Files\Java\JRE7”, entonces el valor de la variable de entorno **JAVA\_HOME** debe ser también “C:\Program Files\Java\JRE7”.

### 2.3 Instalación de JBoss AS

JBoss AS es un servidor de aplicaciones de libre distribución. Puede descargarse gratuitamente desde la siguiente URL (asegurarse que se trate de JBoss AS 7.1):

<http://www.jboss.org/jbossas/downloads/>

Luego de descargarlo, la instalación consiste simplemente en descomprimir el archivo descargado (un archivo comprimido de tipo ZIP) en la ubicación deseada (que será la ubicación final); dicha ubicación será el llamado “directorio de instalación” del servidor de aplicaciones, y de aquí en más será denotado como “**{JBOSS\_HOME}**”.

Para probar el funcionamiento correcto del servidor de aplicaciones se puede ejecutar el script **standalone.bat** (en MS Windows) o **standalone.sh** (en Linux) que se encuentra dentro del directorio **{JBOSS\_HOME}/bin**. Si la instalación es correcta, el servidor de aplicaciones JBoss 7 debería iniciar en unos pocos segundos, mostrando el mensaje “JBoss AS 7.1.1.Final 'Brontes' started”. Si en lugar de ello muestra un error y no inicia, comprobar la configuración de la

variable de entorno JAVA\_HOME como fue explicado en el punto anterior, y la instalación de Java 7.

Configurar los valores de Xmx y Xms, para esto editar el archivo standalone.conf en linux o standalone.conf.bat en Windows que se encuentra en la carpeta {JBoss\_HOME}/bin y modificar la siguiente línea:

```
JAVA_OPTS="-Xms2024m -Xmx2024m -XX:MaxPermSize=512m -  
Djava.net.preferIPv4Stack=true -Dorg.jboss.resolver.warning=true -  
Dsun.rmi.dgc.client.gcInterval=3600000 -Dsun.rmi.dgc.server.gcInterval=3600000"
```

por los valores correctos. Los valores correctos dependen de la memoria RAM y del CPU del servidor donde se está realizando la instalación. Se recomienda como mínimo 2024m.

## 2.4 Instalación de MySQL Server 5.6

MySQL Server también es un producto de distribución libre (en su versión Community). Puede descargarse gratuitamente en la siguiente URL (asegurarse que se trate de la versión 5.6):

<http://dev.mysql.com/downloads/mysql/>

Luego de descargar el instalador, ejecutarlo, y seguir los pasos indicados por el mismo. Como parte del proceso de instalación será necesario establecer una contraseña para el usuario root (el usuario administrador de la instalación). Se recomienda permitir que el instalador configure MySQL como servicio de forma tal que inicie automáticamente cuando se enciende el equipo.

## 2.5 Crear la base de datos para la aplicación

Para crear la base de datos para la aplicación se debe iniciar sesión en el servidor MySQL usando la cuenta root y ejecutar el script sigesXXXX.sql proporcionado. Puede usarse el programa mysql incluido en la instalación de MySQL, o el cliente gráfico MySQL Workbench que puede descargarse del mismo sitio donde se descargó el instalador.

Al finalizar la ejecución del script de creación de la base de datos se recomienda crear una cuenta de usuario específica para la aplicación SIGES, que será utilizada para la comunicación entre el servidor de aplicaciones (JBoss AS) y el servidor de bases de datos (MySQL Server). Si bien puede utilizarse la cuenta root no se recomienda por razones de seguridad. Para crear una nueva cuenta debe ejecutarse la siguiente sentencia SQL:

```
GRANT ALL ON siges.* TO {user}@{host} IDENTIFIED BY '{pass}';
```

donde {user} debe ser remplazado por el nombre de usuario deseado (por ejemplo, "sigesusr"), {pass} por la contraseña deseada (por ejemplo, "sigespwd") y {host} por la dirección IP del equipo donde reside el servidor de aplicaciones JBoss AS (por ejemplo, "192.198.2.68"); si el servidor de aplicaciones JBoss AS y el servidor de bases de datos MySQL Server residen en el mismo equipo, puede usarse el alias "localhost".

## 2.6 Crear un módulo para el conector de MySQL en JBoss AS

Para que el servidor de aplicaciones JBoss AS pueda conectarse al servidor de bases de datos MySQL Server, es necesario crear un módulo con la configuración apropiada. Para hacerlo se debe acceder a la carpeta modules de JBoss AS ({JBoss\_HOME}/modules) y crear la siguiente estructura de directorios: **com/ mysql/main**. Luego, en la carpeta **main**, se deben copiar los archivos **mysql-connector-java.jar** y **module.xml** proporcionados.



## 2.7 Configurar el origen de datos para la aplicación en JBoss AS

Para que la aplicación pueda conectarse a la base de datos MySQL Server a través del servidor de aplicaciones JBoss AS, es necesario definir un origen de datos en este último. Para hacerlo, se debe abrir el archivo **standalone.xml** que se encuentra en el directorio de configuración de JBoss AS (**{JBoss\_HOME}/standalone/configuration/**) y buscar el subsistema cuyo nombre es **"urn:jboss:domain:datasources:1.0"**; dentro de este subsistema, identificar la sección **<datasources>** y dentro de ella copiar el siguiente texto:

```
<datasource jta="true" jndi-name="java:jboss/datasources/MySqlDS"
pool-name="MySqlDS" enabled="true">
  <connection-url>jdbc:mysql://{host}:{port}/siges</connection-url>
  <driver>com.mysql</driver>
  <pool>
    <min-pool-size>10</min-pool-size>
    <max-pool-size>100</max-pool-size>
  </pool>
  <security>
    <user-name>{user}</user-name>
    <password>{pass}</password>
  </security>
  <validation>
    <validate-on-match>false</validate-on-match>
    <background-validation>false</background-validation>
  </validation>
  <statement>
    <share-prepared-statements>true</share-prepared-statements>
  </statement>
</datasource>
```

**Nota:** reemplazar **"{host}"** y **"{port}"** por la dirección IP y número de puerto donde escucha el servidor de bases de datos MySQL (si el servidor de bases de datos MySQL reside en el mismo equipo que el servidor de aplicaciones JBoss AS puede usarse **"localhost"**; el puerto por defecto, a menos que se haya especificado otra cosa durante la instalación es **"3306"**). También reemplazar **"{user}"** y **"{pass}"** por el nombre de usuario y la contraseña especificadas al crear la cuenta de usuario en el punto 2.5.

A continuación, dentro del mismo subsistema, ir hasta la sección **<drivers>** y copiar el siguiente texto:

```
<driver name="com.mysql" module="com.mysql">
  <xa-datasource-class>
    com.mysql.main.mysql-connector-java
  </xa-datasource-class>
</driver>
```

Luego de realizar estos cambios, guardar y cerrar el archivo.

## 2.8 Configurar el servidor de correo electrónico

Para que la aplicación pueda enviar correos electrónicos es necesario definir un servidor SMTP en el servidor de aplicaciones JBoss AS. Para hacerlo, se debe continuar editando el archivo **standalone.xml**. En este archivo, se debe buscar la sección **<socket-binding-group>** (normalmente se encuentra al final del archivo) e incluir dentro de ella el siguiente texto:

```
<outbound-socket-binding name="mail-siges-smtp">
  <remote-destination host="{host}" port="{port}" />
</outbound-socket-binding>
```

**Nota:** reemplazar “{host}” y “{port}” por la dirección IP o el nombre y el puerto del servidor SMTP que será usado para enviar correos electrónicos.

A continuación, se debe buscar el subsistema “urn:jboss:domain:mail:1.0”, y dentro de él copiar el siguiente texto:

```
<mail-session jndi-name="java:jboss/mail/SigesMail" from="{dirección mail from}">  
  <smtp-server ssl="true" outbound-socket-binding-ref="mail-siges-smtp">  
    <login name="{user}" password="{pass}" />  
  </smtp-server>  
</mail-session>
```

**Nota:** reemplazar {dirección mail from} por la dirección del correo que utilizará siges para enviar mails y “{user}” y “{pass}” por el nombre de usuario y la contraseña en el servidor de correo SMTP que el servidor de aplicaciones JBoss debe utilizar para conectarse cada vez que la aplicación requiera el envío de un mensaje de correo electrónico. Normalmente, el nombre de usuario tendrá el formato de una dirección de correo electrónico (por ejemplo, siges@agesic.gub.uy).

Luego de realizar estos cambios, guardar y cerrar el archivo.

**Notas:** El sistema posee dos parámetros que son necesario configurar, esto se hace ingresando al sistema como PMOT en el menú Administración – Configuración - General (es necesario hacer un clic sobre el botón “buscar”):

1. General cambiar valor del parámetro “CON\_CORREO” a “true”.
2. En el parámetro “MAIL\_FROM” ponerle la casilla de correo que va a utilizar el sistema para enviar mails.

## 2.9 Configurar el dominio de autenticación

La aplicación delega al servidor de aplicaciones JBoss la autenticación de los usuarios que intentan acceder a ella (validar nombres de usuario y contraseña). Para que el servidor de aplicaciones JBoss AS pueda encargarse de esta tarea, debe ser configurado un dominio de autenticación. Para hacerlo, se debe continuar editando el archivo **standalone.xml**. En este archivo, se debe buscar el subsistema “urn:jboss:domain:security:1.1” y copiar dentro de él el siguiente texto:

```
<security-domain name="form-auth" cache-type="default">  
  <authentication>  
    <login-module  
      code="com.sofis.web.seguridad.SimpleUsernamePasswordLoginModule"  
      flag="required"/>  
    </login-module>  
  </authentication>  
</security-domain>
```

## 2.10 Configurar los niveles de log

Está previsto que la aplicación escriba sus propios mensajes de log en el archivo de log del servidor de aplicaciones. El nivel de detalle de los mensajes registrados es configurable. Para hacerlo, se debe continuar editando el archivo **standalone.xml**. En este archivo, se debe buscar el subsistema “urn:jboss:domain:logging:1.1” y copiar dentro de él el siguiente texto:

```
<logger category="siges">
  <level name="ALL"/>
</logger>
<logger category="com.sofis">
  <level name="ALL"/>
</logger>
```

Para reducir el nivel de log, puede remplazarse "ALL" por uno de "DEBUG", "INFO", "WARN", "ERROR" (de mayor a menor).

## 2.11 Autorizar el acceso a la aplicación desde otros equipos

Por defecto, el servidor de aplicaciones JBoss AS limita el acceso a las aplicaciones que contiene, permitiendo acceder a ellas únicamente desde el propio equipo donde reside (localhost). Normalmente, los usuarios acceden a las aplicaciones desde otros equipos de la red. Para permitir este tipo de acceso, se debe continuar editando el archivo **standalone.xml**. En este archivo, se debe buscar la sección **<interfaces>**, y dentro de ella modificar la configuración de la interfaz cuyo nombre es **"public"**, cambiando el valor "127.0.0.1" por **"0.0.0.0"**, debiendo resultar algo similar a lo siguiente (solo la línea que está en negrita fue modificada):

```
<interfaces>
  <interface name="management">
    <inet-address value="{jboss.bind.address.management:127.0.0.1}"/>
  </interface>
  <interface name="public">
    <inet-address value="{jboss.bind.address:0.0.0.0}"/>
  </interface>
  <interface name="unsecure">
    <inet-address value="{jboss.bind.address.unsecure:127.0.0.1}"/>
  </interface>
</interfaces>
```

Luego de realizar estos cambios, guardar y cerrar el archivo.

## 2.12 Habilitar el acceso seguro (HTTPS)

La aplicación está diseñada para aceptar conexiones solo por HTTPS (HTTP + SSL/TLS). Sin embargo, por defecto el servidor de aplicaciones JBoss AS no acepta dicho tipo de conexiones. Para habilitarlo se deberá contar con un keystore (un repositorio de certificados digitales) que contendrá únicamente al certificado digital que entregará el servidor de aplicaciones a los usuarios (en realidad, a los navegadores web de los mismos) cada vez que inicien una comunicación nueva. Para esto se tienen dos casos: cuando se cuenta con un certificado digital emitido por una autoridad certificadora reconocida (CA), y cuando no se tiene dicho certificado. En ambos casos se creará un keystore nuevo, llamado **siges.jks**, para asegurar que no se utilice otro certificado no deseado.

En el caso de que se cuente con un certificado digital reconocido, éste debe ser importado al keystore con el siguiente comando:

```
keytool -v -importkeystore -srckeystore {archivo} -srcstoretype PKCS12 -destkeystore siges.jks -deststoretype JKS
```

**Nota:** remplazar {archivo} por el nombre (incluyendo la ruta completa si no está en el directorio actual) del archivo que contiene el certificado digital.

Al ejecutar este comando el programa keytool solicitará la contraseña del certificado, que debe haber sido proporcionada por el emisor del mismo y la contraseña del keystore, que debe ser la misma que la del certificado.

En el caso de que no se cuente con un certificado digital reconocido, se puede crear uno propio, aunque se debe tener en cuenta que todos los usuarios verán un mensaje de advertencia, al acceder a la aplicación, indicándoles que el servidor utiliza un certificado digital que no es de confianza. Para crear un nuevo keystore conteniendo el nuevo certificado digital, se debe ejecutar el siguiente comando:

```
keytool -genkeypair -keystore siges.jks -alias siges
```

Al ejecutar este comando el programa keytool solicitará primero el ingreso de la contraseña para el keystore (habrá que ingresarla dos veces) y luego el ingreso de la información que será incluida en el certificado. Los únicos datos obligatorios son el primero (“nombre”, ingresar “Agesic Siges”) y el último (“país”, ingresar “UY”), en el resto de los casos se puede dejar en blanco. Finalmente el programa keytool solicitará confirmación, a lo que se debe responder “sí” (o “yes” si está en inglés). Para terminar, el programa keytool solicitará la nueva contraseña para el certificado, que debe ser la misma que se usó para el keystore.

**Nota:** si al intentar ejecutar alguno de los comandos anteriores se observa un mensaje indicando que no se reconoce al comando keytool como un programa válido, es porque el directorio de programas de Java no está en la lista de directorios donde el sistema operativo busca programas. En este caso se puede ejecutar el mismo comando especificando la ruta completa al programa keytool (que está en el directorio bin de la instalación del JRE), o modificar la variable de entorno PATH (tanto en MS Windows como en Linux) para incluir el directorio bin de la instalación del JRE.

Una vez creado el keystore, con cualquiera de los dos casos, se debe indicarle al servidor de aplicaciones JBoss que permita las conexiones seguras, y que utilice el keystore creado anteriormente, para lo cual se debe continuar editando el archivo **standalone.xml**. En este archivo se debe buscar el subsistema llamado “**urn:jboss:domain:web:1.1**” y dentro de él reemplazar la definición del conector HTTP (la línea que comienza con “<connector name=“http” ...>” por el siguiente texto (quitar la línea mencionada y poner en su lugar las siguientes):

```
<connector name="http" protocol="HTTP/1.1" scheme="http" socket-binding="http"
redirect-port="8443"/>
<connector name="https" protocol="HTTP/1.1" scheme="https" socket-
binding="https" secure="true">
  <ssl name="ssl" key-alias="{alias}" password="{pass}" certificate-key-
file="{jboss.server.config.dir}/siges.jks" protocol="TLSv1"/>
</connector>
```

**Nota:** reemplazar {alias} por el alias del certificado dentro del keystore (si no se conoce, puede ejecutarse el siguiente comando: keytool -list -keystore siges.jks) {pass} por la contraseña del keystore y del certificado (que deben coincidir).

## 2.13 Habilitar la compresión de datos

Dado que la aplicación puede transferir mucha información hacia los clientes, es recomendable habilitar la compresión para acelerar la transferencia. Para hacerlo, se debe editar el archivo **standalone.xml** y copiar al principio, justo debajo de las extensiones (luego del tag de cierre </extensions>) el siguiente texto:

```
<system-properties>
```

```
<property name="org.apache.coyote.http11.Http11Protocol.COMPRESSION"
value="on"/>
<property
name="org.apache.coyote.http11.Http11Protocol.COMPRESSION_MIME_TYPES"
value="text/javascript,text/css,text/html,text/xml,text/json"/>
</system-properties>
```

## 2.14 Iniciar el JBoss AS

Para iniciar el servidor de aplicaciones JBoss AS se debe ejecutar el script **standalone.bat** (en MS Windows) o **standalone.sh** (en Linux) que se encuentran en la carpeta **{JBoss\_HOME}/bin**, de la misma manera que se hizo al finalizar la sección 2.3.

## 2.15 Instalar la aplicación en el JBoss AS

Para instalar ("deployar") la aplicación en el servidor de aplicaciones JBoss AS, solo se debe copiar el archivo **SS-SIGES-ear-1.0-SNAPSHOT.ear** proporcionado a la carpeta **{JBoss\_HOME}/standalone/deployments**. Tras unos pocos segundos en dicha carpeta aparecerá un nuevo archivo llamado **SS-SIGES-ear-1.0-SNAPSHOT.ear.deployed**, que indica que la aplicación se ha instalado correctamente; si no es así, o el nombre del archivo es **SS-SIGES-ear-1.0-SNAPSHOT.ear.failed**, significa que ocurrió un error y debe examinarse el archivo de log (**{JBoss\_HOME}/standalone/log/server.log**) para intentar determinar la causa del error.

## 2.16 Acceder a la aplicación

Para acceder a la aplicación se debe iniciar un navegador web (Internet Explorer, Mozilla Firefox, Google Chrome) y apuntarlo a cualquiera de las siguientes URLs:

`http://{host}:{port}/SS-SIGES-web`

**Nota:** reemplazar {host} para dirección IP o el nombre del equipo donde reside el servidor de aplicaciones JBoss, y {port} por el número de puerto por el cual atiende conexiones (por defecto es 8080). Si todo funciona correctamente, debería verse la pantalla inicial de la aplicación Siges, solicitando el nombre de usuario y la contraseña; si no es así, se debe mirar el log del servidor para intentar detectar posibles errores.

Utilizar las siguientes credenciales para acceder:

- Usuario: admin
- Contraseña: admin

Luego de ingresar al sistema se podrá cambiar la contraseña.

## 2.17 Cambio de puertos del JBoss AS

Por defecto, los puertos usados por JBoss AS para atender conexiones son el 8080 para HTTP y 8443 para HTTPS. Si se desea cambiar esto, se debe hacer lo siguiente:

- Para cambiar el puerto HTTP: editar el archivo **standalone.xml** y buscar la línea que comienza con "**<socket-binding name="http"**", reemplazando en ella 8080 por el puerto deseado.
- Para cambiar el puerto HTTPS: editar el archivo **standalone.xml** y buscar la línea que comienza con "**<socket-binding name="https"**", reemplazando en ella 8443 por el puerto deseado. Luego, buscar la línea que comienza con "**<connector name="http"**" y

remplazar en ella 8443 por el puerto deseado, que debe ser el mismo que el indicado anteriormente.