

Trabajo fin de carrera

SaveMeSafe 

Manual de instalación y configuración

Semestre 2. Curso 2014-2015

Alumno: Ángel Arias Romero
Consultor: Vicenç Font Sagrista

Tabla de contenido

1	Introducción	2
2	Requisitos técnicos.....	3
2.1	Software	3
2.2	Hardware.....	3
3	Composición del proyecto software	4
4	Instalación y configuración de PostgreSQL	4
4.1	Instalación	4
4.2	Configuración	5
5	Instalación y configuración de JBoss	7
5.1	Instalación	7
5.2	Configuración	8
6	Instalación de la herramienta Apache Ant.....	9
7	Construcción del EAR de la aplicación.....	10
7.1	Compilación y empaquetado	10
7.2	Despliegue en JBoss y arranque de la aplicación	11
8	Entorno de desarrollo con Eclipse.....	12

1 Introducción

En el presente documento se explica detalladamente los pasos a seguir para poder compilar, empaquetar y desplegar la aplicación en el servidor de aplicaciones JBoss. Para ello hay que instalar en primer lugar el software que la aplicación necesita para poder funcionar, su entorno de ejecución, en segundo lugar hay que realizar la configuración necesaria para crear la base de datos que utilizará la aplicación para almacenar los datos que gestiona.

2 Requisitos técnicos

2.1 Software

En el siguiente listado tenemos el software con la versión necesaria para el correcto funcionamiento de la aplicación.

- [JDK 1.7 versión 1.7.0-51](#)
- [Ant, versión 1.9.3](#)
- [PostgreSQL versión 9.4](#)
- [Conector Java Postgresql-9.4-1201.jdbc41](#)
- [JBoss WildFly, versión 8.2.0 Final](#)
- [Eclipse Luna 4.4.2](#)
- [JBoss Tools 4.2.3.Final](#)
- [Primefaces 5.2](#) (ya incluido en el empaquetado EAR de la aplicación)

Todo el software indicado es libre por lo que puede ser descargado de Internet y utilizado sin ninguna restricción importante.

Aunque todas las pruebas y el desarrollo de este software se han llevado a cabo sobre Windows 7, no debería haber ningún problema en utilizarlo sobre cualquier sistema basado en Linux para el que se encuentren disponibles todos los componentes software que indicábamos anteriormente.

El software mínimo para poder compilar y ejecutar la aplicación es la máquina virtual Java, la herramienta Apache Ant, el servidor de aplicaciones JBoss y el gestor de base de datos PostgreSQL junto a su driver para Java.

En el caso de requerir un entorno de depuración y pruebas sería necesario instalar un IDE, que en este caso ha sido Eclipse Luna (aunque no debería haber problemas en utilizar otros como NetBeans o JDevelopers). Con el IDE es recomendable utilizar la facilidades de integración con JBoss que ofrece el plugin JBoss Tools.

2.2 Hardware

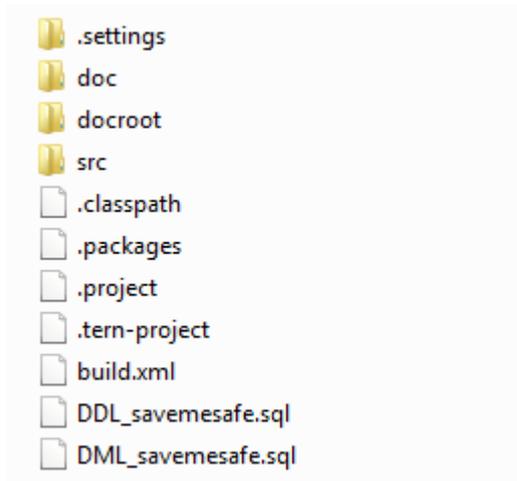
Al tratarse de una aplicación web donde toda la lógica y el procesamiento se encuentra de forma remota al cliente -solo se necesita un browser para acceder a la aplicación- los componentes hardware que son necesarios dependen directamente de la carga que pueda soportar la aplicación. Por tanto, se consideran los de la máquina donde se desplegará el servidor de aplicaciones JBoss y el sistema gestor de base de datos PostgreSQL.

Para un funcionamiento mínimo los requisitos serán los siguientes:

- 1 GB de RAM
- Procesador Intel i3 o equivalente
- Espacio en disco de 10 Gb

3 Composición del proyecto software

El código fuente de la aplicación se distribuye en el fichero `savemisafe.zip` cuyo contenido es el siguiente:



A continuación se comentan los distintos tipos de ficheros y directorios que contiene el empaquetado:

- Todos los ficheros y directorios que empiezan por '.' serán utilizados por el IDE Eclipse para poder importar el proyecto.
- En el directorio `doc` se encuentra la documentación del código en formato `html`.
- El directorio `docroot` almacena la estructura y recursos que conformará el fichero WAR (Web Application Archive), que contiene la parte de presentación de la aplicación.
- El fichero `build.xml` será utilizado por la utilidad Apache Ant para la compilación, construcción del EAR y su despliegue en JBoss.
- Los ficheros con extensión `.sql` son los scripts necesarios para la creación del esquema de la base de datos y de la carga de datos de ejemplo.
- El directorio `src` contiene el código fuente de la aplicación y algunos ficheros necesarios para JBoss para poder interpretar las definiciones de JSF, JPA los EJB3.

4 Instalación y configuración de PostgreSQL

4.1 Instalación

En primer lugar es necesario descargar la versión indicada anteriormente del programa desde la página oficial de este gestor de base de datos relacional <http://www.postgresql.org>.

En el apartado de requisitos software del presente documento se encuentra el enlace directo a la descarga de este producto con la versión requerida.

Durante el proceso de instalación lo único reseñable es la posibilidad elegir directorios distintos para la instalación del run-time de PostgreSQL y otro directorio distinto para los datos que gestionará. El resto de opciones es recomendable mantenerlas con sus valores por defecto. Es importante considerar que la contraseña que se establezca (que estará asociada al superusuario `postgres`) será posteriormente utilizada para configurar el origen de datos de la aplicación en el servidor de aplicaciones JBoss (también es posible crear nuevos roles para esto).

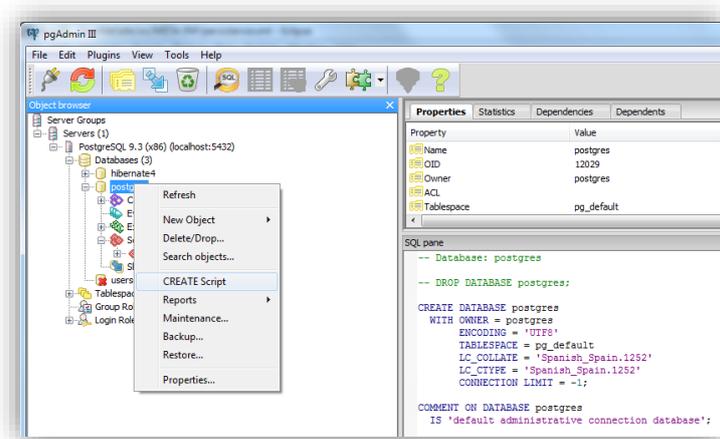
Con la instalación de PostgreSQL se incluye la aplicación pgAdmin III con la que resultará sencillo administrar y realizar posteriormente algunas consultas o verificaciones sobre la base de datos.

4.2 Configuración

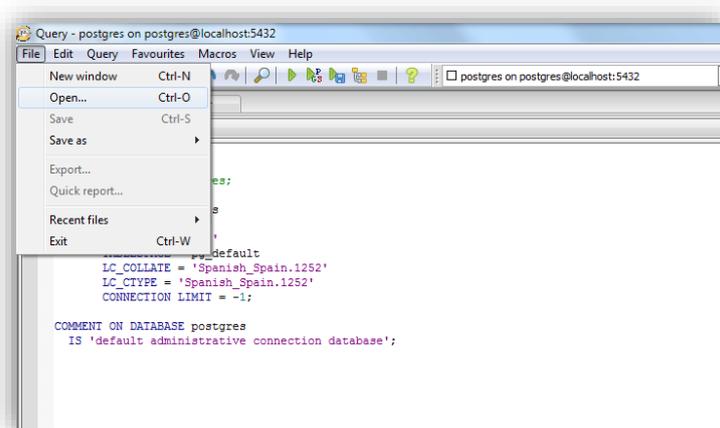
Como se ha comentado, con la aplicación pgAdmin III es posible crear nuevos usuarios y roles para la base de datos. En este caso, por simplificar, vamos a utilizar directamente el usuario por defecto y la contraseña que hemos definido durante el proceso de instalación.

Inicialmente PostgreSQL dispondrá de una base de datos con nombre *postgres*. Aunque es posible crear una base de datos nueva y trabajar sobre ella por simplicidad vamos a trabajar sobre esta base de datos inicial.

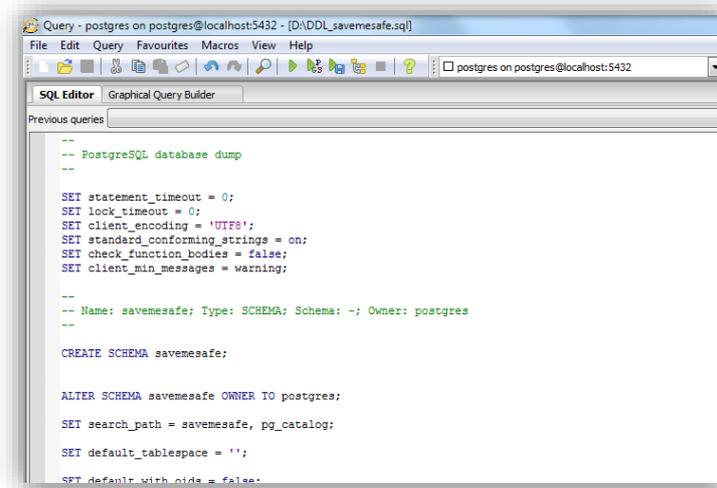
Creamos el esquema con las definiciones y tablas que necesita la aplicación. Para ello se suministra un script llamado *DDL_savemesafe.sql* que incluye todas las sentencias SQL necesarias. Para cargar este script habría que hacer clic con el botón derecho sobre la base de datos *postgres* y elegir la opción “Create Script”.



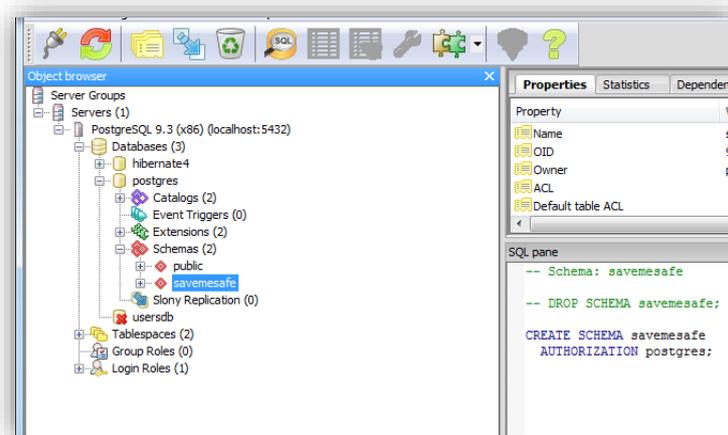
En la nueva pantalla debemos elegir el fichero *DDL_savemesafe.sql*, mediante la secuencia “File -> Open...”



Una vez cargado el script es necesario ejecutarlo desde la botonera superior, en el botón verde con forma de flecha.



Una vez ejecutado podemos ver que se ha creado el nuevo esquema con nombre savemesafe.



Una vez creado el esquema ya tendríamos la base de datos preparada para la aplicación. De forma opcional es posible cargar los datos de ejemplo incluidos en el fichero DML_savemesafe.sql siguiendo los mismos pasos que para ejecutar el script de creación del esquema sobre la base de datos con nombre postgres.

Con la carga del fichero DML_savemesafe.sql podemos hacer uso de un usuario para pruebas con los siguientes datos de acceso:

- Nombre de usuario: userapp
- Contraseña del usuario: userapp

5 Instalación y configuración de JBoss

Es requisito fundamental haber instalado previamente la máquina virtual Java, con versión JDK 1.7 para que pueda funcionar JBoss. Para validar si tenemos disponible Java en el sistema podemos comprobar si existe la variable de entorno `JAVA_HOME` apuntando al directorio de instalación de esta versión de Java, en nuestro caso (`JAVA_HOME = "D:\jdk1.7.0_67"`).

También podemos verificarlo lanzando el comando `java -version` en la consola del Windows.

5.1 Instalación

En el apartado de requisitos software del presente documento se encuentra el enlace directo a la descarga de este producto con la versión requerida.

La instalación del servidor de aplicaciones JBoss no requiere lanzar un instalador, solo será necesario descomprimir el fichero zip (recomendable en la raíz de alguna unidad o en un directorio sin espacios en blanco en su ruta de acceso). En este caso los hemos descomprimido en el directorio `D:\wildfly-8.2.0.Final` por lo que esta será la ruta absoluta sobre la que trabajaremos.

Creamos una variable del sistema con el nombre `JBOSS_HOME` y como valor la carpeta donde está instalado JBoss. Una vez declarada esta variable de entorno, para ejecutar JBoss tendremos que ir al subdirectorio `bin` del directorio donde hemos instalado JBoss y ejecutar el comando `standalone.bat`.

Para comprobar si se ha instalado correctamente sólo debemos ejecutar el navegador y acceder a la siguiente dirección: <http://localhost:8080>. Si aparece la ventana siguiente o similar, es que hemos instalado y ejecutado correctamente JBoss:



5.2 Configuración

5.2.1 Creación de un usuario

Para tener acceso a la consola de JBoss (pulsando sobre "Administration Console"), y también para poder acceder de forma remota a los componentes de JBoss, es necesario dar de alta usuarios de tipo `ManagementRealm` y `ApplicationRealm`, respectivamente. Para ello utilizaremos la herramienta `add-user.bat` que se encuentra en la carpeta `bin` del directorio de instalación de JBoss.

A continuación se muestra de forma simplificada cómo crear un usuario de tipo `ApplicationRealm` utilizando la herramienta `add-user.bat`.

```
What type of user do you wish to add?
a) Management User (mgmt-users.properties)
b) Application User (application-users.properties)
(a): B
Enter the details of the new user to add.
Realm (ApplicationRealm):
Username : savemesafe
Password : operador
Re-enter Password : operador
What roles do you want this user to belong to? (Please enter a comma separated list, or leave blank
for none) : User,Trainer,Administrator
About to add user 'USER' for realm 'ApplicationRealm'
Is this correct yes/no? Y
Added user 'USER' to file 'D:\wildfly-8.2.0.Final\standalone\configuration\ application-
users.properties'
Added user 'USER' to file 'D:\wildfly-8.2.0.Final\domain\configuration\ application-users.properties'
Added user 'USER' with roles User,Trainer,Administrator to file 'D:\wildfly-8.2.0.Final
\standalone\configuration\application-roles.properties'
Added user 'USER' with roles User,Trainer,Administrator to file 'D:\wildfly-8.2.0.Final
\domain\configuration\application-roles.properties'
```

5.2.2 Configuración del origen de datos

Para que JBoss pueda intermediar en el acceso a la base de datos por parte de la aplicación es necesario instalar el controlador JDBC para PostgreSQL y definir el `datasource` en JBoss.

Haciendo uso de la herramienta `jboss-cli.bat`, que se encuentra en el directorio `bin` de la instalación de JBoss, podemos hacerlo de forma sencilla. El importante tener en cuenta que el uso de esta herramienta requiere que JBoss esté arrancado.

Para iniciar esta utilidad debemos ejecutar `jboss-cli.bat` y escribir el comando `connect`. Una vez conectados podemos lanzar los comandos que se indican a continuación.

1- La instalación del driver para PostgreSQL se puede realizar utilizando el siguiente comando (en `resource` hay que indicar la ruta absoluta del jar correspondiente al driver):

```
module add --name=org.postgres --resources=D:\postgresql-9.4-1201.jdbc41.jar
--dependencies=javax.api,javax.transaction.api
```

2- A continuación podemos crear el `datasource`:

```
/subsystem=datasources/jdbc-driver=postgres:add(driver-name="postgres",driver-
module-name="org.postgres",driver-class-name=org.postgresql.Driver)
```

3- En este comando es necesario indicar la URI, el usuario y la contraseña de acceso a PostgreSQL (en este caso `postgres/postgres`):

```
data-source add --jndi-name=java:jboss/postgresDS --name=postgresDS
--connection-url=jdbc:postgresql://localhost:5432/postgres
--driver-name=postgres --user-name=postgres --password=postgres
```

Para salir de esta utilidad utilizamos el comando `quit`.

5.2.3 Configuración del servidor de correo SMTP

Debido a que la aplicación necesita poder enviar correos electrónicos a los usuarios, es necesario realizar algunas definiciones en JBoss para delegar en el servidor la gestión de los recursos necesarios.

Del mismo modo que para la creación del origen de datos, vamos a utilizar la utilidad `jboss-cli.bat` para registrar todo lo necesario en JBoss. Como en el caso anterior, tras lanzar la utilidad es necesario ejecutar el comando `connect` para inicial la ejecución de los comandos, y `quit` tras finalizar.

La cuenta de correo que vamos a utilizar en la aplicación para el envío de correos electrónicos pertenece al dominio Gmail (de Google Inc) y sus datos son los siguientes:

- Nombre de usuario: savemesafe.app@gmail.com
- Contraseña: `savemesafe`
- Servidor de correo SMTP: `smtp.gmail.com`
- Puerto: `465`

Los comandos a ejecutar son los siguientes:

1- Definimos el servidor de correo SMTP para la cuenta de Gmail:

```
/socket-binding-group=standard-sockets/remote-destination-outbound-socket-binding=mail-smtp-gmail:add(host=smtp.gmail.com,port=465)
```

2- Añadimos el recurso JNDI:

```
/subsystem=mail/mail-session=App:add(jndi-name=java:/mail/gmail,from="savemesafe.app@gmail.com",debug=true)
```

3- Indicamos las credenciales de acceso al servidor SMTP de Gmail:

```
/subsystem=mail/mail-session=App/server=smtp:add(outbound-socket-binding-ref=mail-smtp-gmail, username=savemesafe.app@gmail.com, password=savemesafe, ssl=true)
```

6 Instalación de la herramienta Apache Ant

Ant es una herramienta muy utilizada en el mundo del desarrollo en Java debido a que permite automatizar tareas repetitivas, principalmente durante la fase de compilación y empaquetado del software.

En el apartado de requisitos software del presente documento se encuentra el enlace directo a la descarga de este producto con la versión requerida.

Como se ha comentado anteriormente, para el desarrollo de esta aplicación se ha utiliza la versión 1.9.5 de esta utilidad. Una vez descargada, tan solo es necesario descomprimir el empaquetado y definir la variable de entorno `ANT_HOME` referenciando al directorio de instalación de Ant con su ruta absoluto. También es necesario añadir el directorio `bin` al `PATH` del sistema, en nuestro caso `ANT_HOME=D:\apache-ant-1.9.5` y `PATH=%ANT_HOME%/bin;%PATH%` respectivamente.

Podemos validar la correcta instalación de Ant lanzando el comando `ant -version` en una consola de Windows.

7 Construcción del EAR de la aplicación

7.1 Compilación y empaquetado

Llegados a este punto tenemos ya tenemos prácticamente todo preparado para poder probar la aplicación. A continuación se explica cómo poder compilar y empaquetar la aplicación con la utilidad Ant.

Las siguientes variables deben de estar definidas:

- `JAVA_HOME`: debe contener el directorio donde hemos instalado JDK (por ejemplo `D:\jdk1.7.0_51`)
- `JBOSS_HOME`: debe contener el directorio donde se ha instalado JBoss (por ejemplo `D:\wildfly-8.2.0.Final`)
- `ANT_HOME`: debe contener el directorio donde se ha instalado Ant (por ejemplo `D:\apache-ant-1.9.3`).
- `PATH`: debe contener el subdirectorio bin de todo el software que utilizamos: `%JAVA_HOME%\bin;%ANT_HOME%\bin;%JBOSS_HOME%\bin;%PATH%`
- `CLASSPATH`: debe contener el fichero `tools.jar` del JDK, tal como se ha explicado anteriormente por ejemplo `%JAVA_HOME%\lib\tools.jar`;

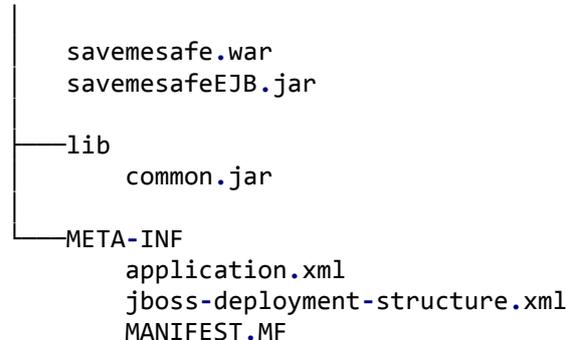
En el empaquetado entregado se incluye el fichero `build.xml` que será el que utilice Ant para ejecutar las sentencias encargadas de compilar, empaquetar e incluso desplegar el EAR en el directorio de despliegue de JBoss.

Para ejecutar Ant por línea de comandos sobre nuestro proyecto es necesario estar posicionarnos en el directorio raíz del proyecto, justo donde se encuentra el fichero `build.xml`. Una vez aquí posicionados bastaría con escribir el comando `ant` en una consola de Windows para que se compile y despliegue la aplicación en JBoss. Si solo queremos que compile y genere el fichero EAR, sin que sea desplegado en JBoss, habría que utilizar el comando `ant ear`.

Como resultado de la ejecución de Ant obtendremos un fichero EAR (Enterprise ARchive), un formato de empaquetado y distribución de aplicaciones empresariales utilizadas en desarrollos JEE.

La estructura de este fichero EAR será la siguiente:

savemesafe.ear



El fichero `savemesafe.war` contiene la lógica de presentación de la aplicación que incluye los Managed Bean y las páginas xhtml de JSF junto los otros recursos adiciones (como fichero de estilos `.ccs`, imágenes...).

La librería `java savemesafeEJB.jar` incluye la lógica de negocio y de persistencia de la aplicación, incluyendo las clases JPA y EJB3 remotos.

En el directorio `lib` se encuentra la librería `common.jar` que incluye clases comunes o que intermedian entre la lógica de presentación y de negocio. Con este empaquetado el servidor de aplicaciones utiliza un `ClassLoader` Java común.

El directorio `META-INF` contiene los ficheros de configuración necesarios para el correcto despliegue de la aplicación así como las definiciones de dependencias entre las distintas librerías Java que conforman la aplicación.

7.2 Despliegue en JBoss y arranque de la aplicación

Es despliegue de la aplicación consiste en copiar el fichero `savemesafe.ear` generado con Ant en el directorio `JBOSS_HOME/standalone/deployments` de JBoss para que el despliegue se realice de forma automática en el arranque del servidor.

Para arrancar el servidor es necesario ejecutar el script `standalone.bat` que se encuentra dentro del directorio `bin` de la instalación de JBoss (`JBOSS_HOME/bin`).

```
...
13:13:42,868 INFO [org.jboss.weld.deployer] (MSC service thread 1-4) JBAS016002: Processing weld deployment savemesafe.ear
...
13:13:42,948 INFO [org.jboss.weld.deployer] (MSC service thread 1-10) JBAS016002: Processing weld deployment savemesafe.war
...
13:13:42,958 INFO [org.jboss.weld.deployer] (MSC service thread 1-5) JBAS016002: Processing weld deployment savemesafeEJB.jar
...
13:13:43,998 INFO [javax.enterprise.resource.webcontainer.jsf.config] (MSC service thread 1-10) Inicializando Mojarra 2.2.8-jbossorg-1 20140822-1131 para el contexto '/savemesafe'
13:13:44,589 INFO [org.primefaces.webapp.PostConstructApplicationEventListener] (MSC service thread 1-10) Running on PrimeFaces 5.2
13:13:44,599 INFO [org.wildfly.extension.undertow] (MSC service thread 1-10) JBAS017534: Registered web context: /savemesafe
13:13:44,609 INFO [org.jboss.as.server] (ServerService Thread Pool -- 28) JBAS018559: Deployed "savemesafe.ear" (runtime-name : "savemesafe.ear")
```

Una vez arrancado podemos acceder a la interfaz de la aplicación a través de la url <http://localhost:8080/savemesafe> que nos llevará directamente a la página de login.



8 Entorno de desarrollo con Eclipse

Con la utilidad Ant hemos podido compilar y empaquetar la aplicación pero sería necesario utilizar un IDE si lo que pretendemos es depurar o probar el código de la aplicación. Para ello podemos hacer uso del entorno de desarrollo Eclipse.

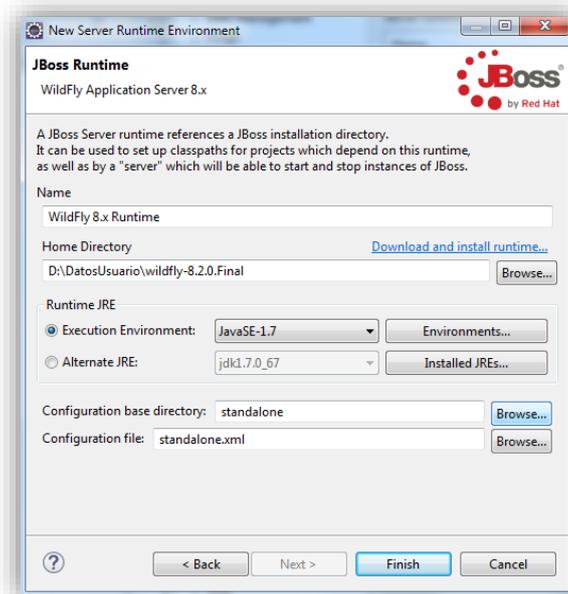
En el apartado de requisitos software del presente documento se encuentra el enlace directo a la descarga de este producto con la versión requerida.

La instalación de Eclipse consiste básicamente en descargar un fichero zip y descomprimirlo, por supuesto, teniendo disponible previamente la máquina virtual Java en el sistema. Para mayor información sobre la instalación de este IDE puede consultar el siguiente enlace: https://wiki.eclipse.org/Eclipse/Installation#Eclipse_4.4_.28Luna.29

El fichero zip que contiene el proyecto de la aplicación puede ser importado como proyecto Java en Eclipse pero será necesario realizar algunas tareas adicionales para la integración de JBoss y Eclipse. La tarea principal consiste en la instalación del plugin JBoss Tools, en su versión 4.2.3. Su instalación en Eclipse queda documentada en el siguiente enlace: <http://tools.jboss.org/downloads/jbosstools/luna/4.2.3.Final.html>.

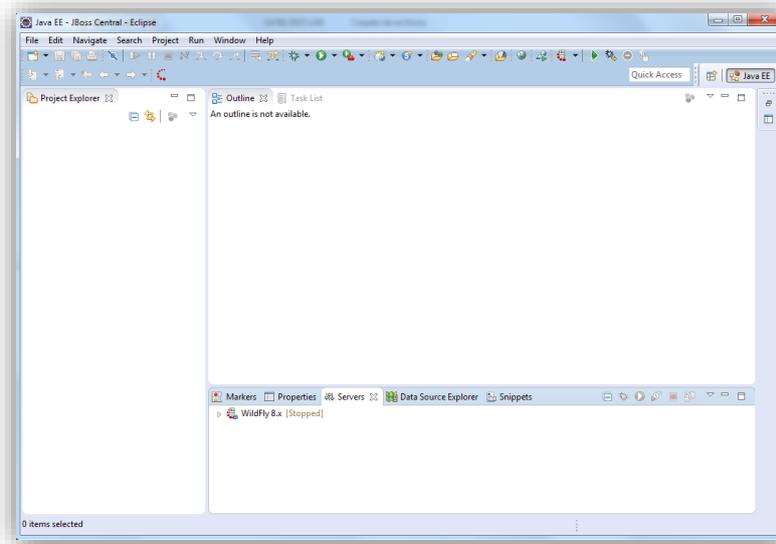
Una vez tenemos instalado eclipse y el plugin JBoss Tools podemos hacer uso del servidor JBoss referenciando una instalación ya existente o creando una nueva instancia del servidor para eclipse.

En la pestaña Server que podemos encontrar en la parte inferior de Eclipse tenemos la opción de añadir un nuevo servidor. Elegimos JBoss Community -> WildFly 8.X Runtime y aparecerá la ventana siguiente:

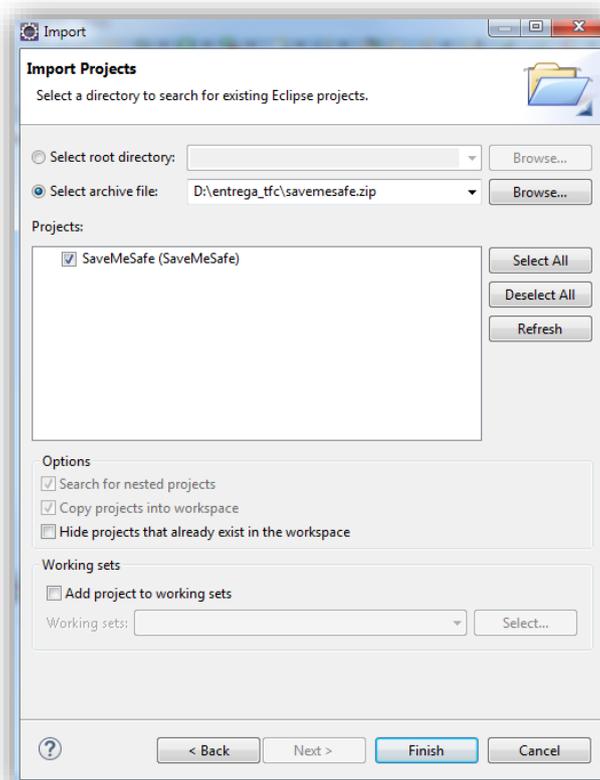


Como puede apreciarse, tenemos la opción de hacer uso de una instalación de JBoss con la que contamos o podemos descargar un nuevo servidor desde la opción Download and install runtime. En nuestro caso vamos a hacer uso de la instalación y configuración de JBoss que se ha realizado a lo largo de este documento, para lo que en Home Directory debemos indicar la ruta de la instalación y hacer clic en finalizar.

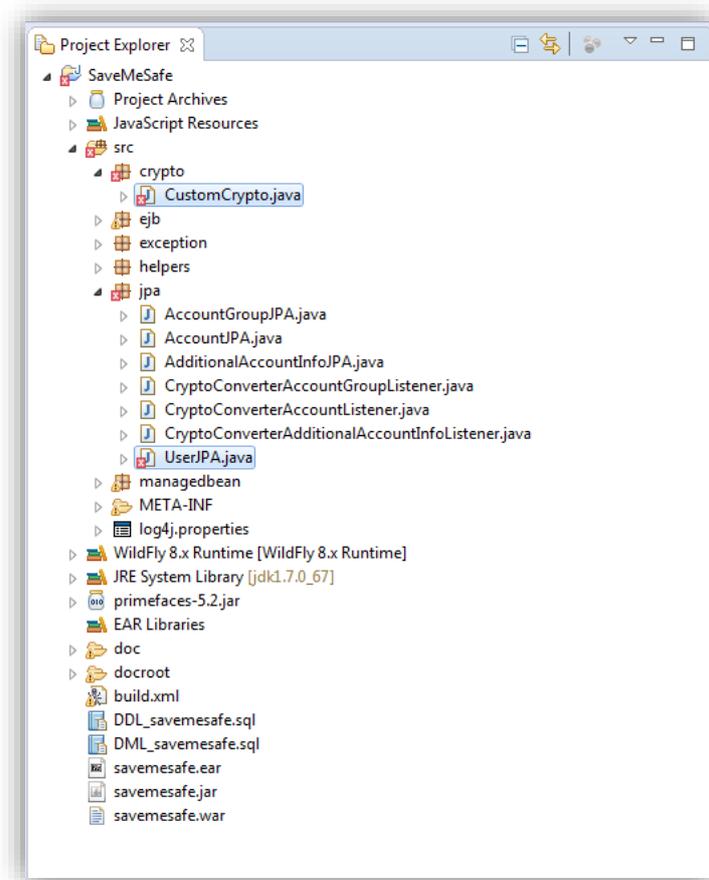
Tras la creación del servidor nos aparecerá en la pestaña Server:



Ahora ya podemos importar el proyecto de la aplicación. Para ello vamos a File -> Import -> General -> Existing Projects into Workspace y se mostrará la siguiente ventana donde debemos seleccionar el fichero savemesafe.ear:

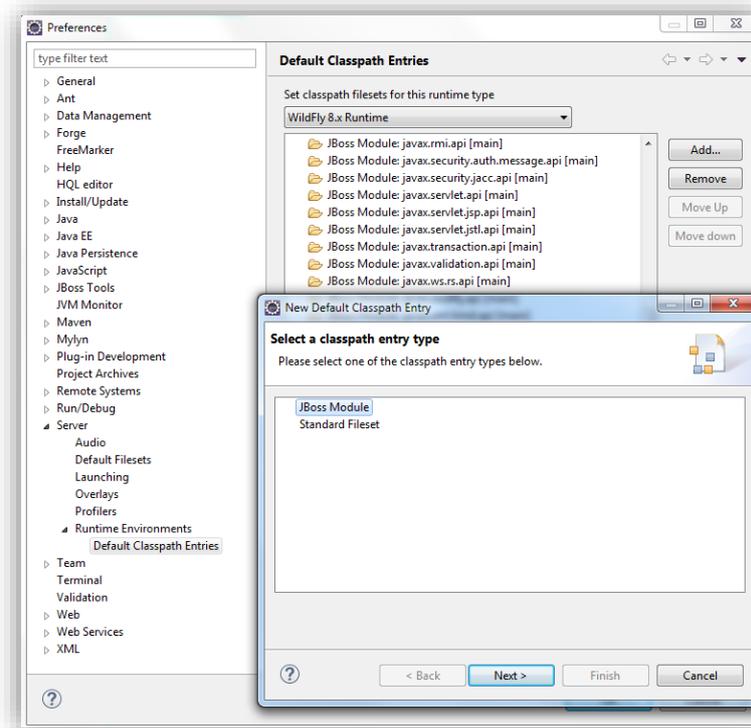


Tras importar el proyecto veremos la siguiente ventana, donde vemos dos errores relacionados con el `Build Path` del proyecto.



Este problema se debe a que por algunas particularidades de la implementación estamos haciendo uso de dos librerías, `hibernate-core-4.3.7.Final.jar` para hacer uso de la anotación `@Fetch` y `commons-codec-1.9.jar` para hacer uso de funciones criptográficas, que no están definidas en el `Build Path` del proyecto. Estas dos librerías están incluidas como módulos en JBoss por lo que tan solo debemos incluirlas como parte del runtime del servidor en el proyecto.

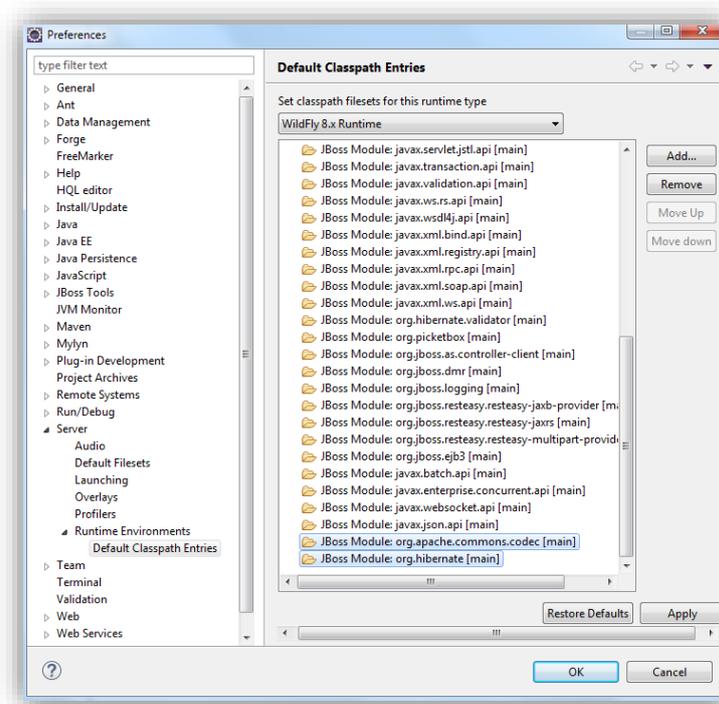
Para incluir estas dos librerías como parte del runtime JBoss vamos a `Windows -> Preferences -> Server -> Runtime Environments -> Default Classpath Entries`. En el combo de la derecha elegimos `WildFly 8.x Runtime`, hacemos clic en `añadir` y en la nueva ventana elegimos `JBoss Module`.



Los módulos a añadir son (indicándolos en el campo Module ID y utilizando el slot Main):

- org.apache.commons.codec
- org.hibernate

Una vez añadidos podemos comprobarlo en el listado de módulos cargados en el runtime del servidor y comprobamos que el proyecto compila perfectamente.



Para lanzar la construcción del EAR y desplegarlo en JBoss podemos hacer uso de la utilidad Ant integrada en Eclipse haciendo clic con el botón derecho sobre el fichero `build.xml` y eligiendo la opción Run AS -> Ant Build.

Para arrancar JBoss desde Eclipse podemos hacer clic con el botón derecho sobre el servidor que se muestra en la pestaña Server y elegir la opción Start.