**Manual de Instalación**

**Kiosko V2 – Aplicación Web**

Tabla de Contenidos

[1. Control del documento 4](#_Toc428204906)

[1.1. Versiones 4](#_Toc428204907)

[1.2. Aprobaciones 4](#_Toc428204908)

[2. Pre-requisitos 5](#_Toc428204909)

[3. Componentes a instalar: 5](#_Toc428204910)

[4. Base de Datos 6](#_Toc428204911)

[4.1. Abrir una terminal de comandos del sistema operativo 6](#_Toc428204912)

[4.2. Abrir la Herramienta de Entorno de Administración de DB2. 8](#_Toc428204913)

[4.3. Crear la Base de Datos 9](#_Toc428204914)

[4.3.1. Especificar nombre para la base de datos (Specify a name for your new database): 10](#_Toc428204915)

[4.3.2. Especificar donde almacenar los datos (Specify where to store your data): 11](#_Toc428204916)

[4.3.3. Especificar la localización de la base de datos (Specify the locale for this database): 12](#_Toc428204917)

[4.3.4. Revisar las acciones que se realizarán cuando se da click en finalizar (Review the actions that will take place when you click Finish): 13](#_Toc428204918)

[4.4. Desplegar Scripts De base de datos. 14](#_Toc428204919)

[5. Creación de Carpeta para Carga de Archivos 19](#_Toc428204920)

[6. Configuración de Java Policy en la JVM 23](#_Toc428204921)

[7. Configuración del WAS 26](#_Toc428204922)

[7.1. Variables de Entorno 27](#_Toc428204923)

[7.2. JNDI 29](#_Toc428204924)

[7.2.1. Crear Proveedor JDBC. 29](#_Toc428204925)

[7.2.2. Crear Origen de datos. 34](#_Toc428204926)

[7.2.3. Crear JAAS – Datos de autenticación J2C. 38](#_Toc428204927)

[7.3. Notificaciones Push para Android 42](#_Toc428204928)

[8. Servicios Web 45](#_Toc428204929)

[8.1. Configuración de propiedades del servicio web. 45](#_Toc428204930)

[8.1.1. Configuración en tabla de base de datos. 45](#_Toc428204931)

[8.1.2. Configuración en archivo de propiedades. 45](#_Toc428204932)

[8.2. Configuración de propiedades de la librería corporativa. 49](#_Toc428204933)

[8.3. Instalación del servicio web. 54](#_Toc428204934)

[8.3.1. Despliegue del aplicativo en Websphere. 54](#_Toc428204935)

[8.3.2. Creación de ruta simbólica en Linux. 59](#_Toc428204936)

[8.4. Desinstalación del servicio web. 61](#_Toc428204937)

[8.4.1. Eliminación de ruta simbólica en Linux. 61](#_Toc428204938)

[8.4.2. Remover el despliegue del Websphere. 62](#_Toc428204939)

[9. Aplicación Web 65](#_Toc428204940)

[9.1. Configuración de propiedades del Sitio Web. 65](#_Toc428204941)

[9.2. Instalación del sitio web. 66](#_Toc428204942)

[9.3. Desinstalación del sitio web. 71](#_Toc428204943)

# Control del documento

## Versiones

| Número de versión | Fecha de elaboración | Autor | Resumen de cambios |
| --- | --- | --- | --- |
| 1.0 | 06/08/2015 | Lennin Dávila | Secciones 1 a 4. |
|  |  |  |  |

## Aprobaciones

| Nombre | Rol |
| --- | --- |
|  | Arquitecto |
|  | Administrador de la base de datos |

# Pre-requisitos

El servidor sobre el cual se realizará el despliegue de la aplicación debe de tener instalado el siguiente software base, el cual se asume como válido:

* SUSE Linux Enterprise Server 11 SP2
* DB2 9.7.0.5
* WAS 8.0.0.3
* JAVA IBM J9 VM (build 2.6, JRE 1.6.0 Linux amd64-64 20120308\_104898)

# Componentes a instalar:

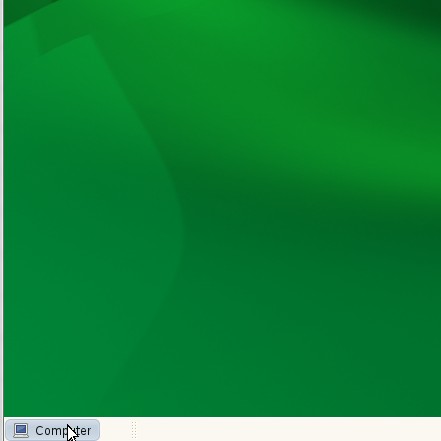
La aplicación Kiosko 2.0 requiere de la instalación de los siguientes componentes como parte de su back-end:

* Base de datos
* Servicios Web
* Aplicación Web

# Base de Datos

# Abrir una terminal de comandos del sistema operativo

Para abrir una terminal de comandos, ir al botón **“Computer”** en el escritorio del sistema Operativo Suse Linux.

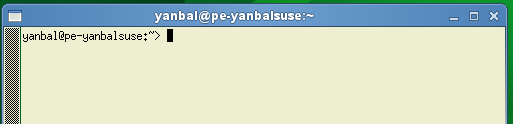


Al hacer click en ese botón, se desplegara las siguientes opciones de menú que se puede ver en la figura siguiente, seleccionar y hacer clic en **“XTerm”:**

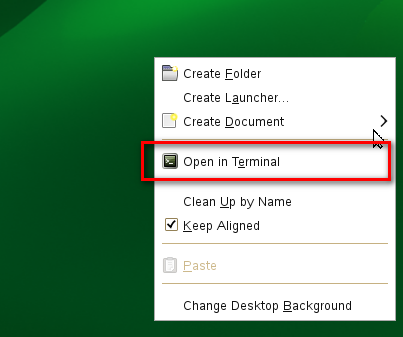


Es decir, para abrir una ventana de línea de comandos se debe ir a las siguientes opciones desde el escritorio del sistema operativo: **Computer** 🡪 **XTerm**

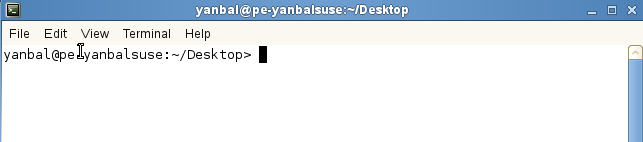
Luego, se mostrará la siguiente pantalla de comandos.



Otra manera de abrir una terminal de comandos es hacer click derecho sobre cualquier espacio de trabajo (Escritorio, Carpeta) de sistema operativo y seleccionar el menú emergente “**Open in Terminal**” como se muestra.



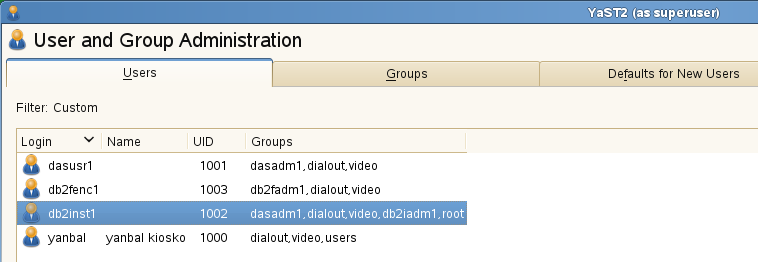
Nos mostrara la siguiente terminal de comandos:



Es decir, otra manera de abrir una ventana de línea de comandos se debe ir a las siguientes opciones desde el escritorio del sistema operativo: “**click derecho 🡪 Open in Terminal”**

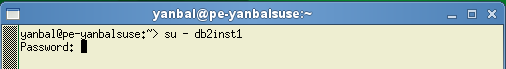
# Abrir la Herramienta de Entorno de Administración de DB2.

Es recomendable asignar privilegios root al usuario de la instancia de base de datos, en la pantalla siguiente podemos visualizar que el usuario **db2inst** se encuentra dentro del grupo **“root”:**

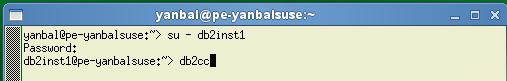


Para abrir la Herramienta de Entorno de Administración del DB2 se debe realizar los pasos siguientes:

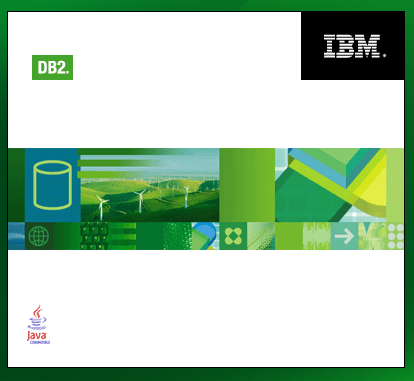
* En la terminal de comandos que se abrió, en el paso anterior escribir “**su – db2inst1”,** luego escribir la contraseña de la cuenta y presionar Enter:



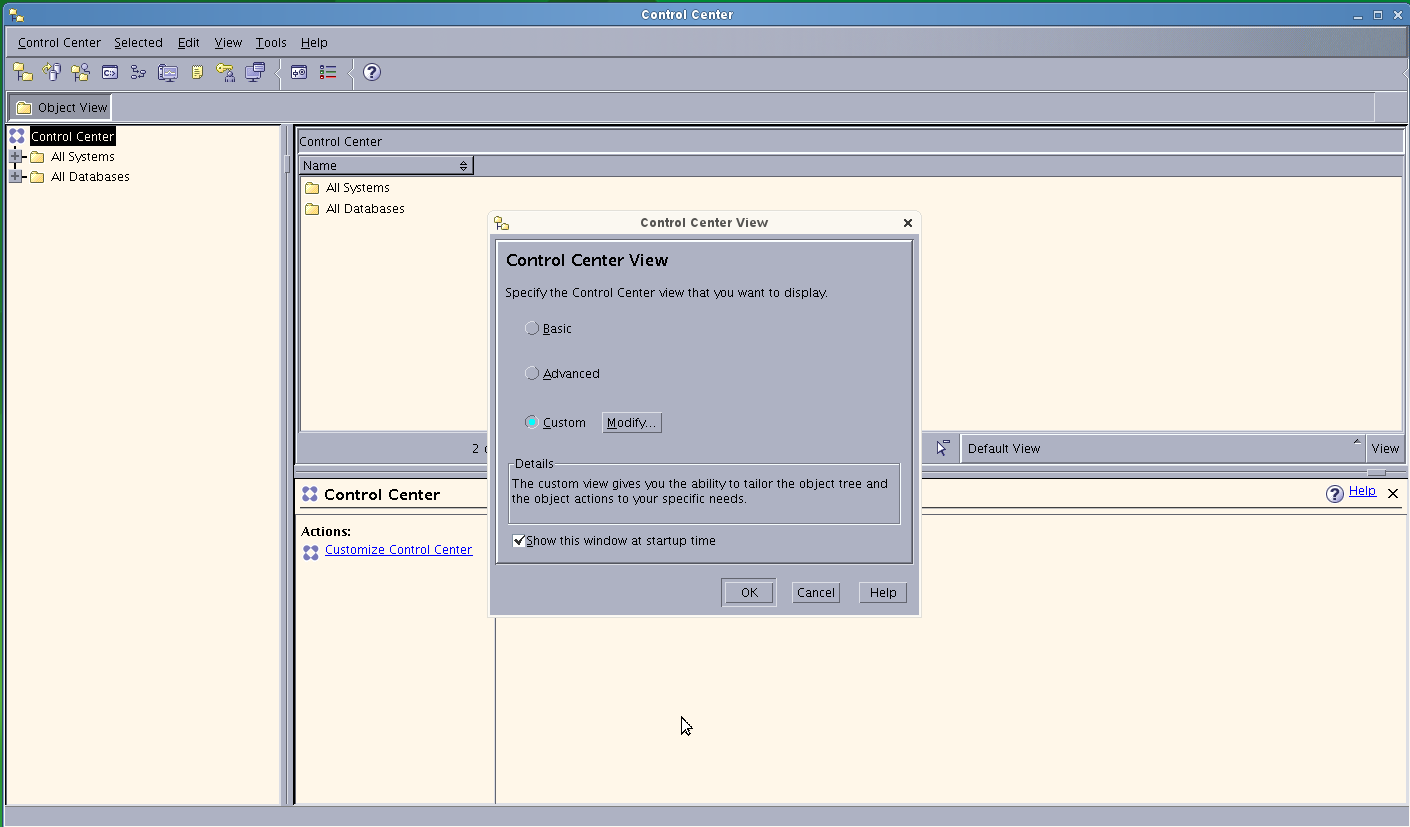
* Escribir el comando “db2cc” para iniciar el Entorno de Administración de DB2, y presionar Enter:



* Se mostrara la pantalla siguiente cuando inicie la herramienta de Administración del DB2.



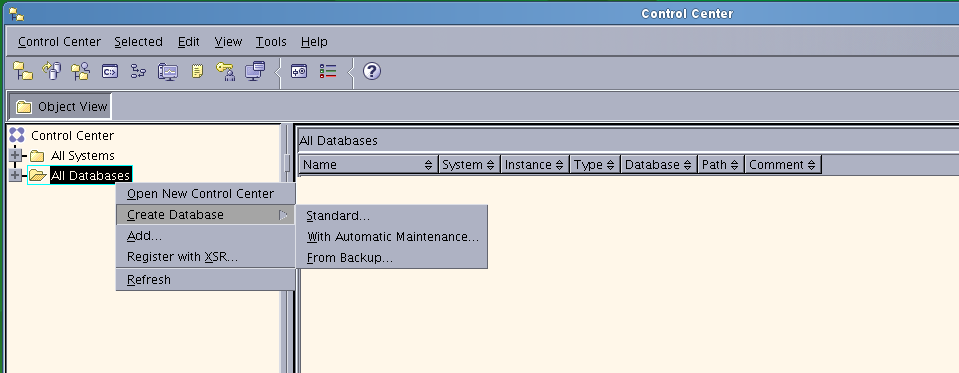
* Luego mostrara la siguiente pantalla de la Herramienta de Entorno de Administración de DB2:



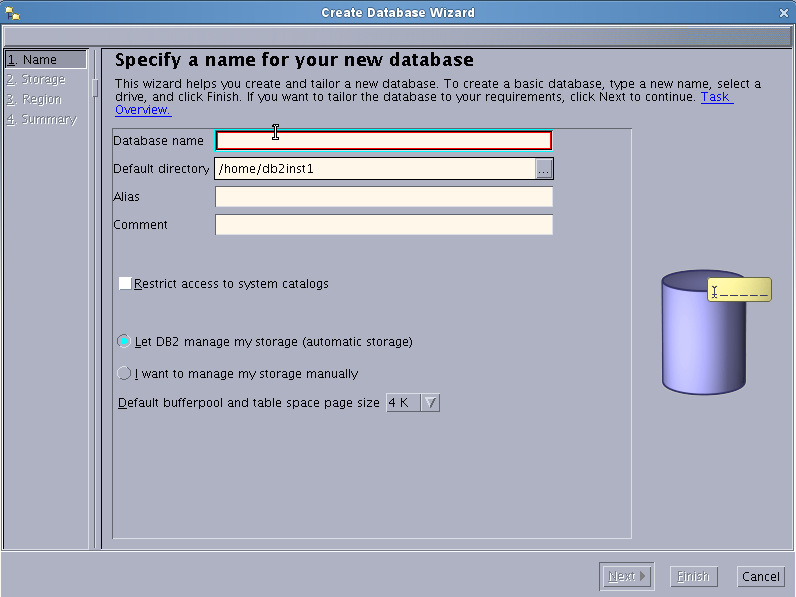
# Crear la Base de Datos

Para crear la base de datos del sistema se deben realizar los pasos siguientes:

* Seleccionar el ítem **“All Databases”** 🡪 **“Create Database”** 🡪 **“Standard”,** como se muestra en la figura siguiente:



* Se mostrará la pantalla del Asistente de creación de Base de datos.



* La creación de la base de datos consiste en cuatro pasos, para llevar a cabo estos pasos exitosamente se debe seguir el asistente de creación de base de datos:

# Especificar nombre para la base de datos (Specify a name for your new database):

Se mostrará la pantalla para especificar algunos atributos de la base de datos:

Database name: **KIOSKO**

Default directory: lo dejamos con la ruta por defecto, en el caso de ejemplo **“/home/db2inst1”**

Alias: (Lo dejamos en blanco)

Comment: (Lo dejamos en blanco)

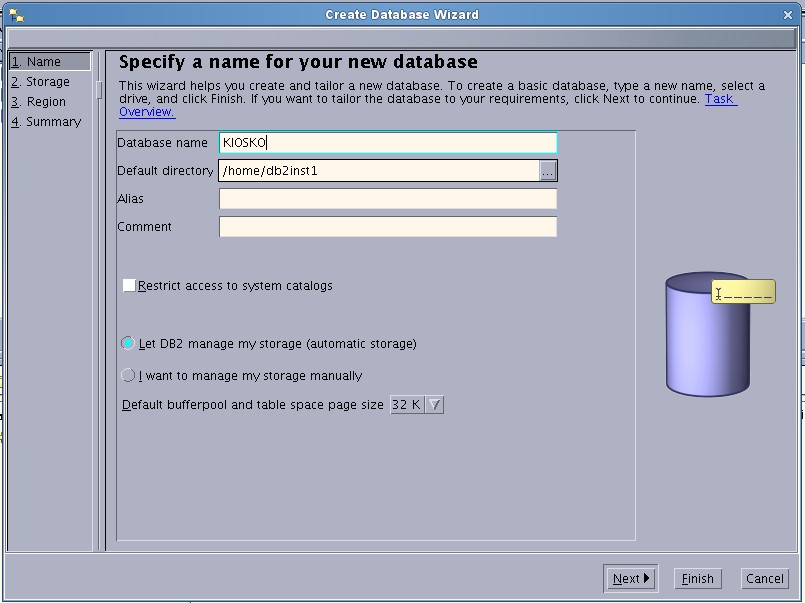
Dejamos desmarcada la opción “Restrict Access to system catalogs”

Seleccionamos la opción “Let DB2 manage my storage (automatic storage)”

Y Asignamos el valor de “32K” al campo “Default bufferpool and table space page size”

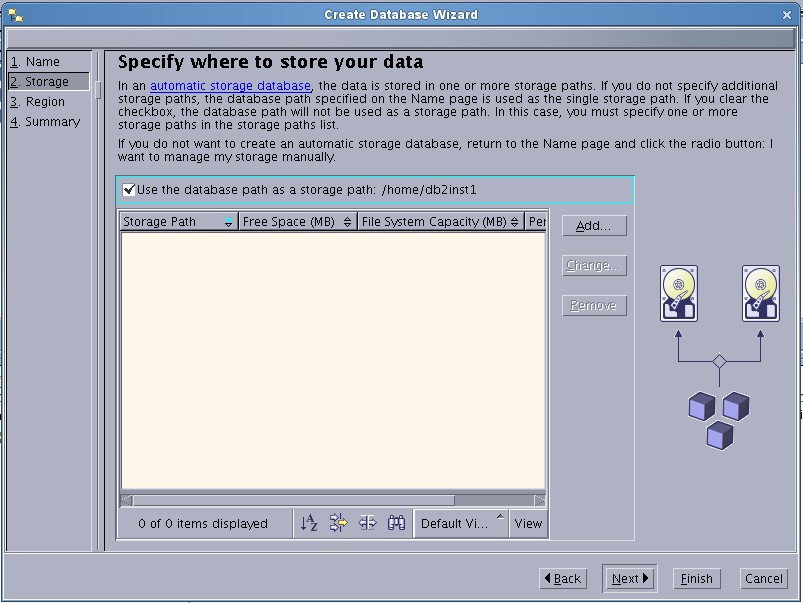


La pantalla debe quedar del modo siguiente, presionar clic en el botón “**Next**”



# Especificar donde almacenar los datos (Specify where to store your data):

En este paso dejamos las opciones por defecto que nos muestre el asistente, tal como se aprecia en la pantalla siguiente, presionamos en el botón **“Next”:**



# Especificar la localización de la base de datos (Specify the locale for this database):

En este paso elegimos la configuración de la localización de la Base de Datos. Los campos de este formulario lo seleccionamos con los siguientes valores:

Country/Region: **Peru**



Territory: Lat



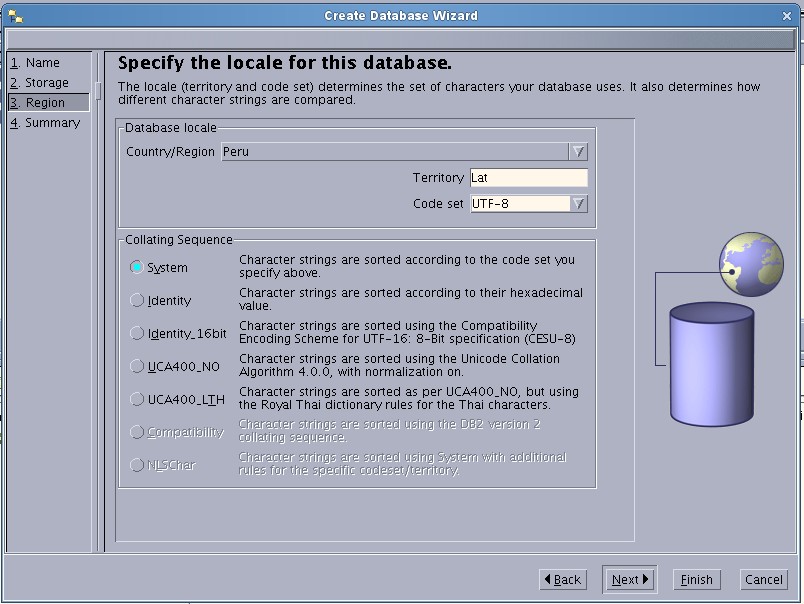
Code set: UTF-8



En la sección **Collating sequence** seleccionamos la opción **“System”** como se muestra a continuación:

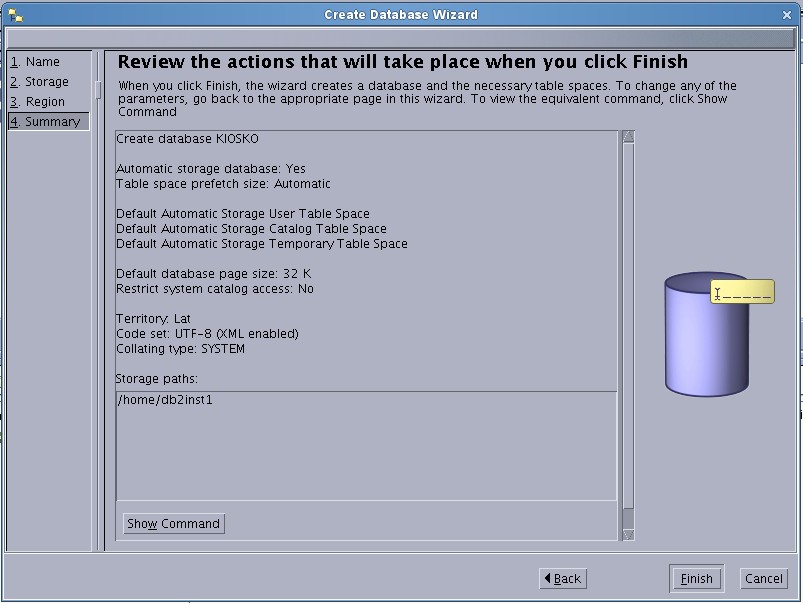


La configuración de esta sección debe quedar como se muestra en la pantalla siguiente, presionamos click en **“Next”:**

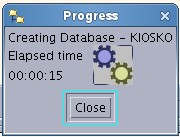


# Revisar las acciones que se realizarán cuando se da click en finalizar (Review the actions that will take place when you click Finish):

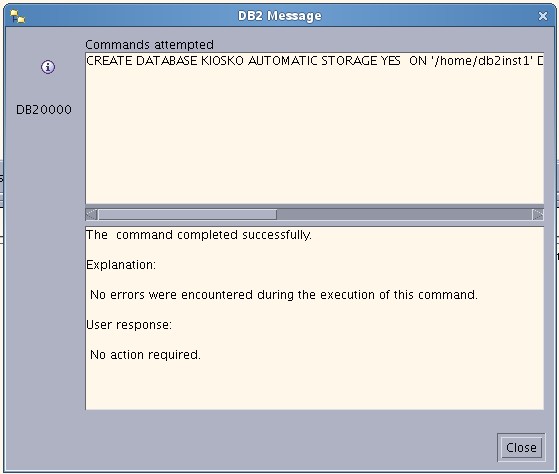
En este paso nos muestra un resumen de todas las opciones de configuración que hemos seleccionado para nuestra base de datos como se muestra en la pantalla siguiente, presionamos el botón **“Finish”** para proceder a la creación de la base de datos.



Mientras el Sistema crea la base de datos se mostrara la siguiente pantalla, este proceso. Tomará algunos minutos.



Cuando el proceso de creación de base de datos termine, se mostrara la siguiente pantalla con el resultado del proceso, presionamos el botón **“Close”** para cerrar esta ventana.



# Desplegar Scripts De base de datos.

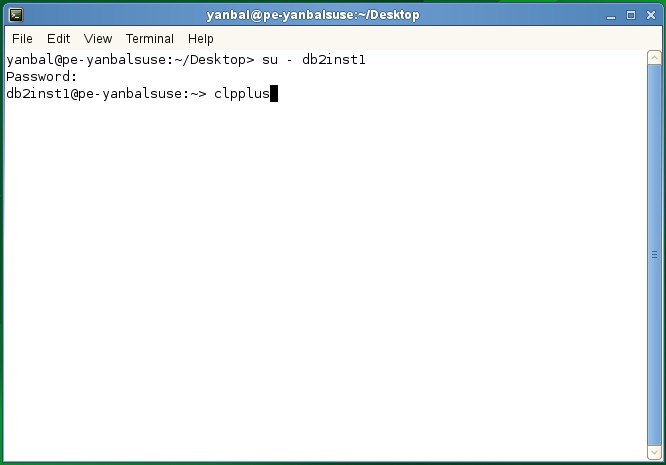
Para realizar el despliegue de los scripts se utilizará la herramienta CLPPlus de DB2.

Para ejecutar la herramienta CLPPLUS, en la terminal de comandos, se debe iniciar sesión con el usuario **db2inst1** (Omitir este paso si ya estamos con ese usuario).

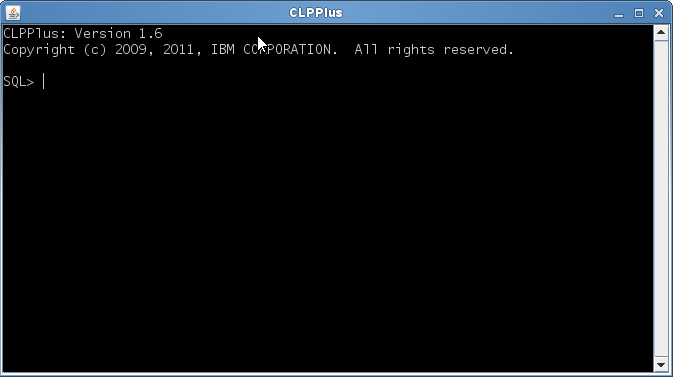
Escribimos los comandos:

su – db2inst1 y luego escribir la contraseña de la cuenta **“db2inst1”**

Luego, escribimos el comando **“clpplus”** y presionamos Enter, como se muestra en la imagen siguiente:

****

Este comando nos mostrará la ventana siguiente para ejecutar comandos SQL en DB2:



Para conectarnos a la base de datos creada escribimos el comando **“connect”** y presionamos Enter.



Ingresamos el **nombre de la base de datos** y presionamos Enter.



Ingresamos el **nombre o ip del servidor** y presionamos Enter.



Ingresamos el **puerto del servidor** y presionamos Enter.



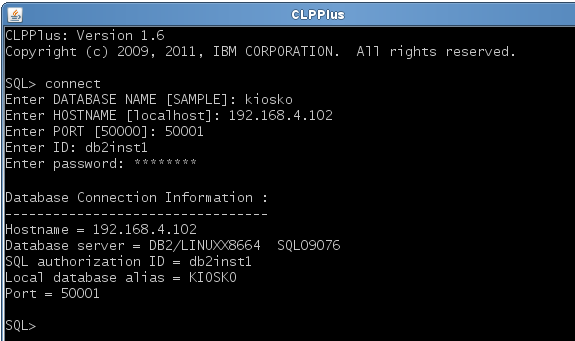
Ingresamos el **nombre de usuario** y presionamos Enter.



Ingresamos la **contraseña de usuario** y presionamos Enter.



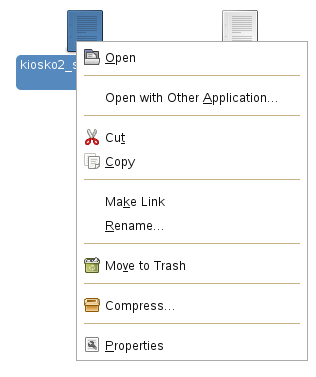
Si la conexión fue satisfactoria, se mostrará la pantalla siguiente:



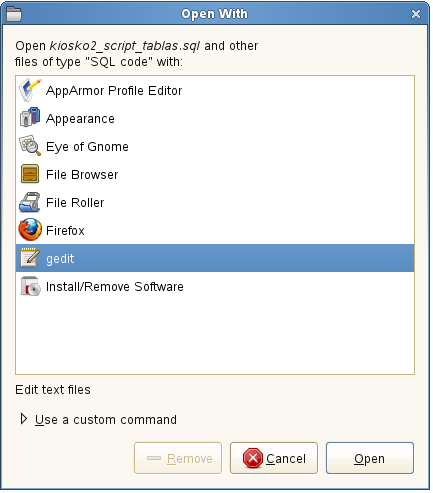
Una vez conectados a la base de datos, ejecutamos los scripts del Kiosko:

* Creación de tablas y Referencias (kiosko2\_script\_tablas.sql)
* Creación de Procedimientos y funciones (kiosko2\_script\_sp.sql).
* Carga inicial de datos (kiosko2\_script\_carga\_inicial.sql).

Para abrir los scripts con el editor de texto gpedit de Suse, seleccionamos el script y hacemos click derecho, luego seleccionamos la opción de menú **“Open with Other Application”,** como se muestra en la imagen siguiente.



Nos mostrara la siguiente ventana, en ella seleccionamos **“gedit”** y click en **“Open”.**

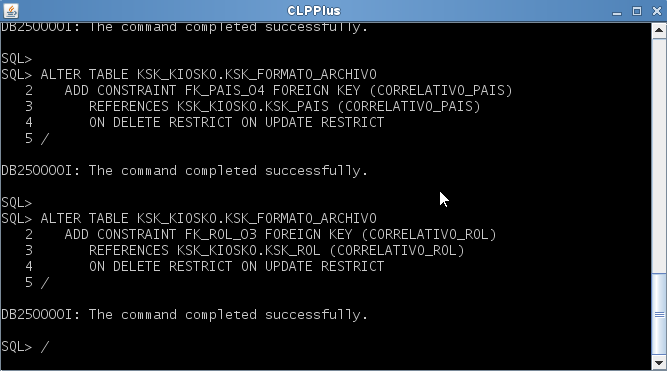


Nos mostrara el contenido del script en el editor de texto, seleccionamos todo el contenido y lo copiamos (**Ctrl + A**) (**Ctrl + C**).

Este paso se realiza con cada script de base de datos, para luego ejecutar cada uno en la línea de comandos del clpplus de db2.

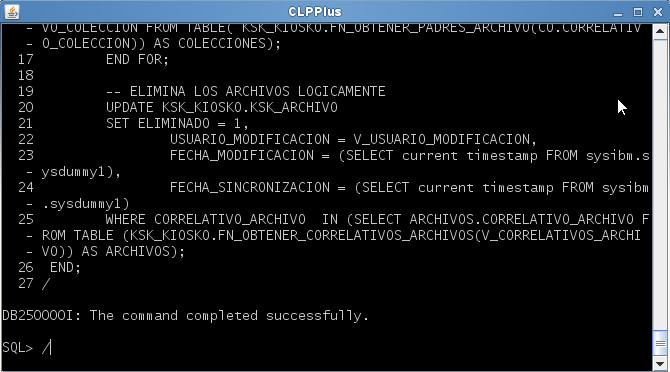
* kiosko2\_script\_tablas.sql:

Pegamos el contenido en la ventana (**clic derecho sobre la consola para pegar texto**) de comandos y esperamos mientras se ejecuta el script. Luego presionamos Enter, escribimos el comando **“clear”** para limpiar la pantalla y presionamos Enter.



* kiosko2\_script\_sp.sql:

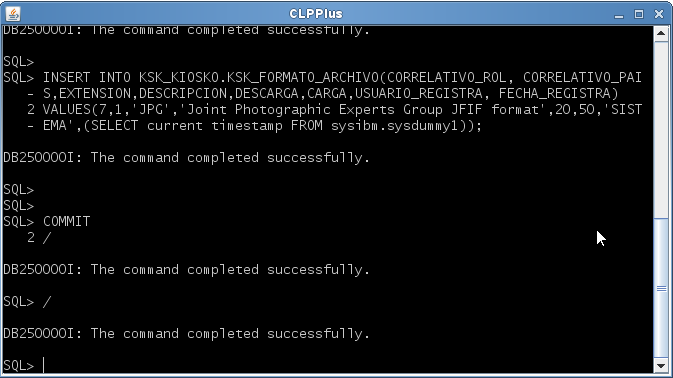
Para el script de procedimientos y funciones, realizamos lo mismo que en el caso anterior, y esperamos mientras se ejecuta el script. Luego, presionamos Enter, escribimos el comando “clear” para limpiar la pantalla y presionamos Enter.



Para el script de procedimientos y funciones ha habido ciertos casos en que es mejor compilarlos dos veces, ya que a pesar de haber sido creados, el sistema no puede acceder a el por falta de algún permiso, esto se ha solucionado solo ejecutando una vez más el procedimiento. Es por ello que recomendamos ejecutar dos veces seguidas este script.

* kiosko2\_script\_carga\_inicial.sql:

Realizamos el mismo paso para el script de carga inicial, y esperamos mientras se ejecuta el script. Luego presionamos Enter, escribimos el comando “clear” para limpiar la pantalla y presionamos Enter.



Con esto tendríamos desplegada la base de datos para el sistema.

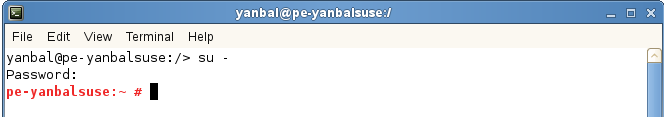
# Creación de Carpeta para Carga de Archivos

En la raíz del Sistema Operativo **“/”** crearemos nuestra estructura de directorios que nuestra aplicación utiliza, en este manual muchas configuraciones van dirigidas hacia esta estructura de carpetas que crearemos a continuación.

Visualizamos la siguiente pantalla en la raíz del disco del sistema **“/”**.



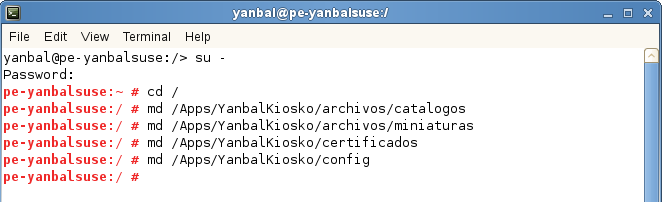
Abrimos una terminal de comandos e iniciamos sesión como un super administrador con el comando **“su -“** e ingresamos la contraseña de “**root**” y presionamos “**Enter”**.



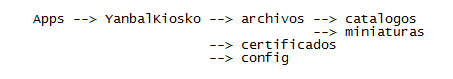
Nos posicionamos en la raíz del disco con el comando “**cd /**” y ejecutamos los siguientes comandos para crear nuestra estructura de carpetas:

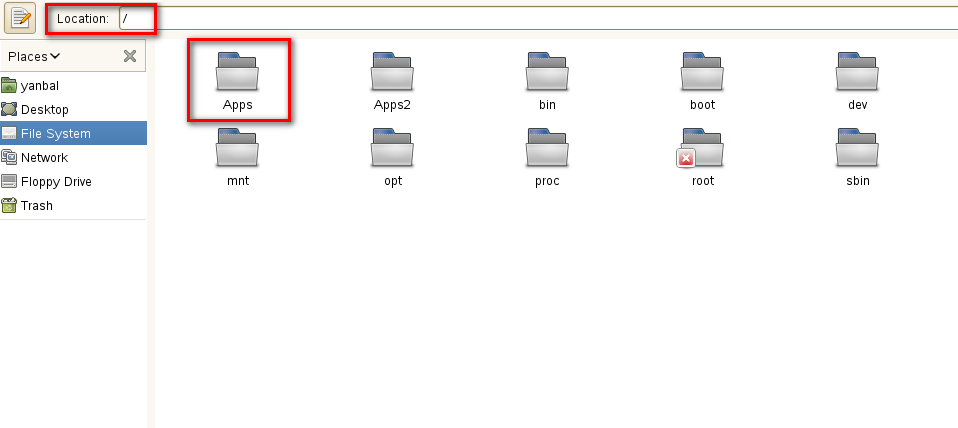
* **md /Apps/YanbalKiosko/archivos/catalogos**
* **md /Apps/YanbalKiosko/archivos/miniaturas**
* **md /Apps/YanbalKiosko/certificados**
* **md /Apps/YanbalKiosko/config**

Como se muestra en la siguiente pantalla:

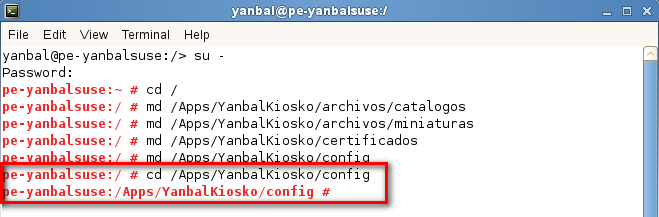


Y tendremos la siguiente estructura de carpetas:

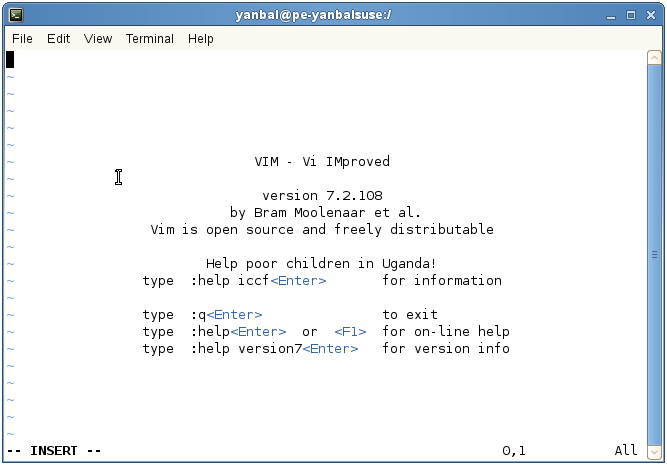




Nos movemos dentro de la ruta **“/Apps/YanbalKiosko/config”** con el comando “**cd** **/Apps/YanbalKiosko/config”** como se muestra:



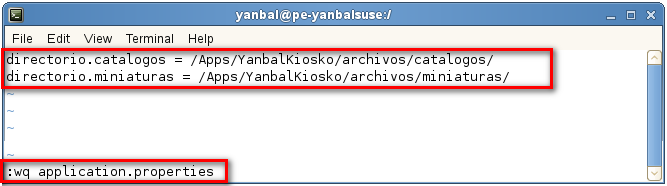
Ahora creamos el archivo “**application.properties**”, para ello ejecutamos el comando “**vi**” y apretamos Enter, luego la tecla “**insert**” y Enter.



Editamos el archivo con los valores siguientes.

* **directorio.catalogos = /Apps/YanbalKiosko/archivos/catalogos/**
* **directorio.miniaturas = /Apps/YanbalKiosko/archivos/miniaturas/**

Luego guardamos los cambios y asignamos un nombre al archivo creado con el comando “**:wq application.properties**”



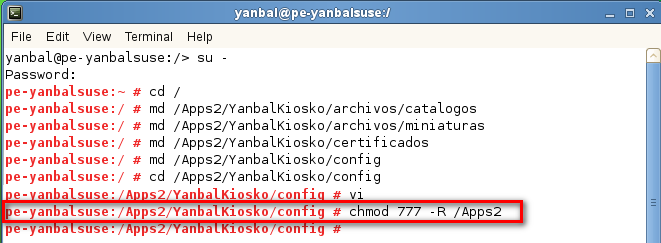
Si vamos a la ruta **“/Apps/YanbalKiosko/config”** podremos encontrar el archivo recién creado, como se muestra en la imagen:



Con el contenido siguiente:



Asignamos permisos a nuestro usuario a la estructura de carpetas que hemos creado, para ello ejecutamos el siguiente comando “**chmod 777 –R /Apps**”, con estos permisos podremos editar los archivos desde interfaz grafica o crear nuevos archivos sin necesidad de iniciar sesión como super usuario desde una terminal.



Dentro de la ruta **“/Apps/YanbalKiosko/certificados”** copiaremos los archivos necesarios para el envío de notificaciones push IOS, estos archivos son generados desde la cuenta de desarrollador IOS y son los archivos con las extensiones

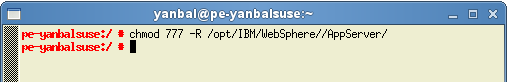
* **.p12**
* **.Cer**
* **.pem**

# Configuración de Java Policy en la JVM

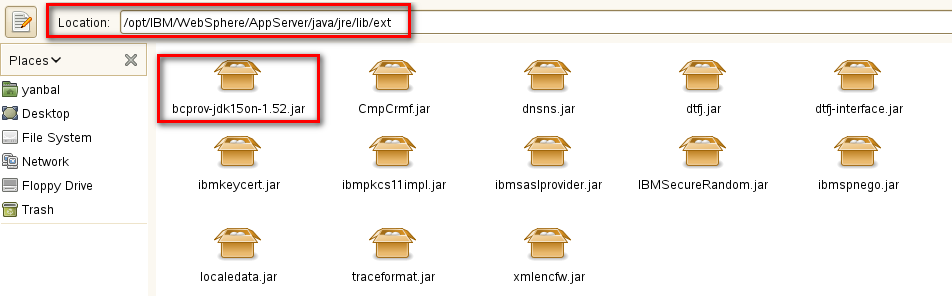
Descargar la librería Bouncy Castle (**bcprov-jdk15on-152.jar**) de la ruta:

[https://www.bouncycastle.org/latest\_releases.html](https://www.bouncycastle.org/latest_releases.html" \o "https://www.bouncycastle.org/latest_releases.html)

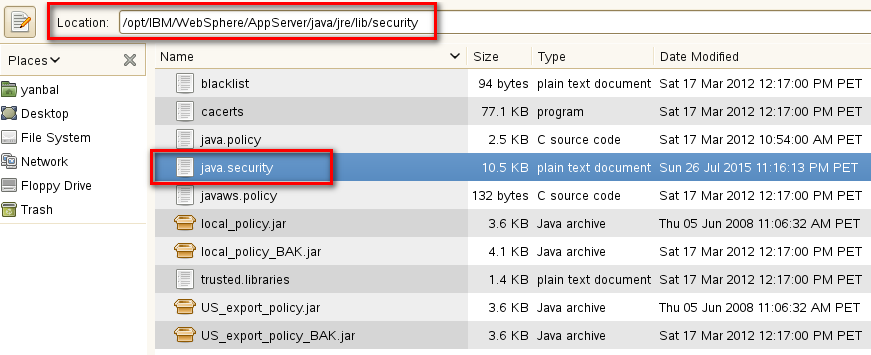
Nos asignamos permisos con una terminal de comandos con privilegios de super usuario y ejecutamos el comando chmod donde se encuentra instalado el servidor Websphere.



Ubicar el jar descargado en la ruta **%JRE\_HOME%/lib/ext**



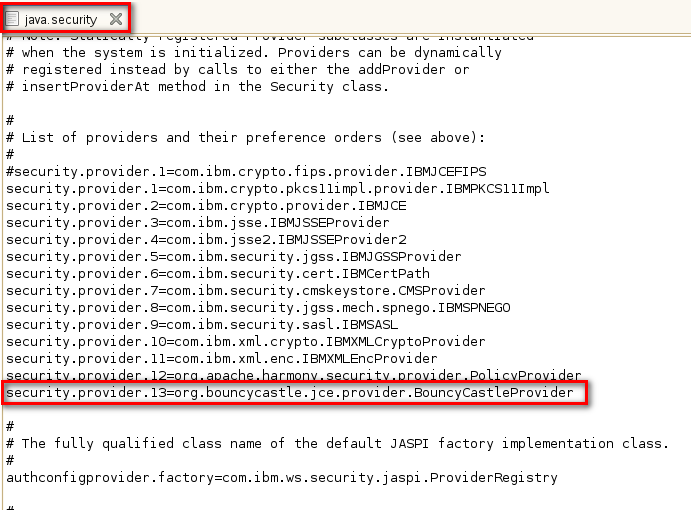
Abrir el archivo **java.security**, ubicado en la ruta **%JRE\_HOME%/lib/security**



Agregar BouncyCastleProvider a las lista de proveedores de seguridad. Para ello, escribir la siguiente línea:

**security.provider.<nro\_orden>=org.bouncycastle.jce.provider.BouncyCastleProvider**

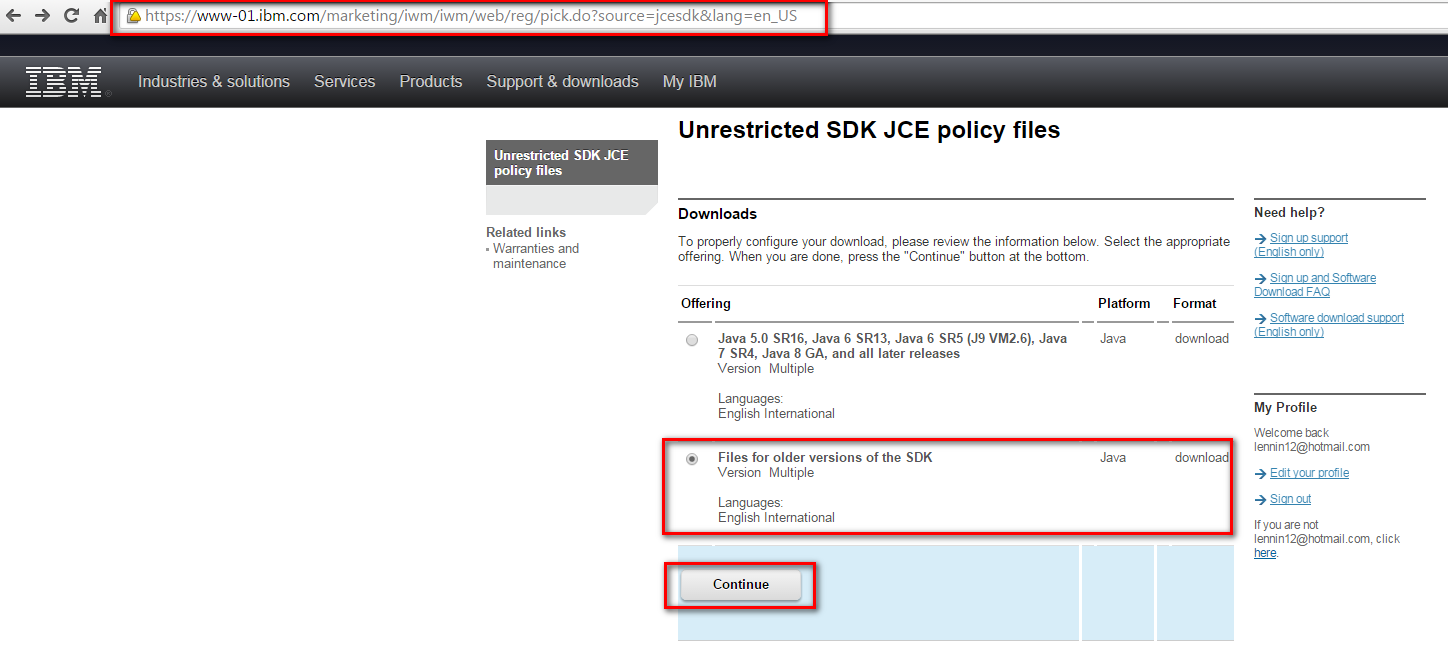
Donde **nro\_orden** deberá ser el siguiente valor ordinal de los proveedores existentes.



Descargar **Java Cryptography Extension (JCE) Unlimited Strength Jurisdiction Policy Files** de la web oficial de IBM en la siguiente ruta:

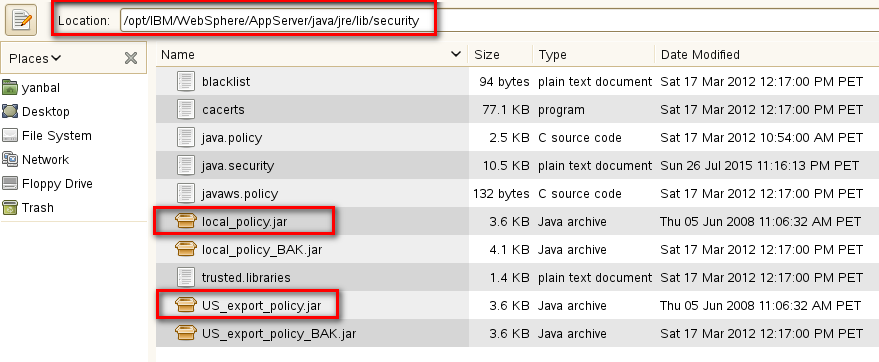
[https://www14.software.ibm.com/webapp/iwm/web/preLogin.do?source=jcesdk](https://www14.software.ibm.com/webapp/iwm/web/preLogin.do?source=jcesdk" \o "https://www14.software.ibm.com/webapp/iwm/web/preLogin.do?source=jcesdk)

Es necesario contar con una cuenta IBM (**Puede crearse una de manera gratuita**).



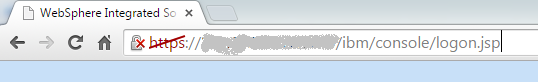
Copiar los archivos **local\_policy.jar** y **US\_export\_policy.jar** en la carpeta **%JRE\_HOME%/lib/security**.  
Se recomienda crear una copia de seguridad de los jar’s existentes en caso se desee revertir la configuración.

Luego reiniciar la Java Virtual Machine (JVM).



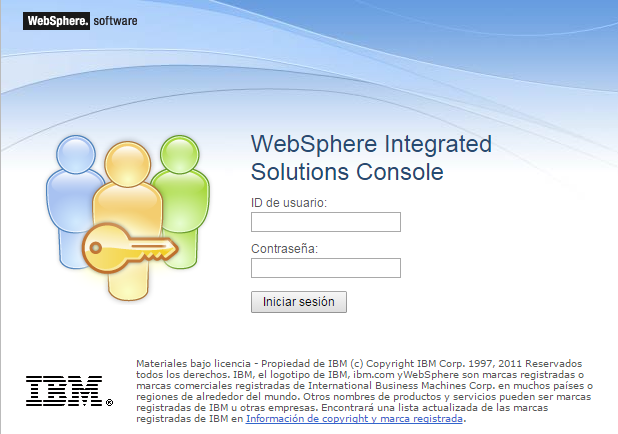
# Configuración del WAS

Para realizar las configuraciones necesarias del Websphere, ingresaremos a la consola de Administración del Websphere 8.0.0.3, ingresando el siguiente URL como se muestra.

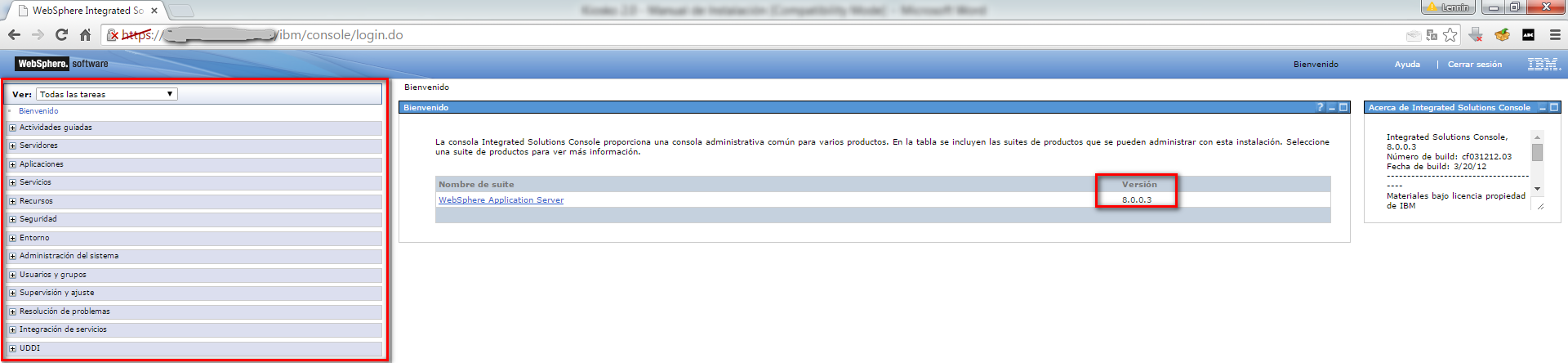


<https://XXX.XXX.X.XXX:XXXX/ibm/console/logon.jsp>

Nos mostrara la pantalla de login de la consola. Ingresamos usuario y contraseña y click en “**Iniciar sesión**”.



Luego de iniciar sesión en la consola de Administración del WebSphere, se nos mostrara la siguiente pantalla, en el lado izquierdo de la pantalla tenemos disponible las opciones de menú de configuración.



Dentro de la consola podremos realizar las configuraciones del Websphere necesarias para el correcto funcionamiento del sistema.

# Variables de Entorno

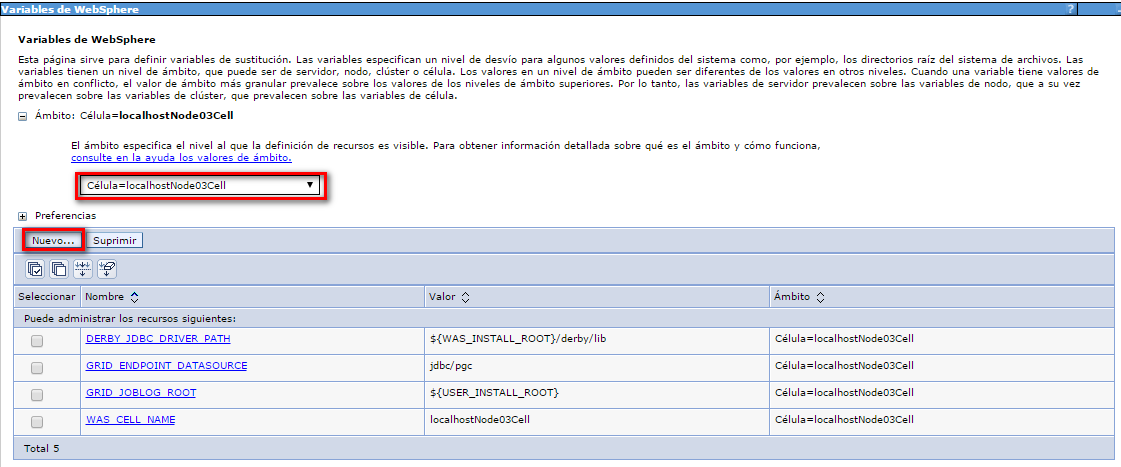
Para crear variables de entorno en Websphere, realizaremos los siguientes pasos.

Seleccionamos la opción **“Entorno” 🡪 “Variables de Websphere”** como se muestra en la imagen siguiente:

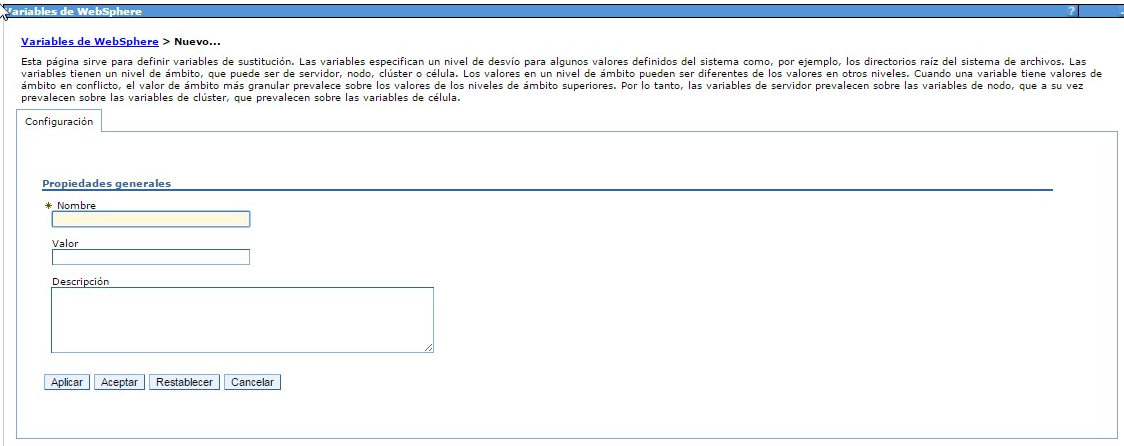


Nos mostrara la pantalla siguiente, en esta pantalla asegurarnos que el combo este en la opción **“Célula=localhostNode03Cell”** y presionar el botón “**Nuevo…**”.

El nombre de la Célula puede variar de acuerdo a la configuración de instalación del Websphere. En el caso de ejemplo tenemos el nombre **localhostNode03Cell.**



Se presentara la siguiente ventana:



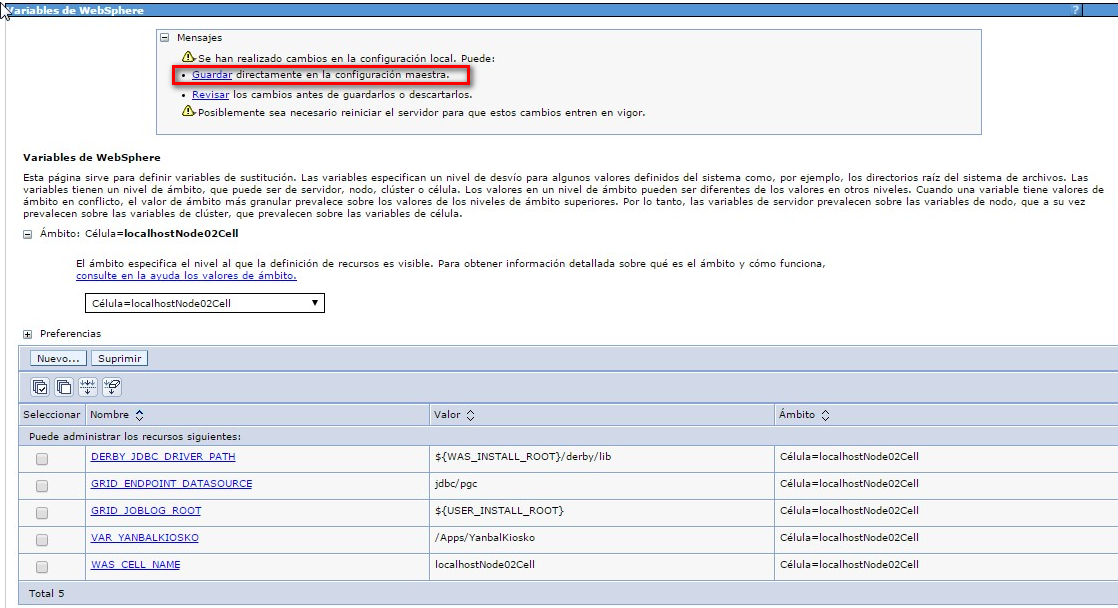
En esta ventana ingresar los valores para los campos siguientes

* Nombre: **“VAR\_YANBALKIOSKO”**
* Valor: **“/Apps/YanbalKiosko”**

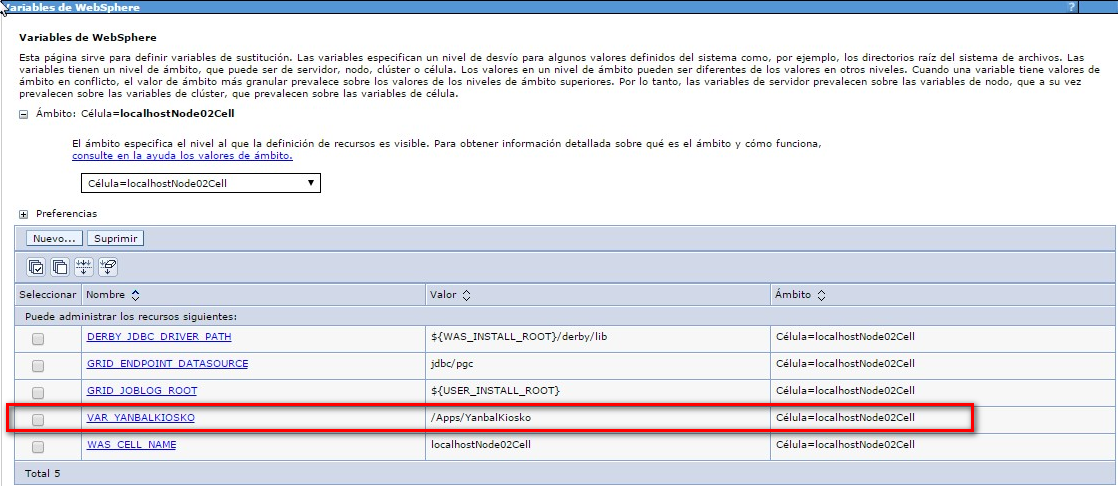
Como muestra en la pantalla siguiente, luego presionar el botón “**Aceptar**”.



Luego nos mostrara la siguiente pantalla, en esta pantalla seleccionamos la opción “**Guardar**”



Luego nos mostrara la siguiente pantalla donde podremos visualizar nuestra variable “**VAR\_YANBALKIOSKO**” recién creada.

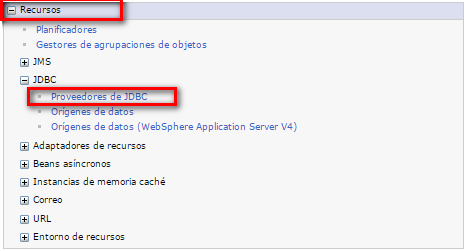


# JNDI

# Crear Proveedor JDBC.

Crear un Proveedor JDBC para el DB2, si ya se tuviera definido un proveedor para DB2, se puede omitir este paso.

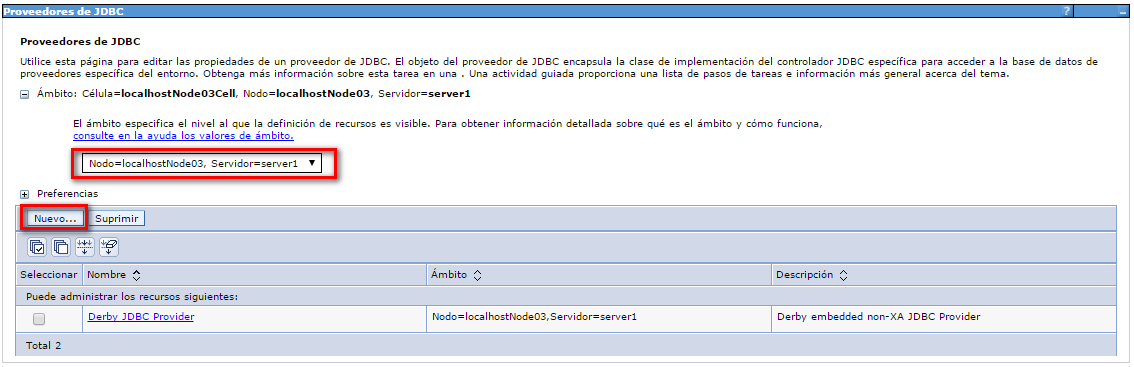
Seleccionamos la opción **“Recursos” 🡪 “JDBC” 🡪 “Proveedores de JDBC”** como se muestra en la imagen siguiente:



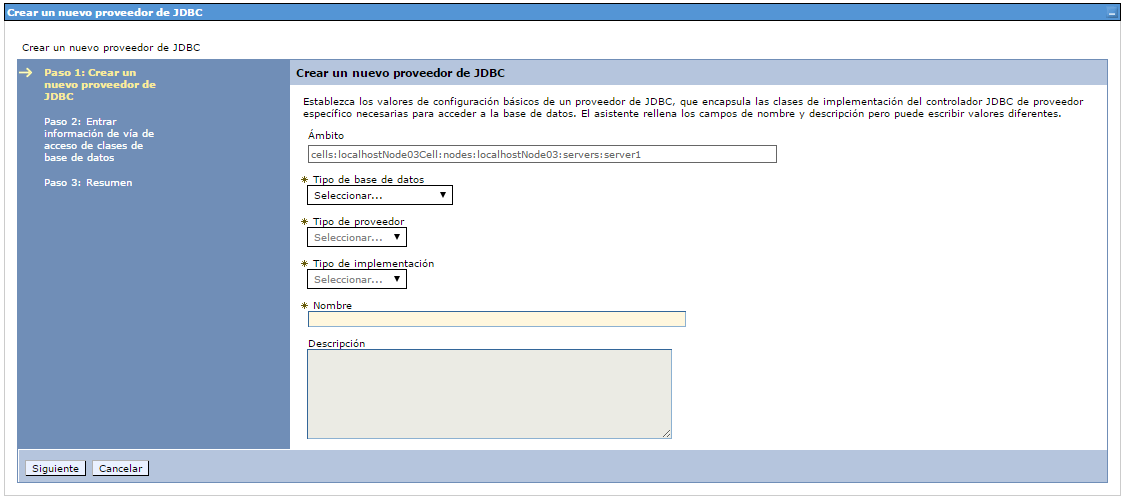
Mostrara la siguiente pantalla para crear el Proveedor JDBC.

Nos mostrara la pantalla siguiente, en esta pantalla asegurarnos que el combo este en la opción **“Nodo= localhostNode03, Servidor=server1”** y presionar el botón **“Nuevo…”.**

El nombre de la Célula puede variar de acuerdo a la configuración de instalación del Websphere. En el caso de ejemplo tenemos el nombre **Nodo= localhostNode03, Servidor=server1**



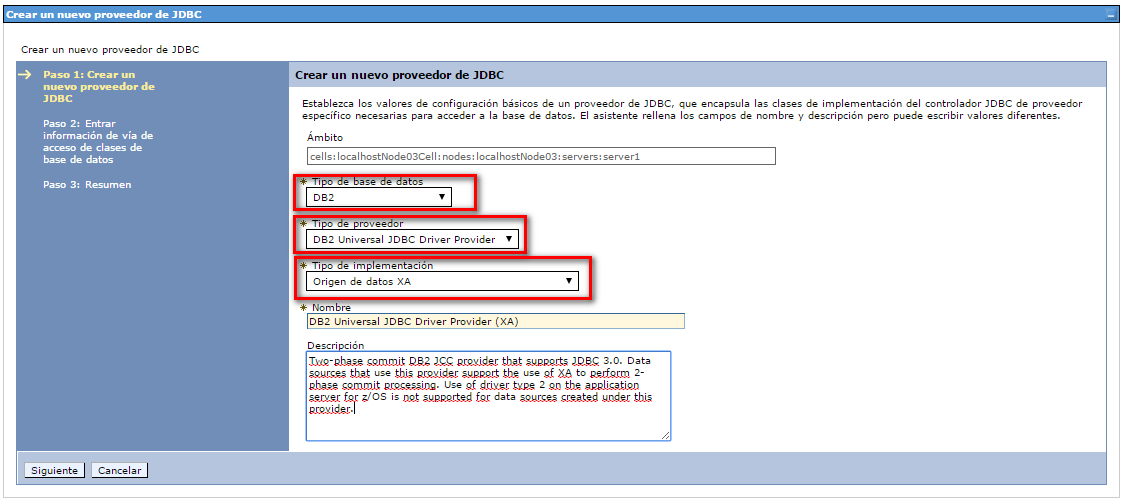
Se presentara la siguiente ventana:



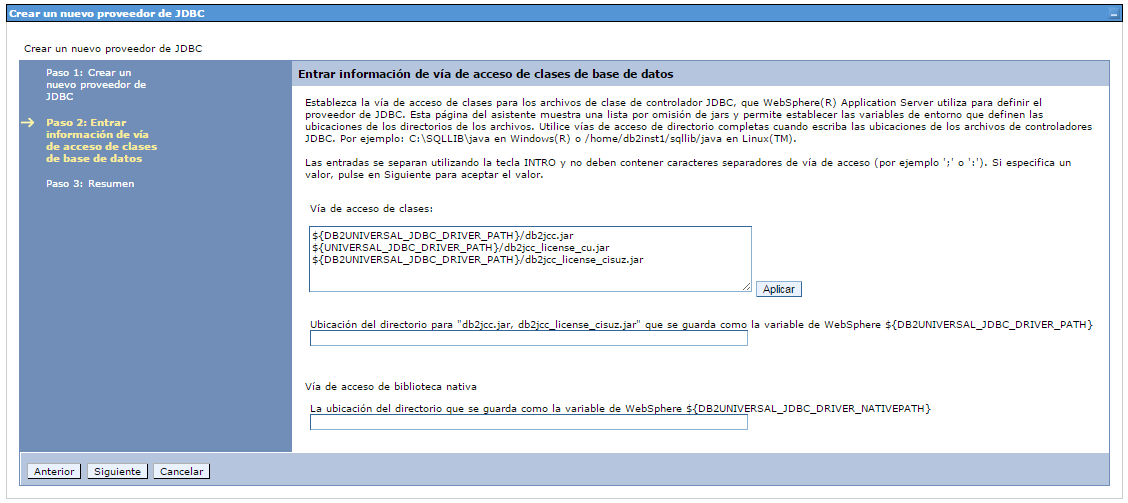
En esta ventana ingresar los valores para los campos siguientes

* Tipo de base de datos: “**DB2**”
* Tipo de proveedor: “**DB2 Universal JDBC Driver Provider**”
* Tipo de Implementación: “**Origen de datos XA**”
* Nombre: “**DB2 Universal JDBC Driver Provider (XA)**”

Como se muestra en la pantalla siguiente, luego presionar el botón “**Siguiente**”.



Se presentara la siguiente ventana:



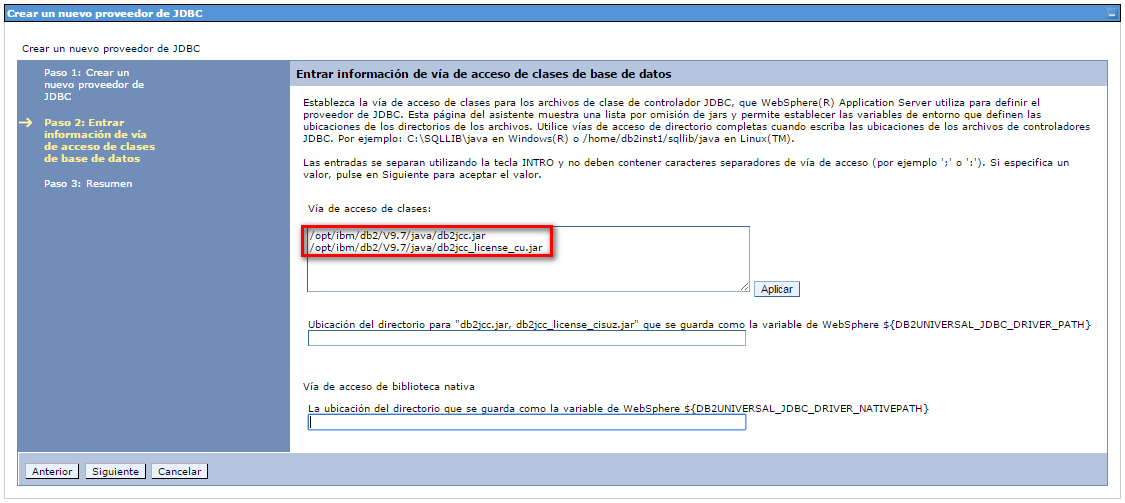
En esta ventana ingresar los valores para los campos siguientes

Vía de acceso de clases:

* **/opt/ibm/db2/v9.7/java/db2jcc.jar**
* **/opt/ibm/db2/v9.7/java/db2jcc.jardb2jcc\_license\_cu.jar**

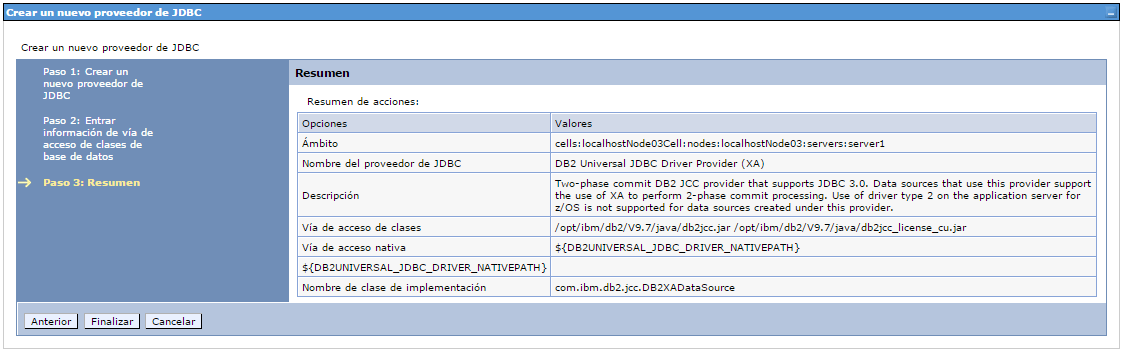
Estos jar se instalaron en nuestro sistema en la instalación del db2, si no poseemos estos jar en nuestro servidor, copiarlos desde el servidor de base de datos del db2 a una ruta similar a la mostrada en el ejemplo en el servidor de aplicaciones.

Como muestra en la pantalla siguiente, luego presionar el botón “**Siguiente**”.



Se presentara la siguiente ventana:

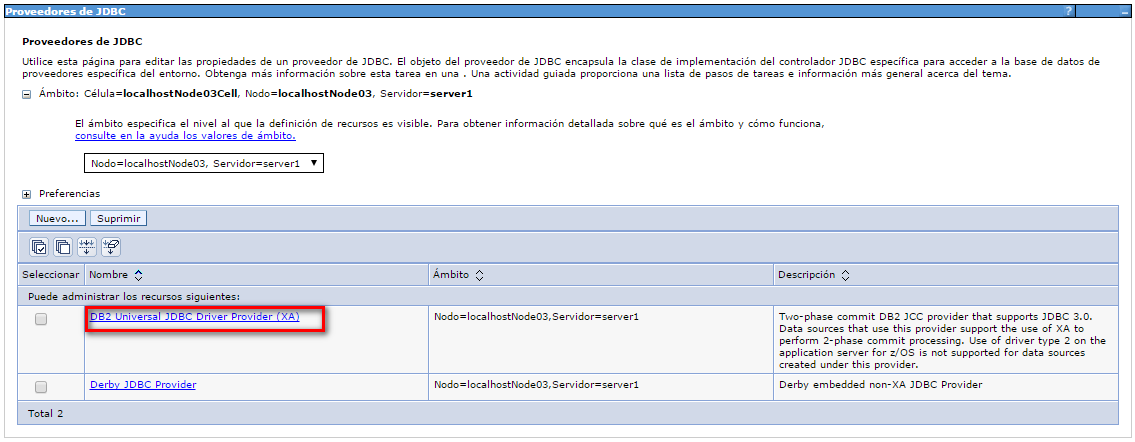
Esta ventana nos muestra un resumen de la configuración del proveedor de jdbc que estamos creando, luego presionar el botón “**Finalizar**”.



Luego nos mostrara la siguiente pantalla, en esta pantalla seleccionamos la opción “**Guardar**”

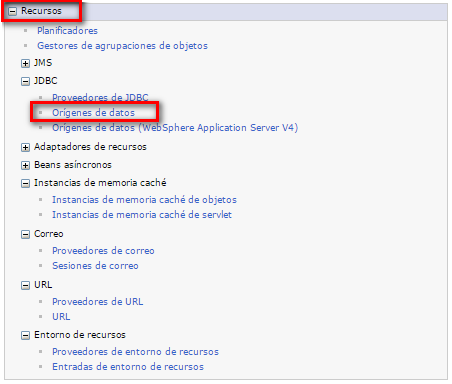


Luego nos mostrara la siguiente pantalla donde podremos visualizar nuestro proveedor JDBC **“DB2 Universal JDBC Driver Provider (XA)”** recién creado.



# Crear Origen de datos.

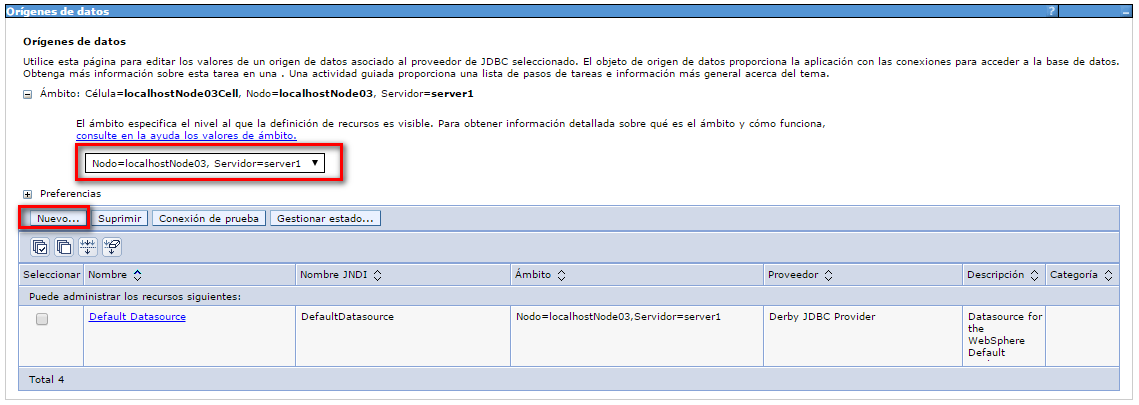
Seleccionamos la opción **“Recursos” 🡪 “JDBC” 🡪 “Orígenes de datos”** como se muestra en la imagen siguiente:



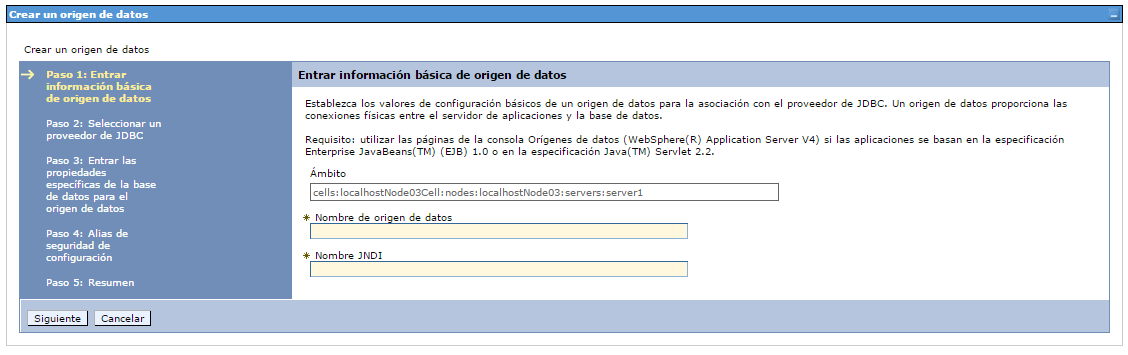
Mostrara la siguiente pantalla para crear el origen de datos.

Nos mostrara la pantalla siguiente, en esta pantalla asegurarnos que el combo este en la opción “**Nodo= localhostNode03, Servidor=server1**” y presionar el botón “**Nuevo…**”.

El nombre de la Célula puede variar de acuerdo a la configuración de instalación del Websphere. En el caso de ejemplo tenemos el nombre **Nodo= localhostNode03, Servidor=server1**



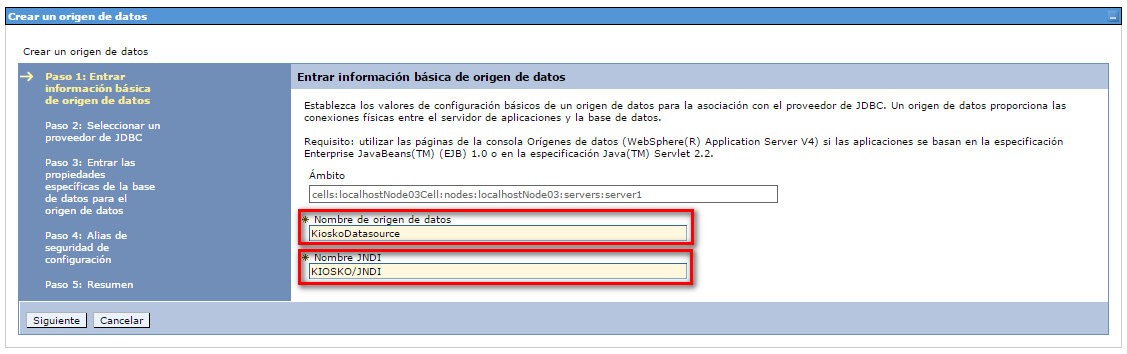
Se presentara la siguiente ventana:



En esta ventana ingresar los valores para los campos siguientes

* Nombre de Origen de datos: “**KioskoDatasource**”
* Nombre de JNDI: “**KIOSKO/JNDI**”

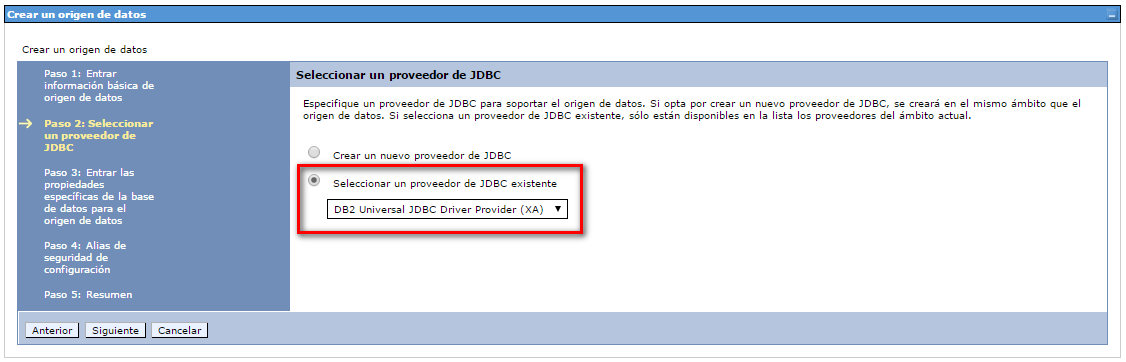
Como se muestra en la pantalla siguiente, luego presionar el botón siguiente.



En esta ventana seleccionar los valores para los campos siguientes

Seleccionar un proveedor de JDBC existente: “**DB2 Universal JDBC Driver Provider (XA)**”

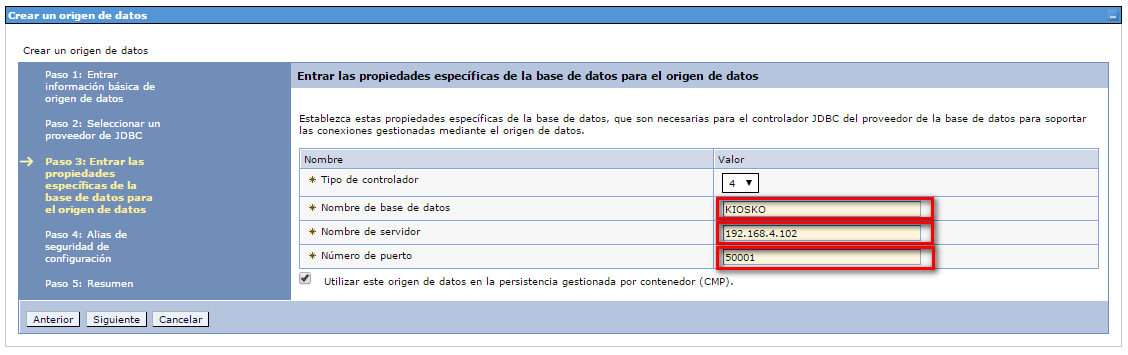
Como se muestra en la pantalla siguiente, luego presionar el botón siguiente.



En esta ventana ingresamos los valores para los campos siguientes

* Tipo de controlador: “**4**”
* Nombre de base de datos: “**KIOSKO**”
* Nombre de servidor: “**192.168.4.102**”
* Número de puerto: “**50001**”

Como se muestra en la pantalla siguiente, luego presionar el botón “**Siguiente**”.



Luego nos mostrara la siguiente pantalla, en esta pantalla seleccionamos la opción “**Siguiente**”

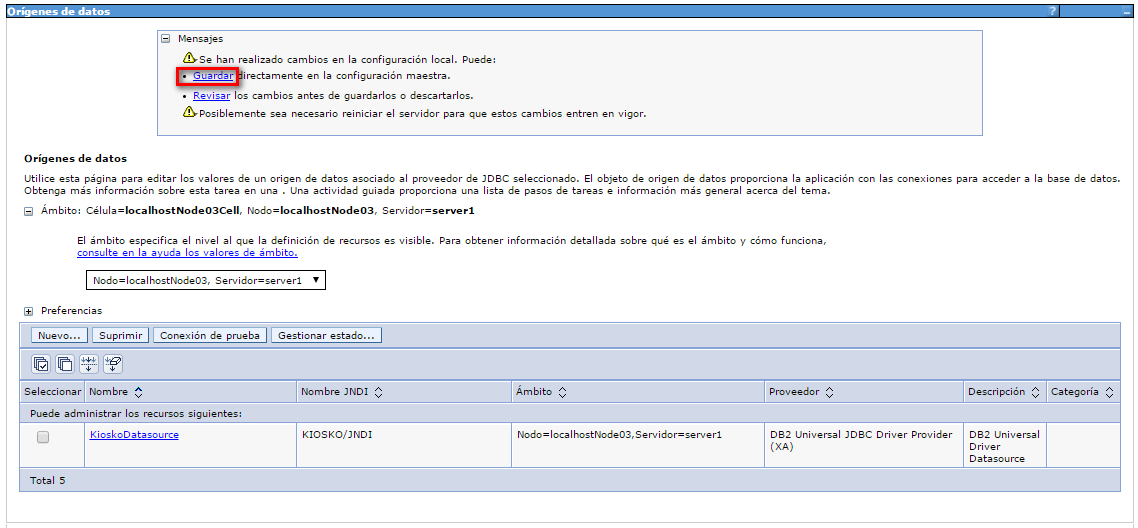


Se presentara la siguiente ventana:

Esta ventana nos muestra un resumen de la configuración del origen de datos que estamos creando, luego presionar el botón “**Finalizar**”.

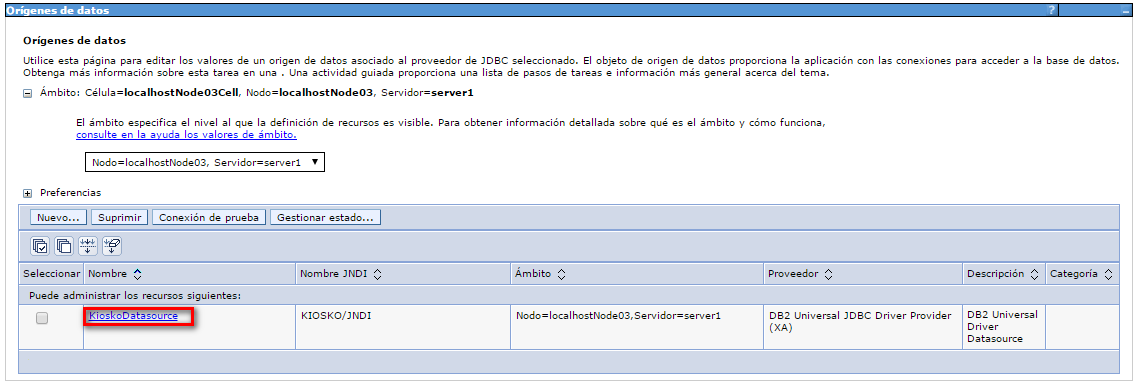


Luego nos mostrara la siguiente pantalla, en esta pantalla seleccionamos la opción “**Guardar**”

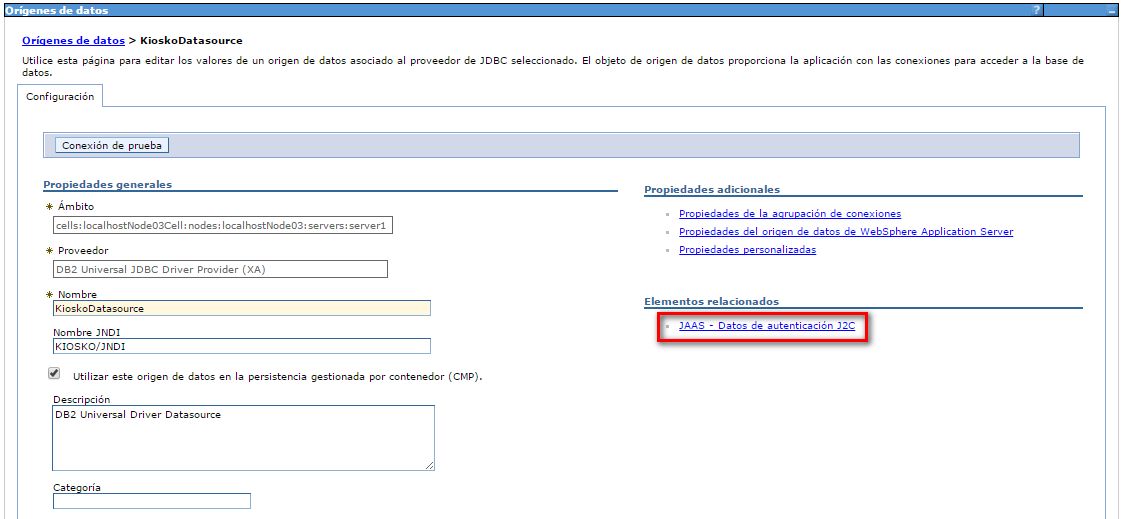


# Crear JAAS – Datos de autenticación J2C.

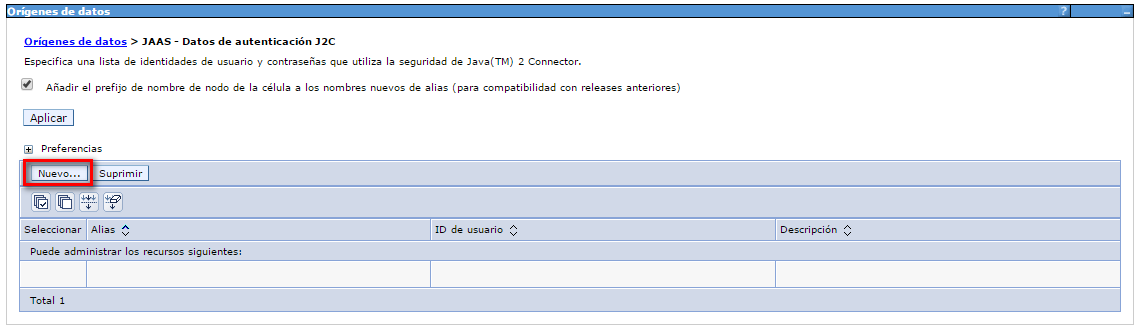
Luego nos mostrara la siguiente pantalla donde podremos visualizar nuestro origen de datos “**KioskoDatasource**” recién creado. Hacemos doble clic sobre nuestro origen de datos para editarlo.



Luego nos mostrara la siguiente pantalla, en esta pantalla seleccionamos la opción “**JAAS – Datos de autenticación J2C**”



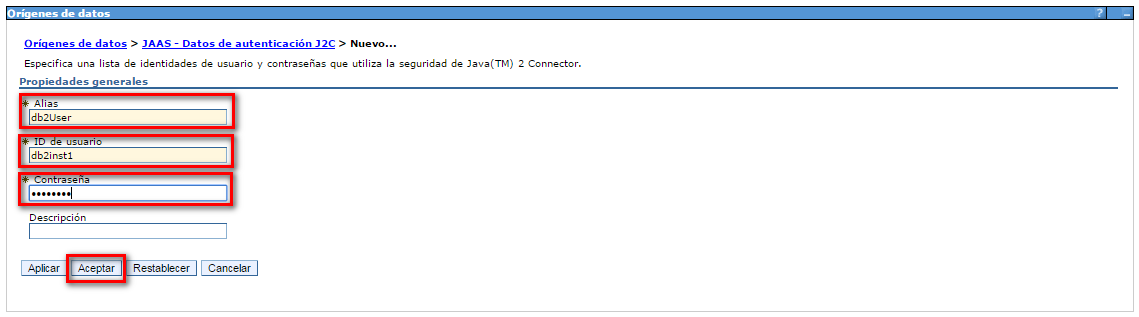
Luego nos mostrara la siguiente pantalla, en esta pantalla seleccionamos la opción “**Nuevo**”



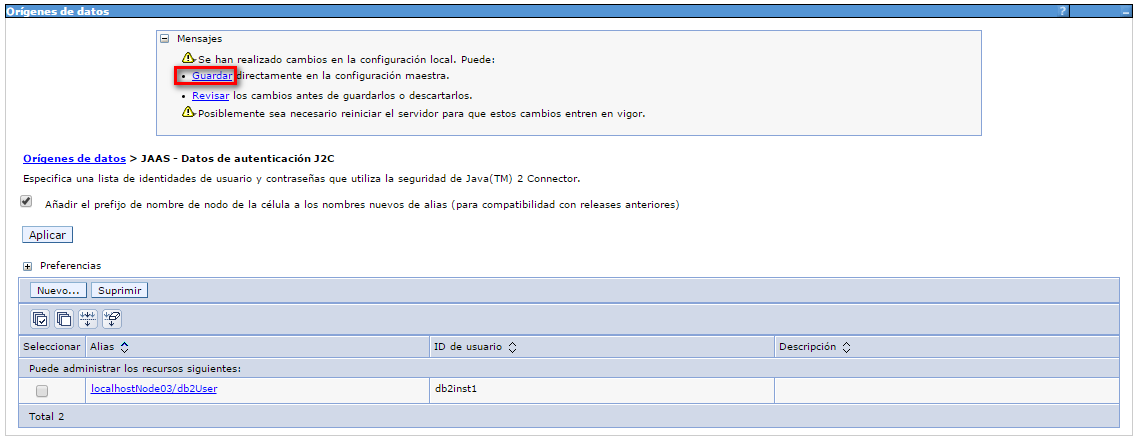
En esta ventana ingresamos los valores para los campos siguientes

* Alias: “**db2User**”
* ID de usuario: “**db2inst1**”
* Contraseña: “**Passw0rd**” (En nuestro caso de ejemplo)

Como se muestra en la pantalla siguiente, luego presionar el botón “**Aceptar**”.



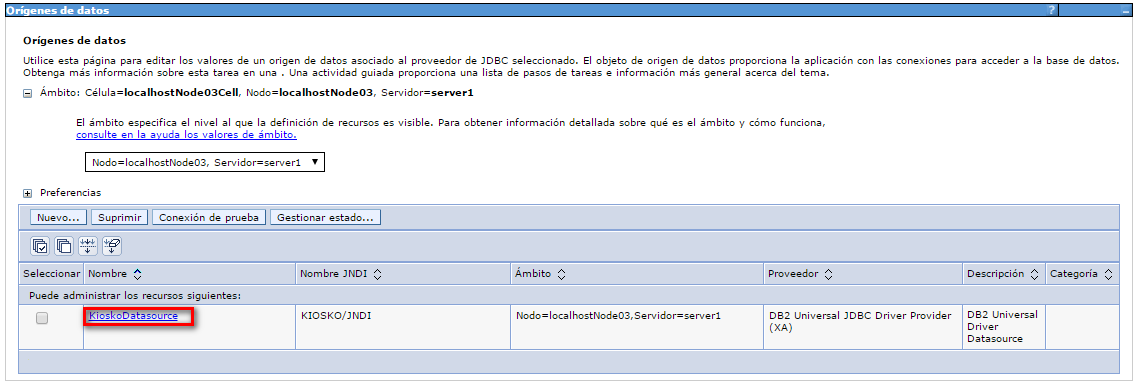
Luego nos mostrara la siguiente pantalla, en esta pantalla seleccionamos la opción “**Guardar**”



Luego nos mostrara la siguiente pantalla donde podremos visualizar nuestros datos de autenticación para nuestro origen de datos. Seleccionamos el link “**Orígenes de datos**” para visualizar nuestro origen de datos.



Luego nos mostrara la siguiente pantalla donde podremos visualizar nuestro origen de datos “**KioskoDatasource**”. Hacemos doble clic sobre nuestro origen de datos para editarlo.

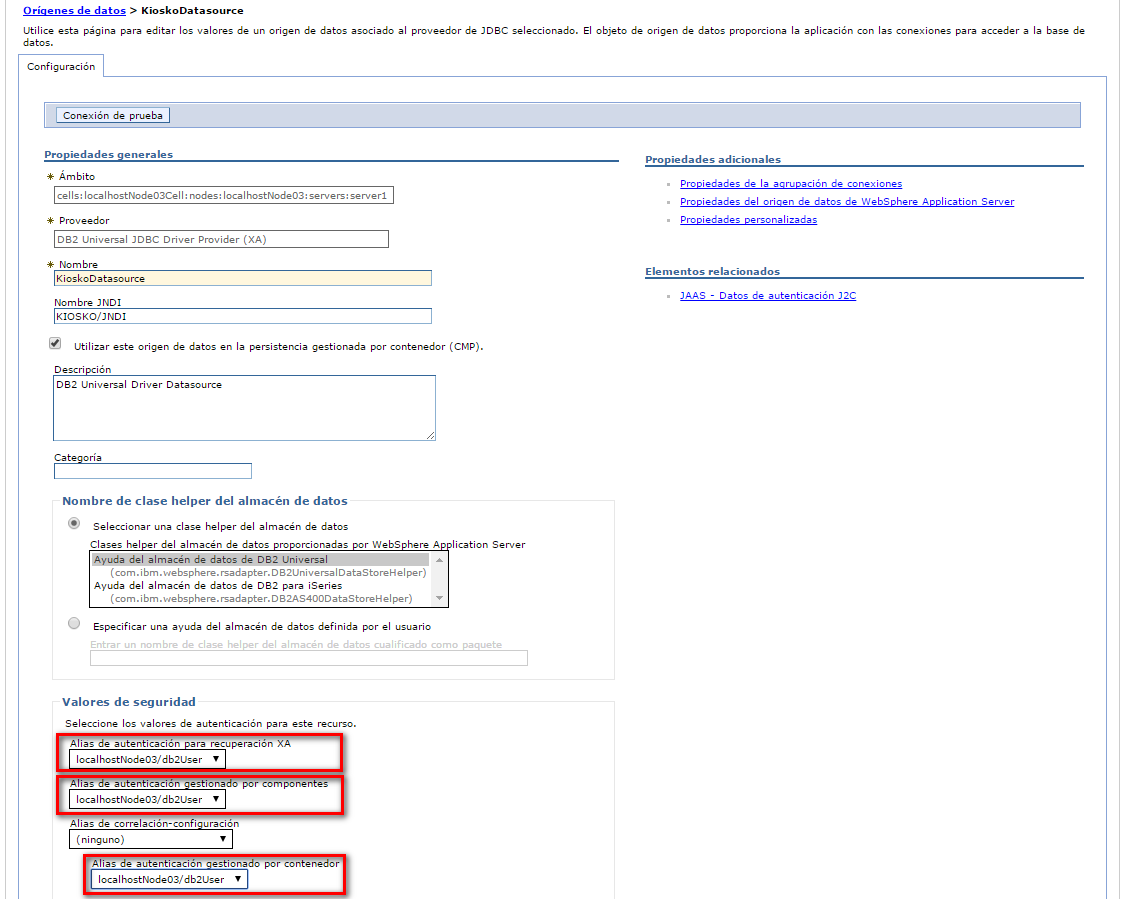


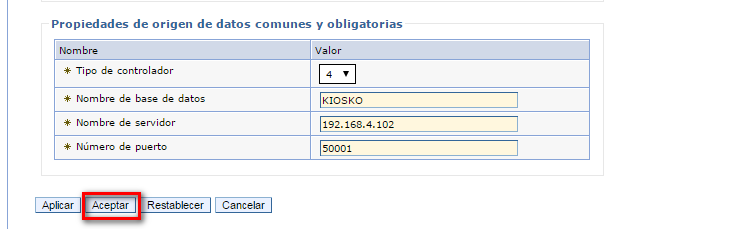
Luego nos mostrara la siguiente pantalla, en esta pantalla seleccionamos los valores siguientes en la sección “**Valores de seguridad**”

* Alias de autenticación para recuperación de XA: “**localhostNode03/db2User**”
* Alias de autenticación gestionado por componentes: “**localhostNode03/db2User**”
* Alias de autenticación gestionado por contenedor: “**localhostNode03/db2User**”

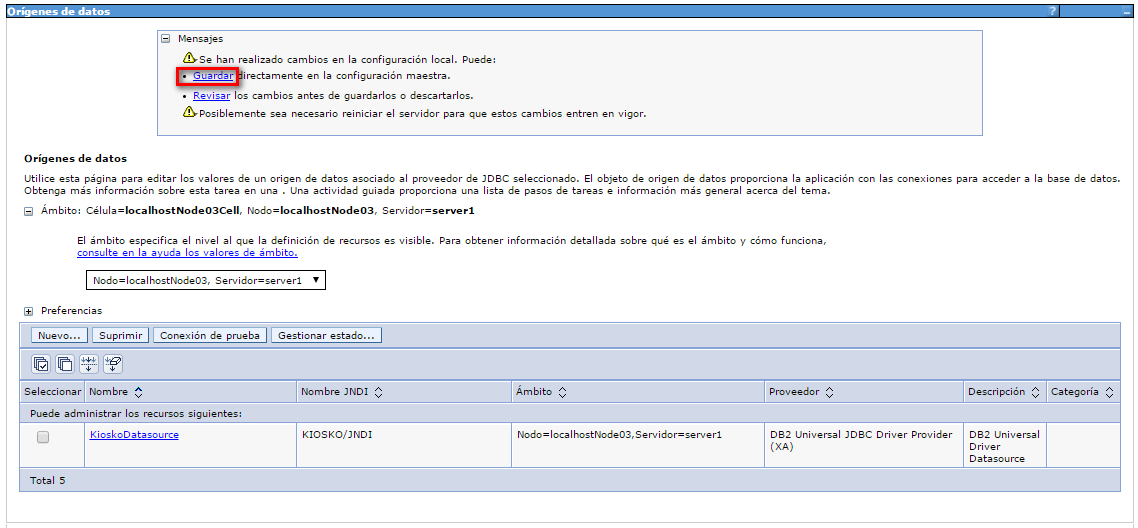
Los valores seleccionados corresponden al JAAS que hemos creado en el paso anterior.

Como se muestra en la pantalla siguiente, luego presionar el botón “**Aceptar**”.

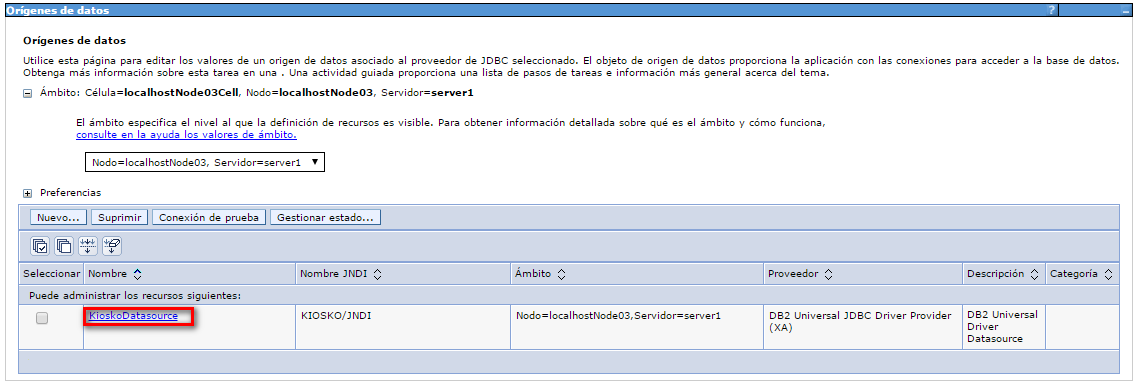




Luego nos mostrara la siguiente pantalla, en esta pantalla seleccionamos la opción “**Guardar**”



Luego nos mostrara la siguiente pantalla donde podremos visualizar nuestro origen de datos.



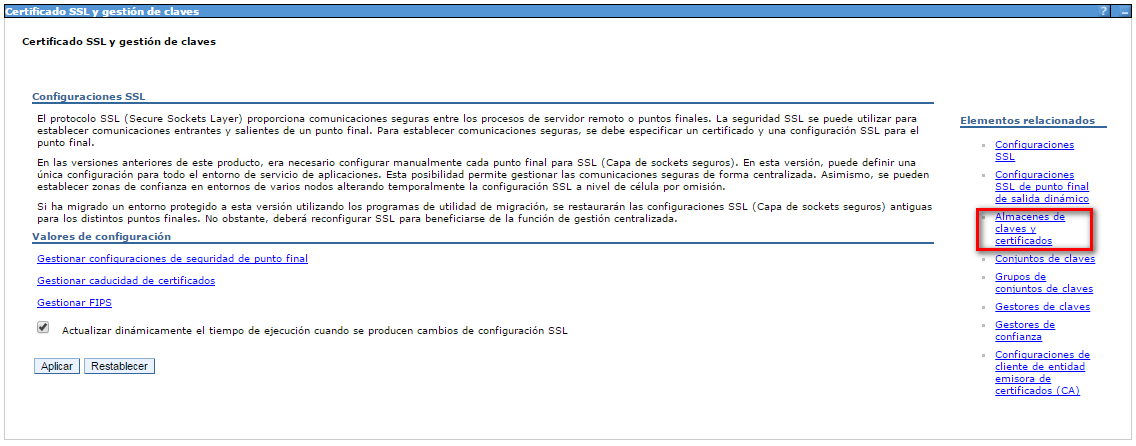
# Notificaciones Push para Android

Para permitir el envío de notificaciones Push al servicio de Google Cloud Message, es necesario realizar la siguiente configuración en el servidor Websphere.

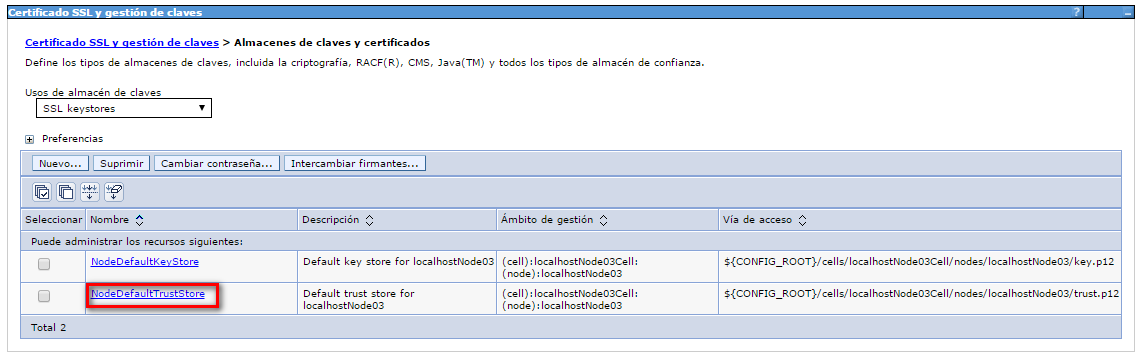
Seleccionamos la opción **“Seguridad” 🡪 “Certificado SSL y gestión de claves”** como se muestra en la imagen siguiente:



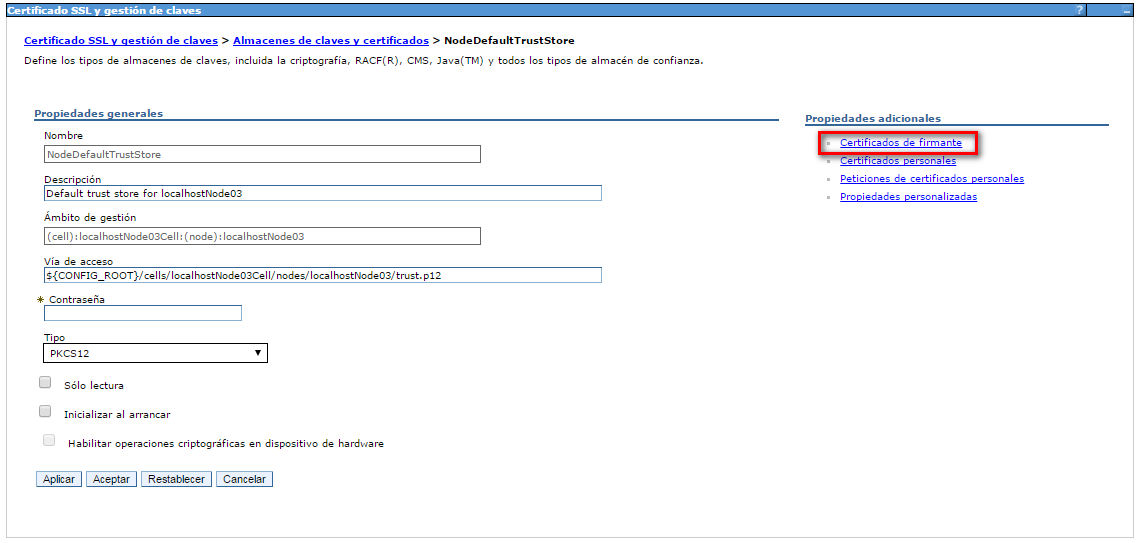
Mostrara la siguiente pantalla, en ella seleccionamos la opción “**Almacenes de claves y certificados**”.



Mostrara la siguiente pantalla, en ella seleccionamos la opción “**NodeDefaultTrustStore**”



Mostrara la siguiente pantalla, en ella seleccionamos la opción “**Certificados del firmante**”



Mostrara la siguiente pantalla, en ella seleccionamos la el botón “**Recuperar de puerto**”



Luego nos mostrara la siguiente pantalla, en esta pantalla seleccionamos los valores siguientes:

* Host: “**android.googleapis.com**”
* Puerto: “**443**”
* Alias: “**android.googleapis.com\_cert**”

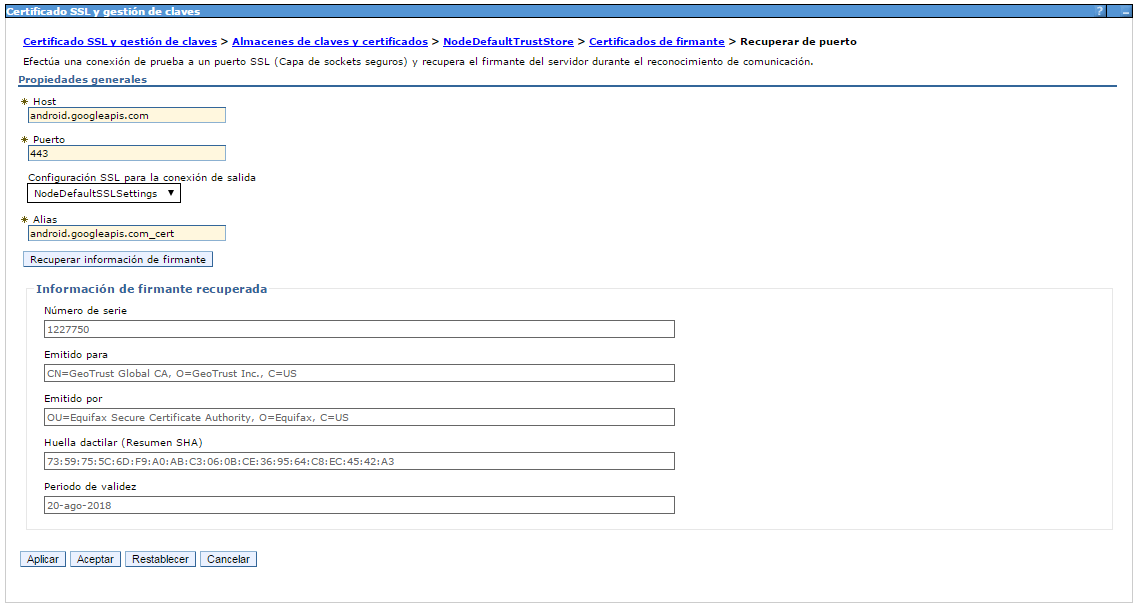
Esta información ha sido extraída desde:

[www-01.ibm.com/support/knowledgecenter/SSKTXQ\_9.0.0/admin/config/config\_security\_gcm.dita](www-01.ibm.com/support/knowledgecenter/SSKTXQ_9.0.0/admin/config/config_security_gcm.dita" \o "www-01.ibm.com/support/knowledgecenter/SSKTXQ_9.0.0/admin/config/config_security_gcm.dita)

Como se muestra en la pantalla siguiente, luego presionar el botón “**Recuperar información del firmante**”.



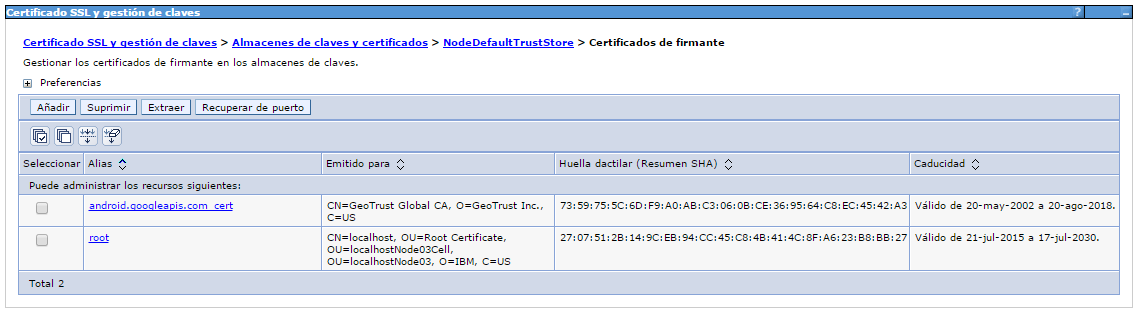
Luego nos mostrara la siguiente pantalla, luego presionar el botón “**Aceptar**”.



Luego nos mostrara la siguiente pantalla, en esta pantalla seleccionamos la opción “**Guardar**”



Luego visualizaremos nuestro certificado de firmante.



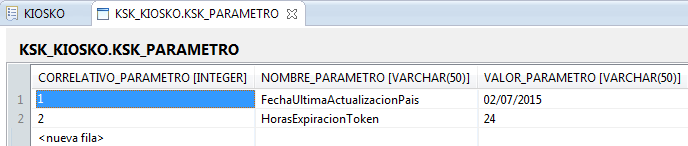
# Servicios Web

# Configuración de propiedades del servicio web.

# Configuración en tabla de base de datos.

Algunas propiedades del sistema se encuentran a nivel de base de datos, estas variables han sido ingresadas durante la carga inicial cuando se ejecutaron los scripts de base de datos.

Para visualizarlos se puede consultar la tabla **KSK\_KIOSKO.KSK\_PARAMETRO**, como se ve en la imagen siguiente.



Los parámetros que se pueden configurar son:

**FechaUltimaActualizacionPais:** Este parámetro indica cuando se realizo la última actualización de los países desde el servicio web que brinda Yanbal. Se recuerda que esta lista de países solo se actualizara una vez por día.

**HorasExpiracionToken :** Es la cantidad de horas que se establece para la duración del token o sesión en la aplicación web, si la sesión supera este numero será una sesión invalida y por lo tanto se deberá ingresar nuevamente al sistema KIOSKO.

# Configuración en archivo de propiedades.

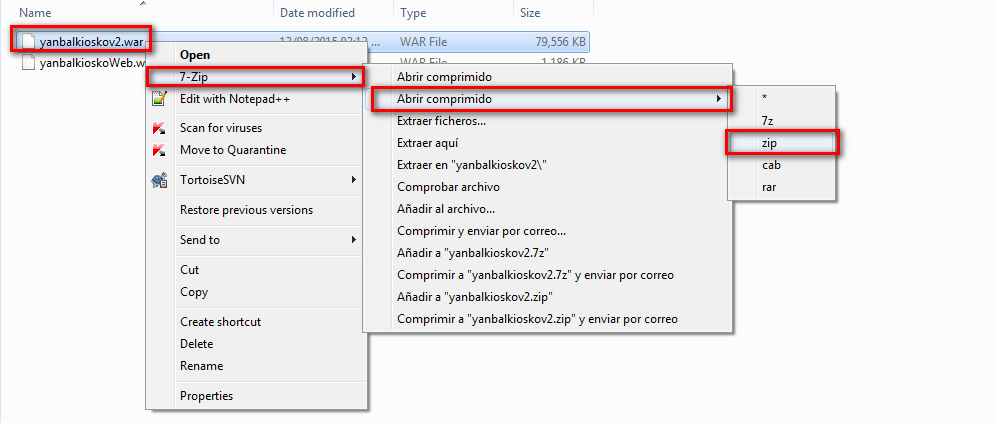
Para hacer cambios en la configuración del backend, basta con realizar las modificaciones en los archivos properties del mismo.

Los archivos de propiedades son los siguientes:

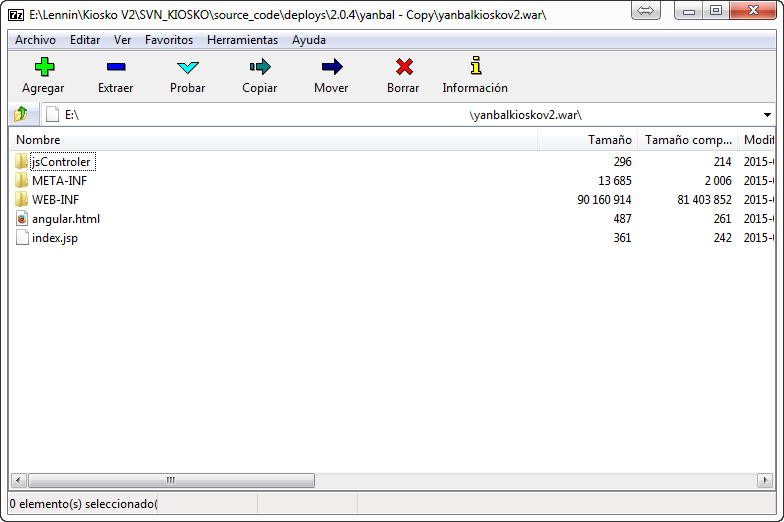
* **kioskov2.properties**

Para modificar este archivo de propiedades, abriremos los archivos “**war**” del aplicativo con alguna herramienta zip, en nuestro ejemplo se realizara con el 7zip como se muestra en la imagen siguiente, clic derecho sobre “**yanbalkioskov2.war**”.

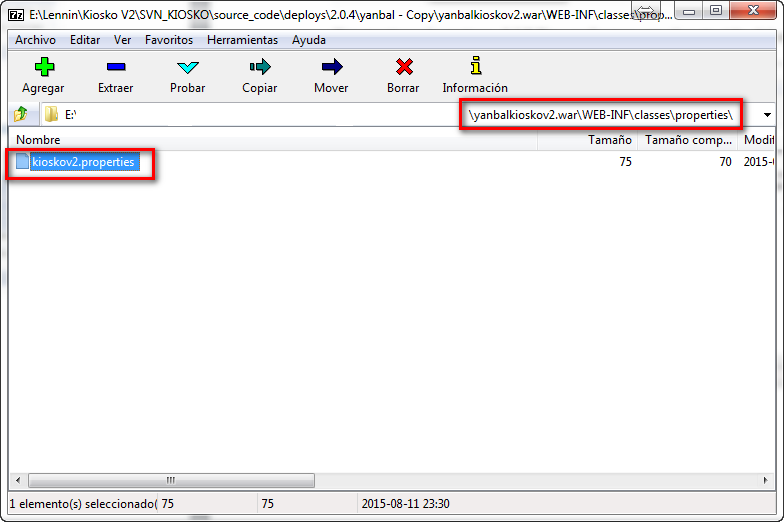
**“7-Zip” 🡪 “Abrir comprimido” 🡪 “zip”**



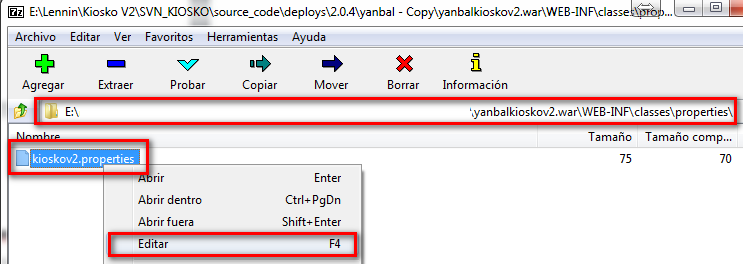
Nos mostrara la ventana del **7-Zip** y podremos ver el contenido del archivo.



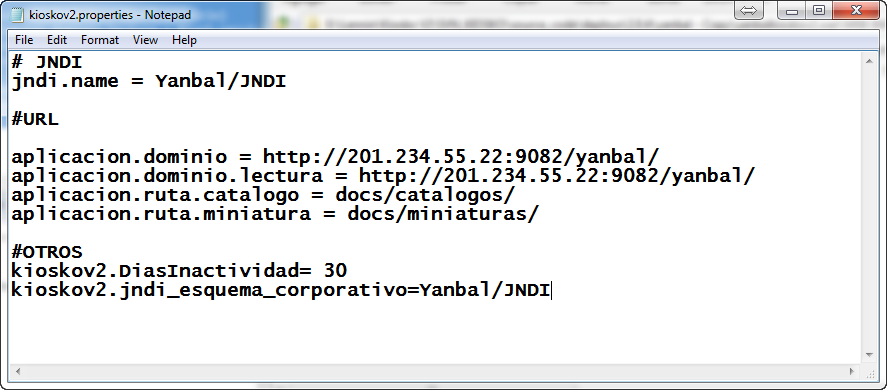
Nos ubicamos en la ruta siguiente dentro del archivo war “**WEB-INF\classes\properties**”



Nos ubicamos en la ruta siguiente dentro del archivo war “**WEB-INF\classes\properties**”



Veremos el contenido del archivo de propiedades y dentro podremos configurar los parámetros del sistema, como se muestra en la pantalla siguiente.



Los parámetros que se pueden configurar son:

**jndi.name :** Nombre del datasource que apunta a la base de datos del sistema backend.

**aplicacion.dominio :** Nombre o url de dominio del sistema

**aplicacion.dominio.lectura :** Nombre o url de dominio del sistema

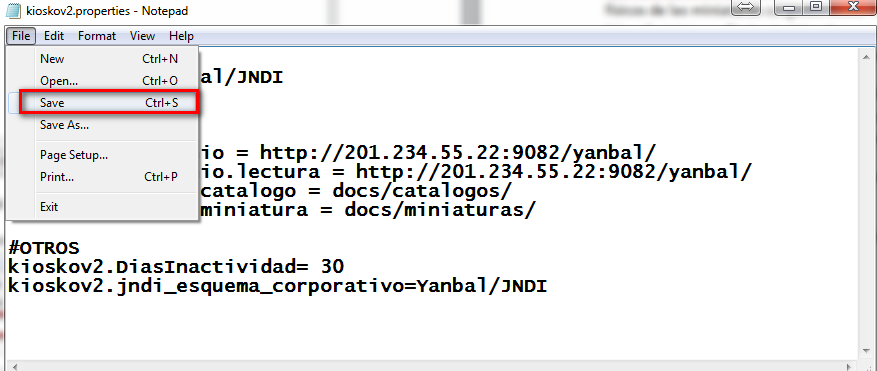
**aplicacion.ruta.catalogo :** Nombre de la ruta url por donde se puede acceder a los archivos físicos de los catálogos cargados desde el sistema, en el caso ejemplo se encuentra el valor: “**docs/catalogos/**”, donde “**docs**” corresponde al nombre de la ruta simbólica que se creara luego de la instalación del sistema en el servidor de aplicaciones.

**aplicacion.ruta.miniatura :** Nombre de la ruta url por donde se puede acceder a los archivos físicos de las miniaturas cargados desde el sistema, en el caso ejemplo se encuentra el valor: “**docs/miniaturas/**”, donde “**docs**” corresponde al nombre de la ruta simbólica que se creara luego de la instalación del sistema en el servidor de aplicaciones.

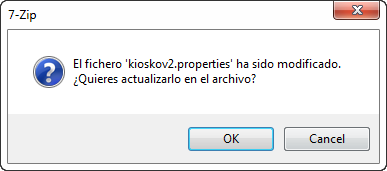
**kioskov2.DiasInactividad :** Cantidad de días de inactividad, este parámetro se utiliza para obtener todos los dispositivos que se encuentran activos durante los últimos n días que se establezca en este atributo para realizar el envío de notificaciones push.

**kioskov2.jndi\_esquema\_corporativo :** Nombre del datasource donde se guardara el log de envío push.

Luego de modificar los valores, guardar los cambios como se muestra en la imagen siguiente y cerrar el archivo de texto.



Se mostrara el siguiente mensaje por parte del **7-Zip**, clic en “**OK**” y esperar que se actualice el archivo “**war**”



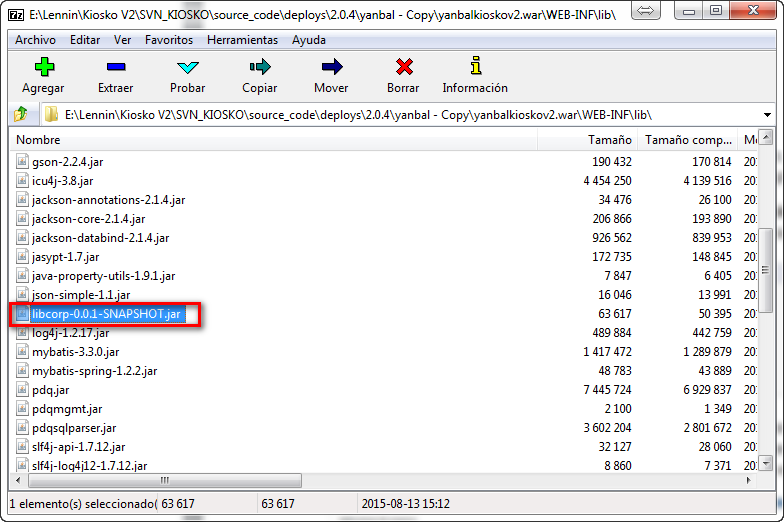
# Configuración de propiedades de la librería corporativa.

Para hacer cambios en la configuración de la librería corporativa basta con realizar las modificaciones en los archivos properties del mismo.

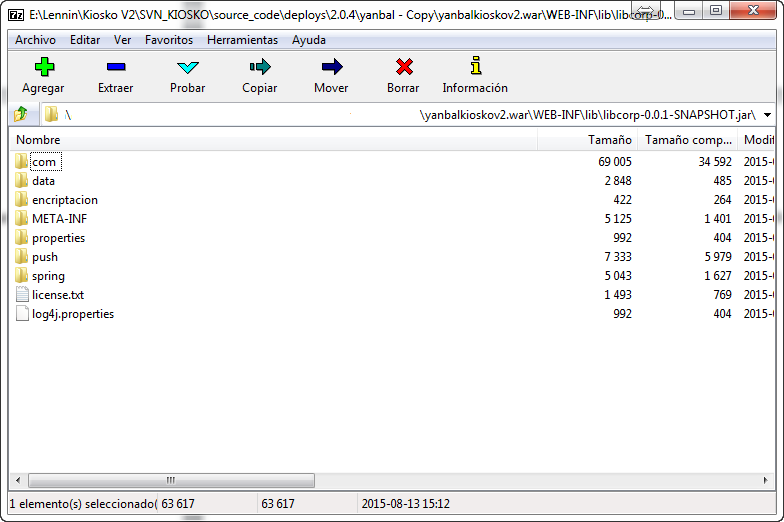
Los archivos de propiedades son los siguientes:

* **encriptación/encryptionParameters.properties**
* **properties/log4j.properties**
* **push/notification.properties**
* **push/pushAndroid.properties**
* **push/pushIOS.properties**
* **push/pushWindowsPhone.properties**

Nos ubicamos en la ruta siguiente dentro del archivo war “**WEB-INF\lib**” y seleccionamos el jar “**libcorp-0.0.1-SNAPSHOT.jar**” y doble click.

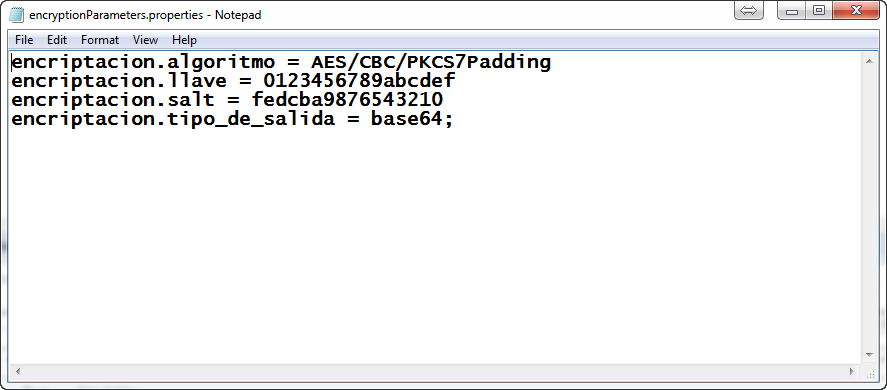


Nos mostrara la ventana del **7-Zip** y podremos ver el contenido del archivo.



Ubicamos los properties que se indican y los modificamos.

**encriptación/encryptionParameters.properties:** Para la encriptación.



Los parámetros que se pueden configurar son:

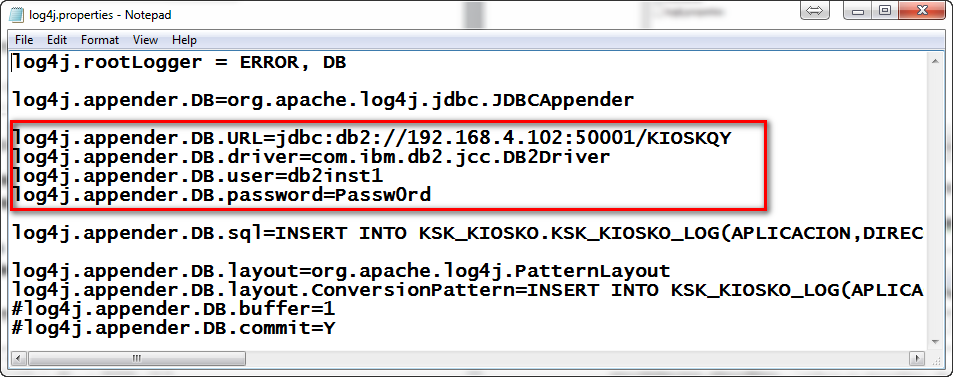
**encriptacion.algoritmo:** Define el algoritmo de encriptación.

**encriptacion.llave:** Define la llave de la encriptación.

**encriptacion.salt :** Define el salt de la encriptación.

**encriptacion.tipo\_de\_salida :** Define el tipo de salida de encriptación.

**properties/log4j.properties:** Para el registro de log de la aplicación en la base de datos.



Los parámetros que se pueden configurar son:

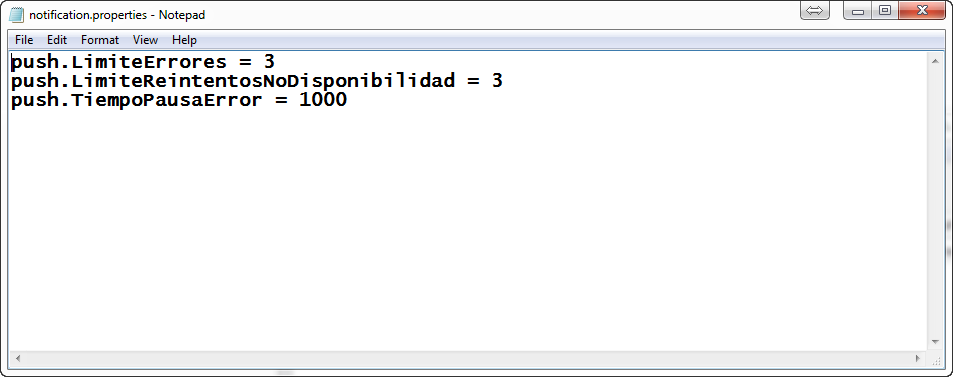
**log4j.appender.DB.URL:** Url jdbc para conexión a la base de datos.

**log4j.appender.DB.driver:** Driver que se usara para la conexión a la base de datos.

**log4j.appender.DB.user:** Usuario para la conexión a la base de datos.

**log4j.appender.DB.password:** Contraseña para la conexión a la base de datos.

**push/notification.properties:** Para el envío de notificaciones push.



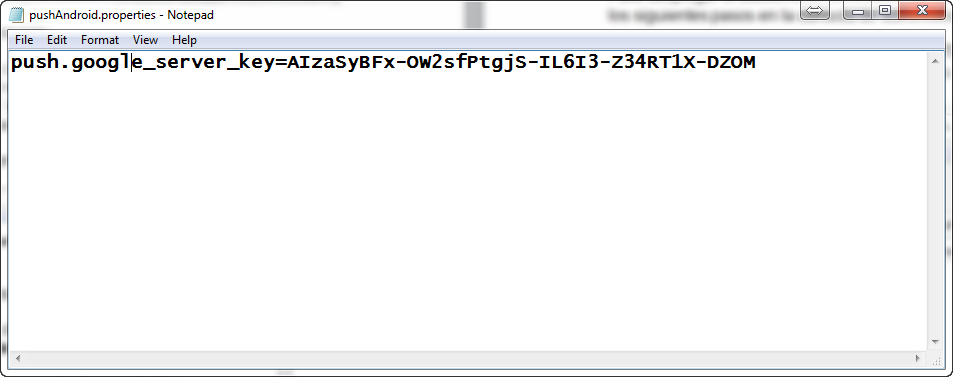
Los parámetros que se pueden configurar son:

**push.LimiteErrores:** Cantidad máxima de limite de errores durante el envío.

**push.LimiteReintentosNoDisponibilidad:** Numero de veces de reintento en caso de error durante el envío.

**push.TiempoPausaError:** Tiempo de pause entre los reintentos.

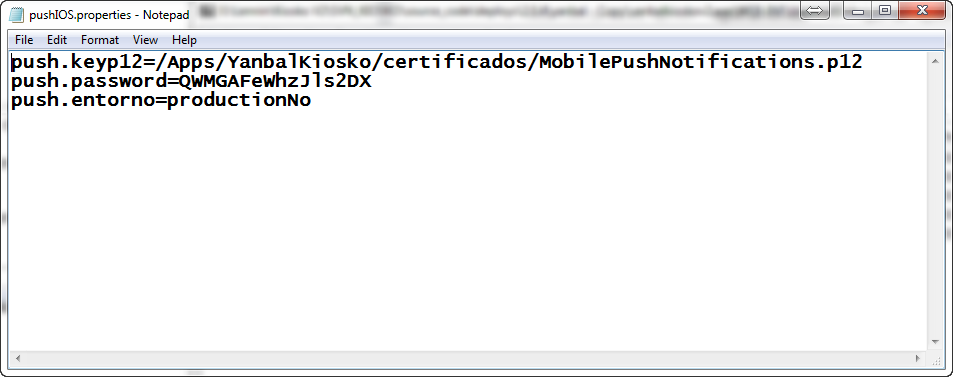
**push/pushAndroid.properties:** Para la conexión con el servidor de Google para el envío de las notificaciones push.



Los parámetros que se pueden configurar son:

**push.google\_server\_key:** clave de conexión del aplicativo al servidor de google, esta clave se debe obtener desde una cuenta del vinculada al aplicativo.

**push/pushIOS.properties:** Para la conexión con el servidor de Apple para el envío de las notificaciones push.



Los parámetros que se pueden configurar son:

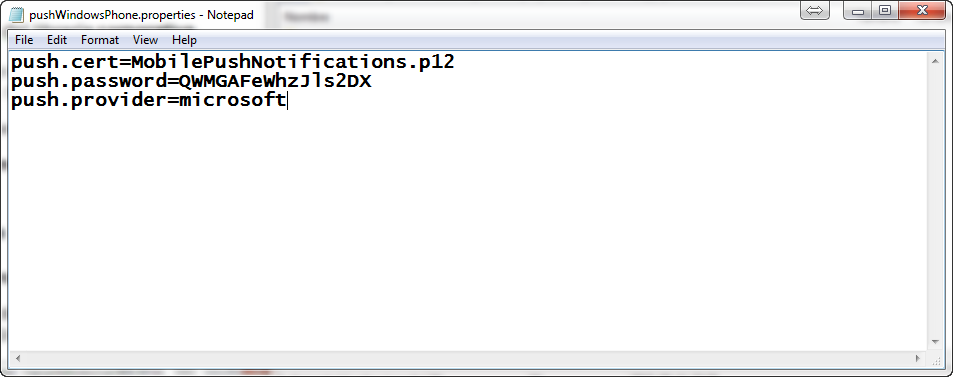
**push.keyp12:** Ruta donde se encuentra el archivo p12.

**push.password:** Clave con la que se genero el archivo p12.

**push.entorno:** Define si es para el entorno de producción o entorno sandbox, para indicar el entorno de producción poner el texto “**production**”.

.

**push/pushWindowsPhone.properties:** Para la conexión con el servidor de Windows para el envío de las notificaciones push.



Los parámetros que se pueden configurar son:

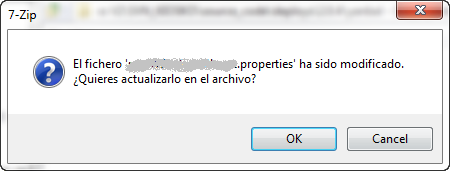
**push.cert:** Ruta donde se encuentra certificado.

**push.password:** Clave con la que se genero el certificado.

**push.provider:** Define el proveedor, el texto por defecto es “**microsoft**”.

Luego de modificar los valores, guardar los cambios en cada archivo cerrar cada uno de los archivos properties modificados.

Se mostrara el siguiente mensaje por parte del **7-Zip**, clic en “**OK**” y esperar que se actualice el archivo “**war**”



# Instalación del servicio web.

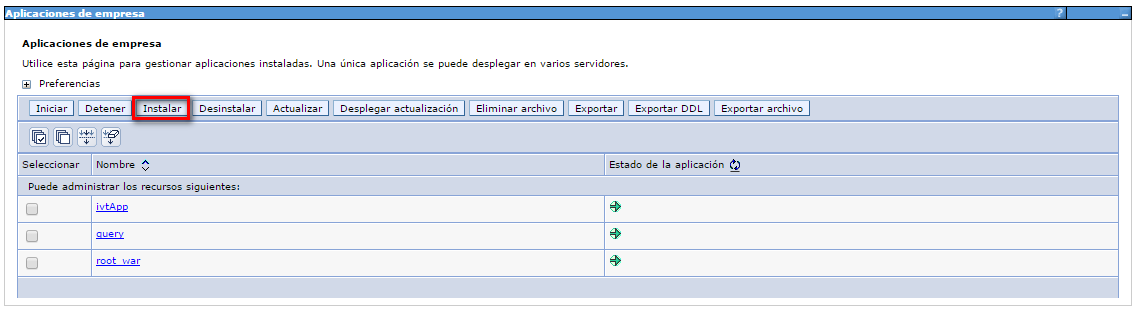
# Despliegue del aplicativo en Websphere.

Para desplegar el backend de servicios web del sistema kiosko, se realizara siguiendo los siguientes pasos en la consola de administración del Websphere.

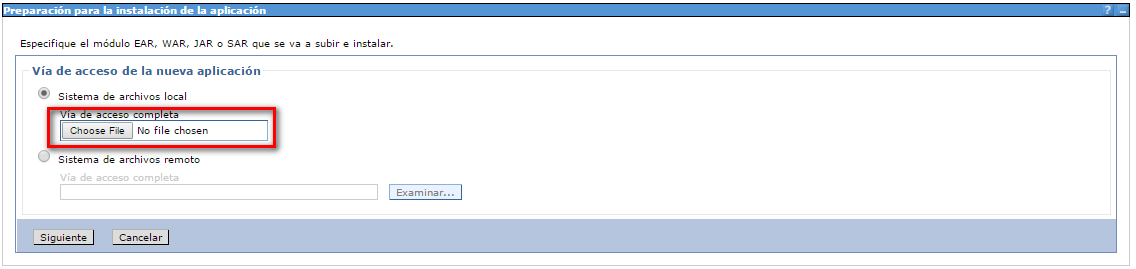
Seleccionamos la opción **“Aplicaciones”** 🡪 **“Tipo de Aplicación” 🡪 ”Aplicaciones de empresa de WebSphere”** como se muestra en la imagen siguiente:



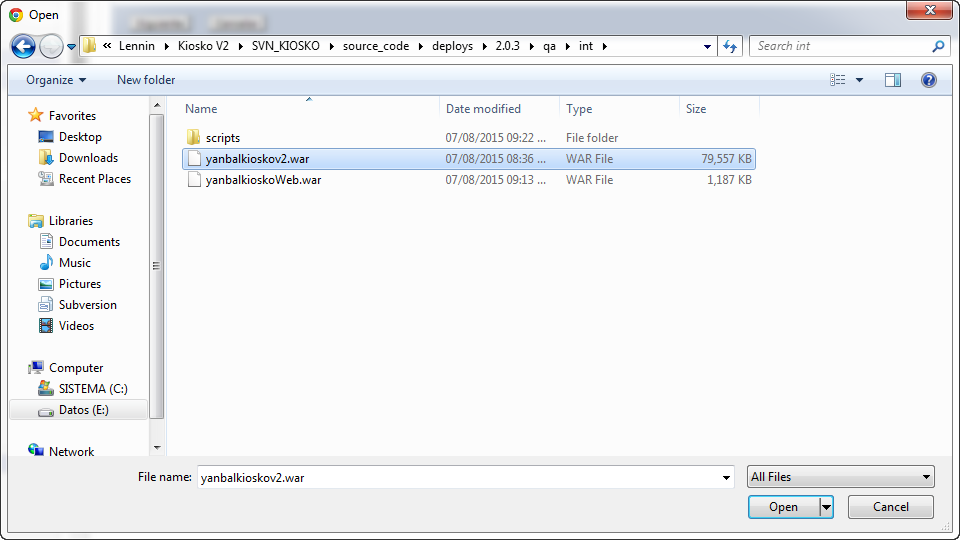
Nos mostrara la siguiente pantalla, en esta pantalla presionamos el botón instalar. (Si ya tenemos una instalación previa de nuestro backend, será necesario desinstalarlo, para ello seguiremos los pasos descritos mas abajo siguiendo el [siguiente enlace](#_Desinstalación_del_servicio)).



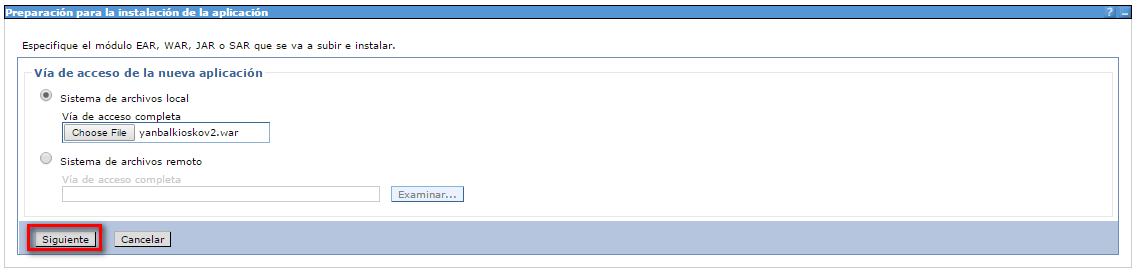
Nos presentara la siguiente pantalla donde seleccionaremos el “**war**” o “**ear**” del sistema, seleccionamos la opción “**Choose File**”.



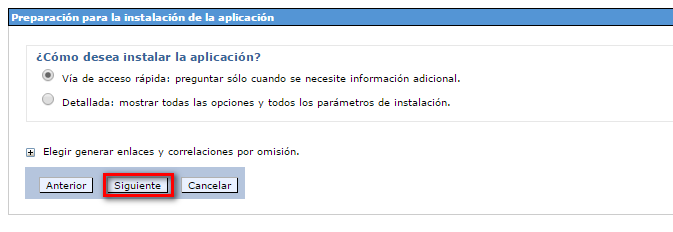
Nos presentara la siguiente pantalla del browser donde buscaremos nuestro archivo “**war**” o “**ear**” que deseamos desplegar, lo seleccionamos y presionamos el botón “**Open**”.



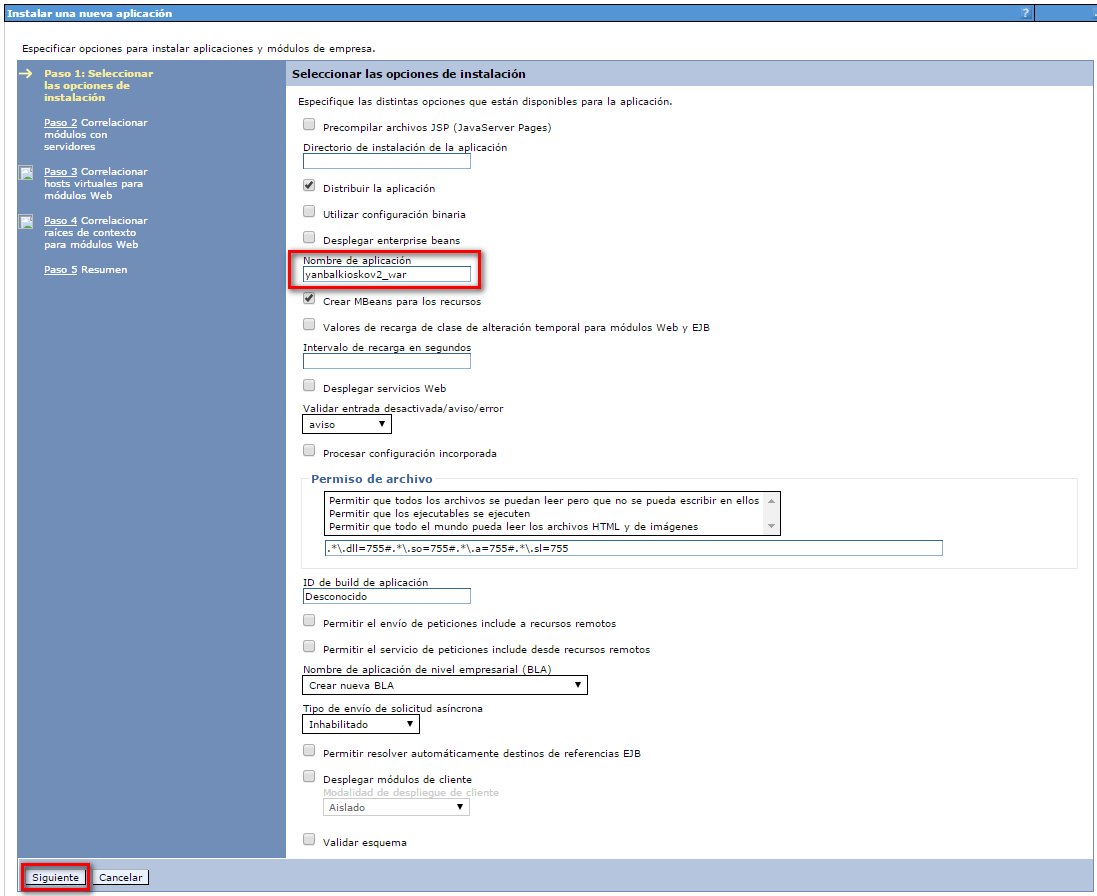
En la siguiente pantalla, seleccionamos el botón “**Siguiente**”.



En la siguiente pantalla, seleccionamos el botón “**Siguiente**”.



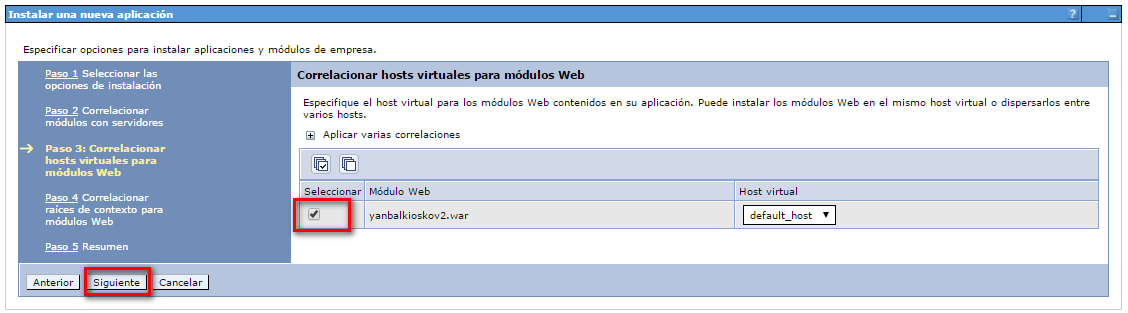
Se nos presentara la siguiente ventana, en ella podemos cambiar el nombre del Aplicativo en el campo “**Nombre de la aplicación**”, en el despliegue de ejemplo lo dejaremos con su nombre original por defecto y presionamos el botón “**Siguiente**”.



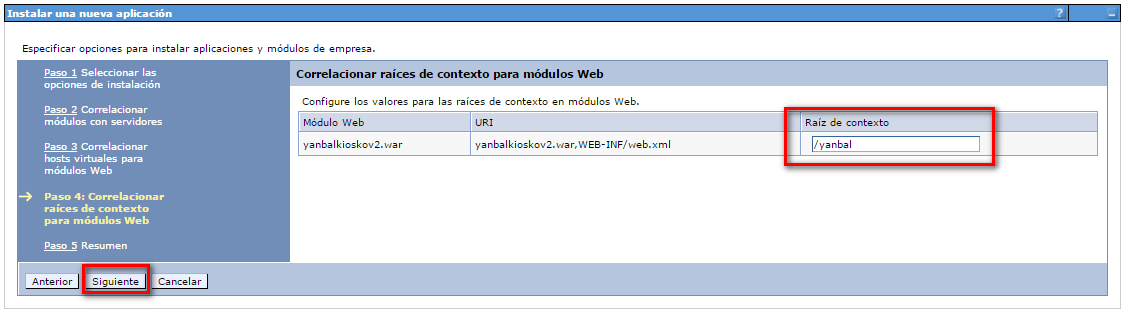
Se nos presenta la siguiente ventana, seleccionamos y activamos la casilla de verificación y presionamos el botón “**Siguiente**”.



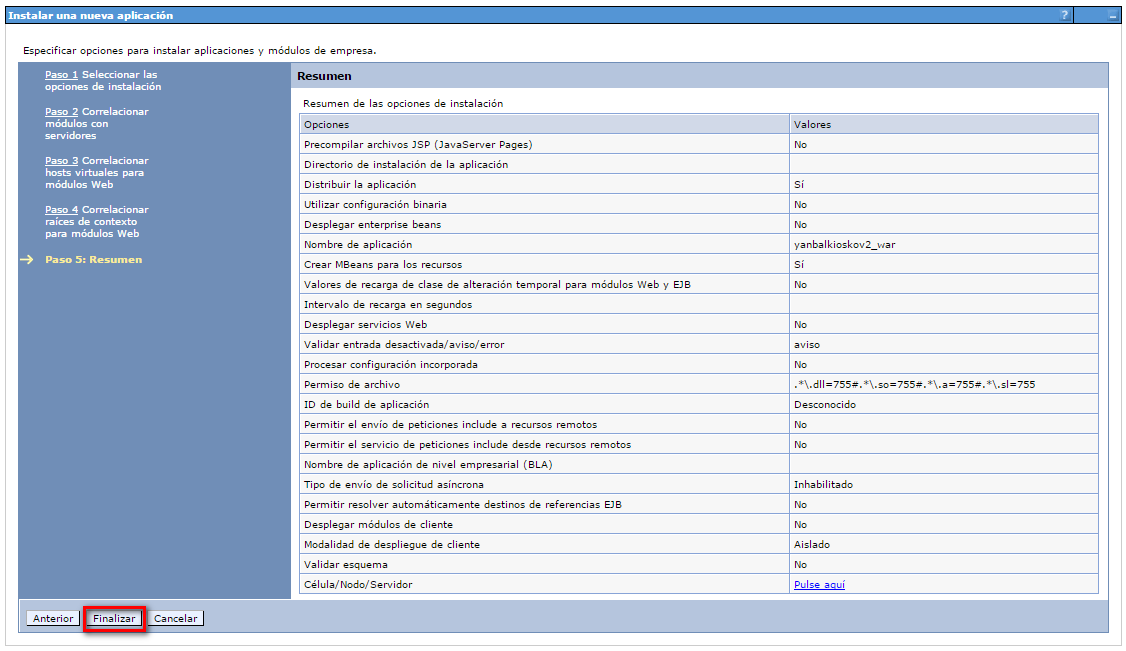
Se nos presenta la siguiente ventana, seleccionamos y activamos la casilla de verificación y presionamos el botón “**Siguiente**”.



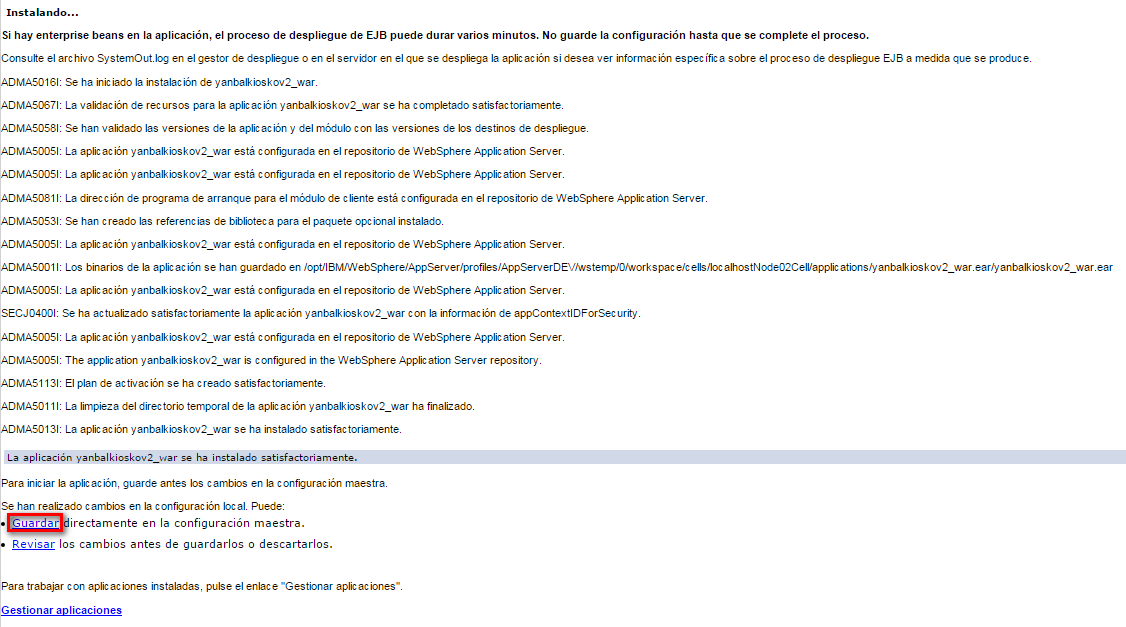
Se nos presenta la siguiente ventana, ingresamos un nombre para el contexto, en el ejemplo asignaremos el nombre de “**/yanbal**” y presionamos el botón “**Siguiente**”.



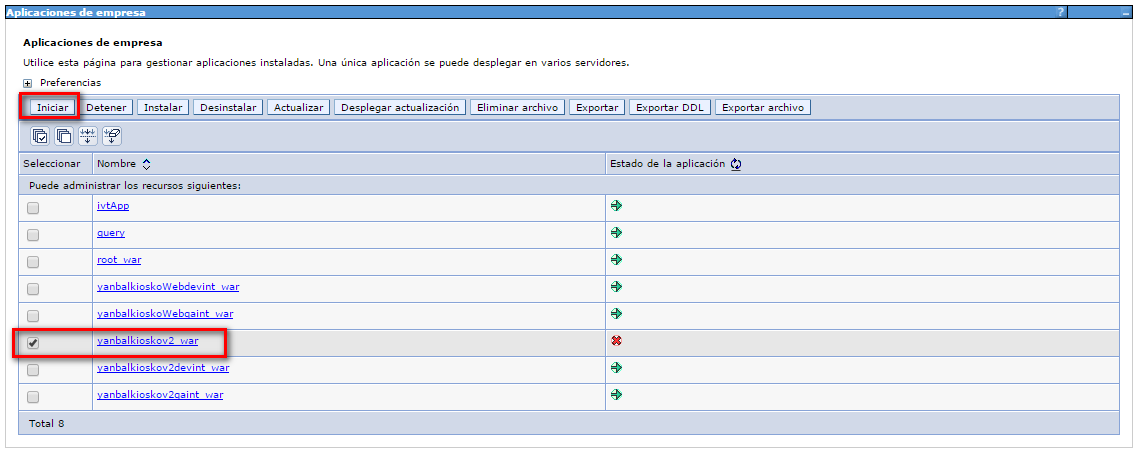
Nos muestra una pantalla con el resumen de los atributos del despliegue, presionamos el botón “**Finalizar**”.



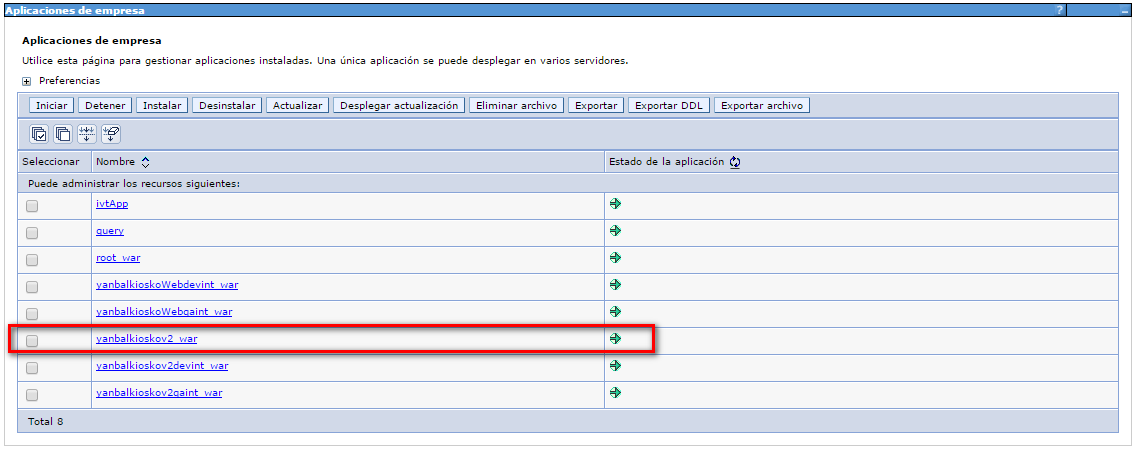
Nos muestra la pantalla con la información del proceso de instalación, cuando finaliza presionamos la opción Guardar para confirmar la instalación.



Nos muestra el listado de aplicaciones instaladas en nuestro servidor, marcamos la casilla de verificación de nuestro sistema y presionamos el botón “**Iniciar**”.



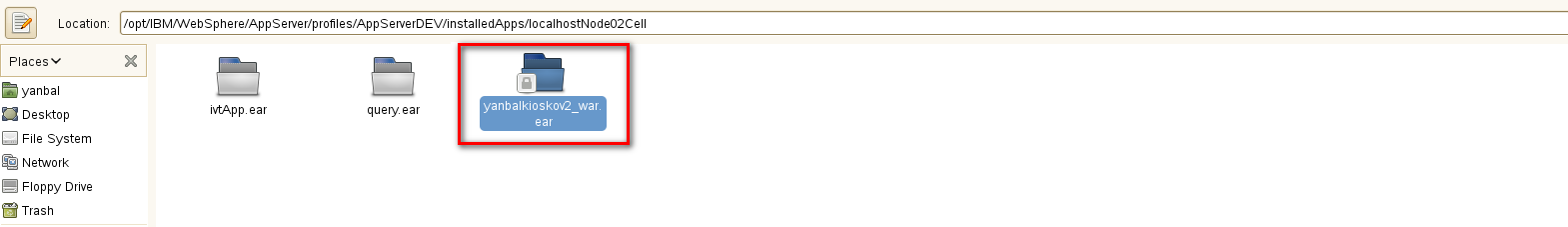
Luego de unos instantes veremos nuestro sistema ya iniciado.



# Creación de ruta simbólica en Linux.

Ahora ubicamos la carpeta donde se instalo nuestro aplicativo backend, en nuestro ejemplo se encuentra instalado en la siguiente ruta:

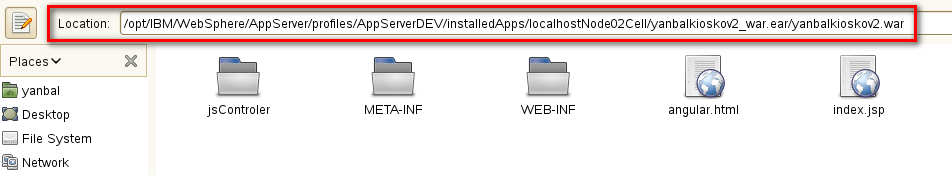
* **/opt/IBM/WebSphere/AppServer/profiles/AppServerDEV/installedApps/localhostNode03Cell**



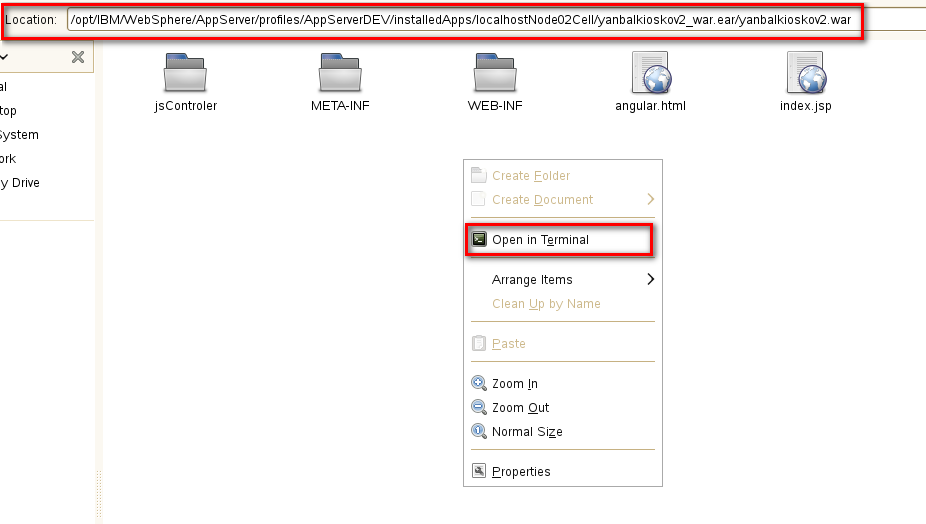
Dentro de ella encontraremos la carpeta:

* **yanbalkioskov2\_war.ear**

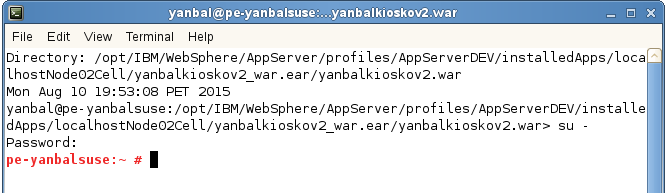
Entramos a nuestra carpeta del aplicativo hasta ubicarnos dentro de la carpeta **“yanbalkioskov2.war”** como se aprecia en la imagen:



Abrimos una terminal de línea de comandos, clic derecho y seleccionamos la opción “**Open in Terminal**”.



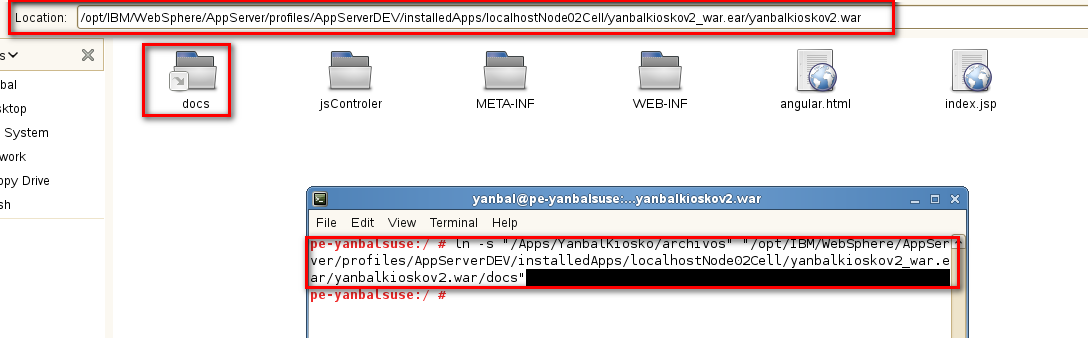
Nos mostrara la consola de comandos siguiente, iniciamos sesión como super usuario con el comando: “**su -**“ e ingresamos la contraseña de **root**.



Ejecutamos el siguiente comando para crear la ruta simbólica.

* **ln -s "/Apps/YanbalKiosko/archivos" "/opt/IBM/WebSphere/AppServer/profiles/AppServerDEV/installedApps/localhostNode02Cell/yanbalkioskov2\_war.ear/yanbalkioskov2.war/docs"**

Observaremos que luego de ejecutar el comando se mostrara una carpeta de nombre “**docs**” como se puede ver en la imagen.

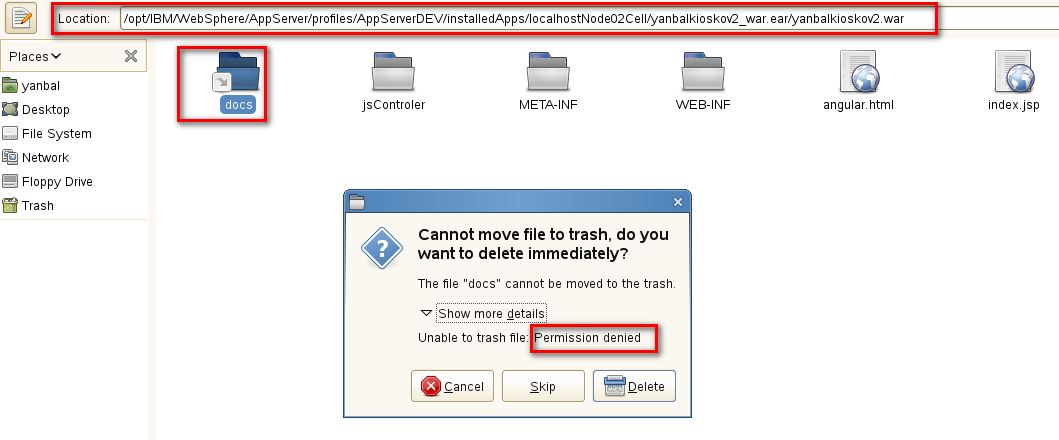


# Desinstalación del servicio web.

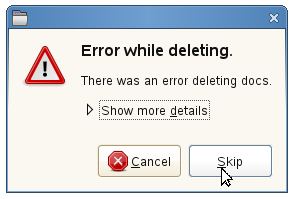
# Eliminación de ruta simbólica en Linux.

Para desinstalar el sistema primero debemos de eliminar la ruta simbolica, para ello nos ubicamos hasta el lugar donde se creo la misma.

En el ejemplo podemos ver que el sistema operativo no permite eliminar la ruta simbólica por falta de permisos.



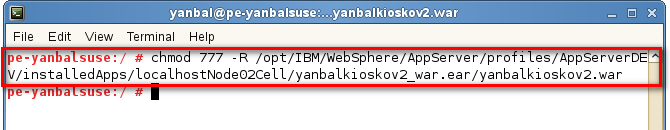
Y nos muestra el mensaje siguiente



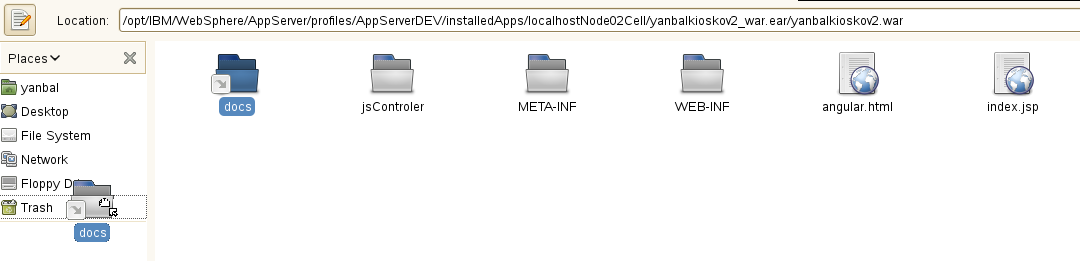
Para pasar el error debemos asignar permisos a la carpeta donde se encuentra creada la ruta simbólica, para ello ejecutamos el siguiente

comando:

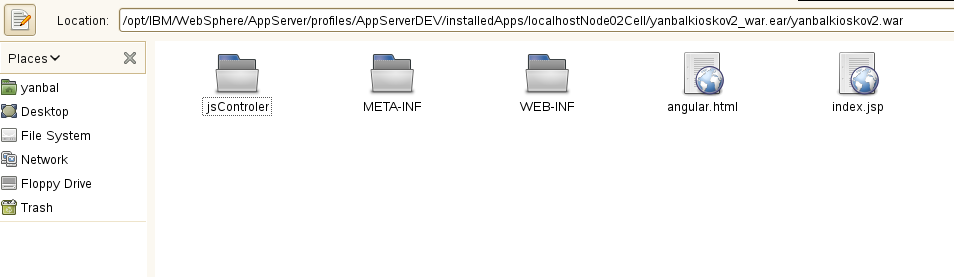
* **chmod 777 – R /ruta**



Una vez ejecutado el comando anterior, podremos eliminar la ruta simbólica con solo arrastrarla al tacho de basura “**Trash**” del sistema operativo, como se aprecia en la imagen siguiente:



Podemos ver que ya se elimino la ruta simbólica de la carpeta donde se encuentra nuestro aplicativo.



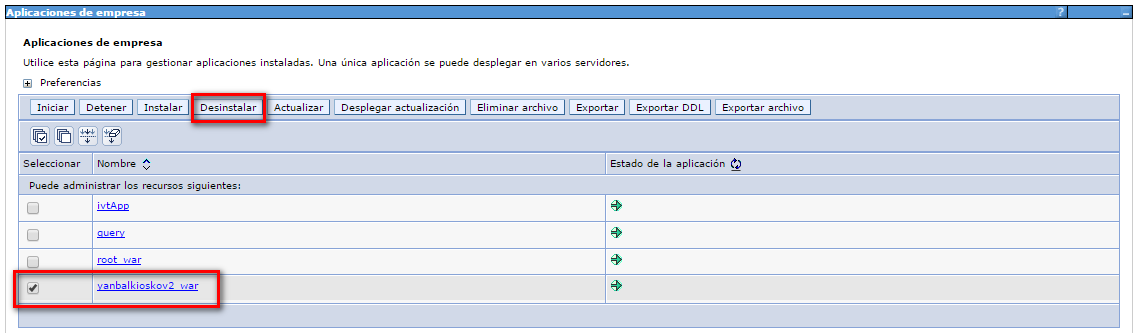
# Remover el despliegue del Websphere.

Una vez eliminado la ruta simbólica, procedemos a desinstalar el aplicativo desde la consola de Administración del Websphere.

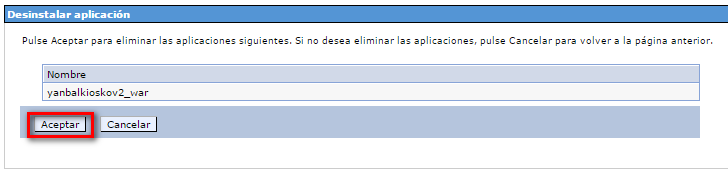
Seleccionamos la opción “**Aplicaciones**” 🡪 “**Tipo de Aplicación**”🡪”**Aplicaciones de empresa de WebSphere**” como se muestra en la imagen siguiente:



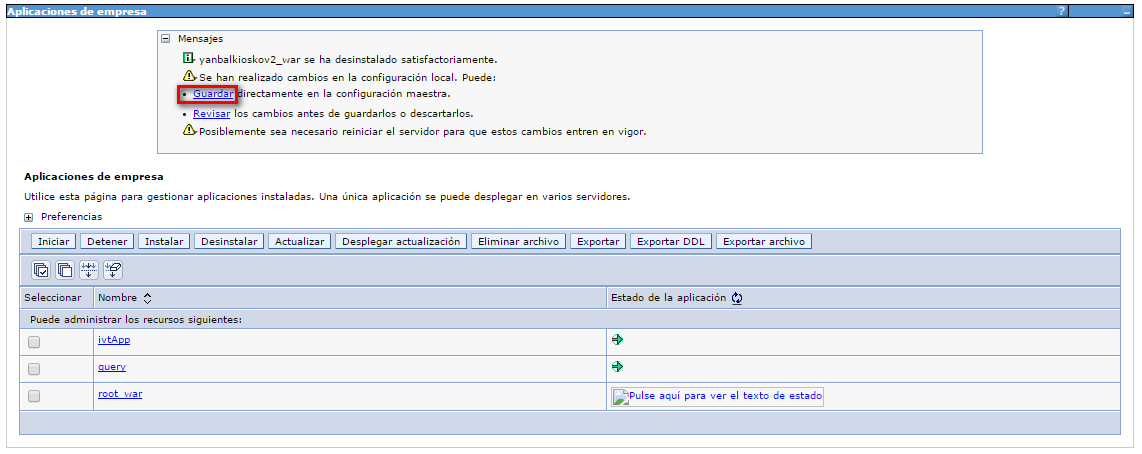
Nos mostrara la pantalla con la lista de aplicaciones instaladas, ubicamos nuestra aplicación y la seleccionamos activando la casilla de verificación y luego presionamos el botón “**Desinstalar**” como se muestra en la imagen.



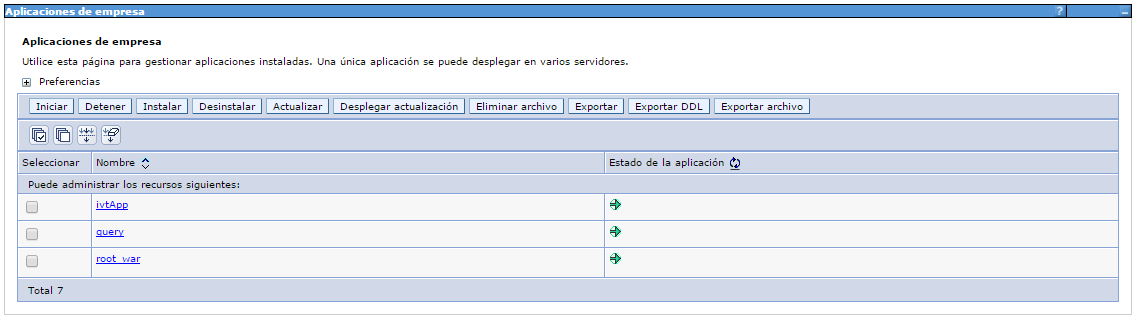
Nos mostrara la pantalla siguiente antes de continuar con la desinstalación del sistema, presionamos el botón “**Aceptar**”.



Luego pedirá confirmar los cambios, seleccionamos la opción “**Guardar**” para confirmar la desinstalación del aplicativo.



Luego de confirmar la desinstalación, veremos la pantalla de listado de aplicaciones sin nuestro aplicativo, como se muestra en la pantalla siguiente.



# Aplicación Web

# Configuración de propiedades del Sitio Web.

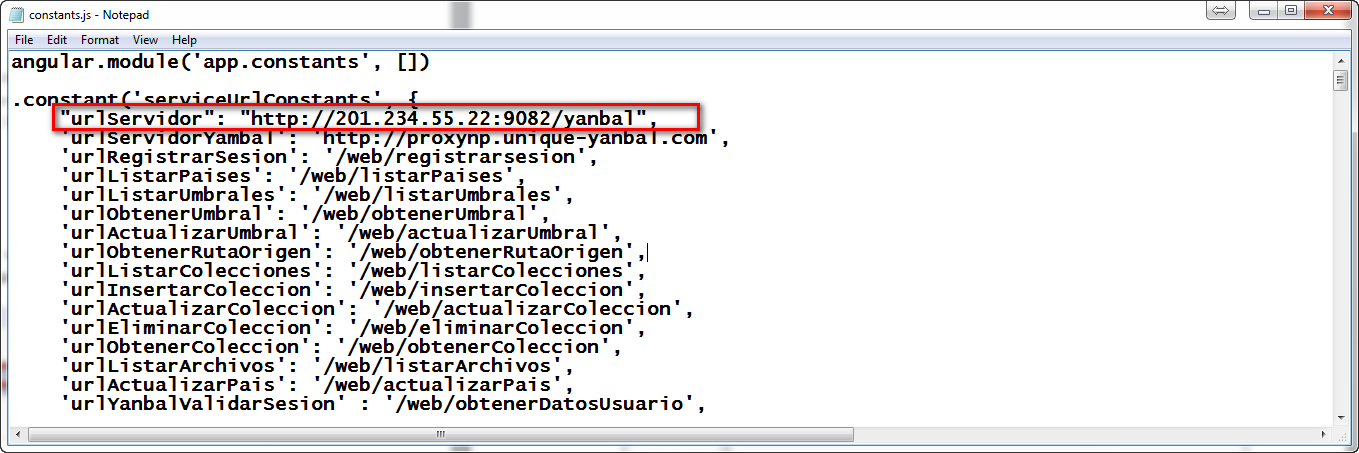
Para hacer cambios en la configuración del sitio web, basta con realizar las modificaciones en unos archivos.

Los archivos a modificar son los siguientes:

* **constants.js**
* **actualizar-archivo-modal.html**
* **insertar-archivo-modal.html**

Abrimos el archivo war con una herramienta zip siguiendo los pasos del [punto 8.1](#_Configuración_de_propiedades), luego nos ubicamos dentro de la ruta “**components\general\utils**” y abrimos el “**constants.js**” del mismo modo que abrimos los properties en el [punto 8.1](#_Configuración_de_propiedades)

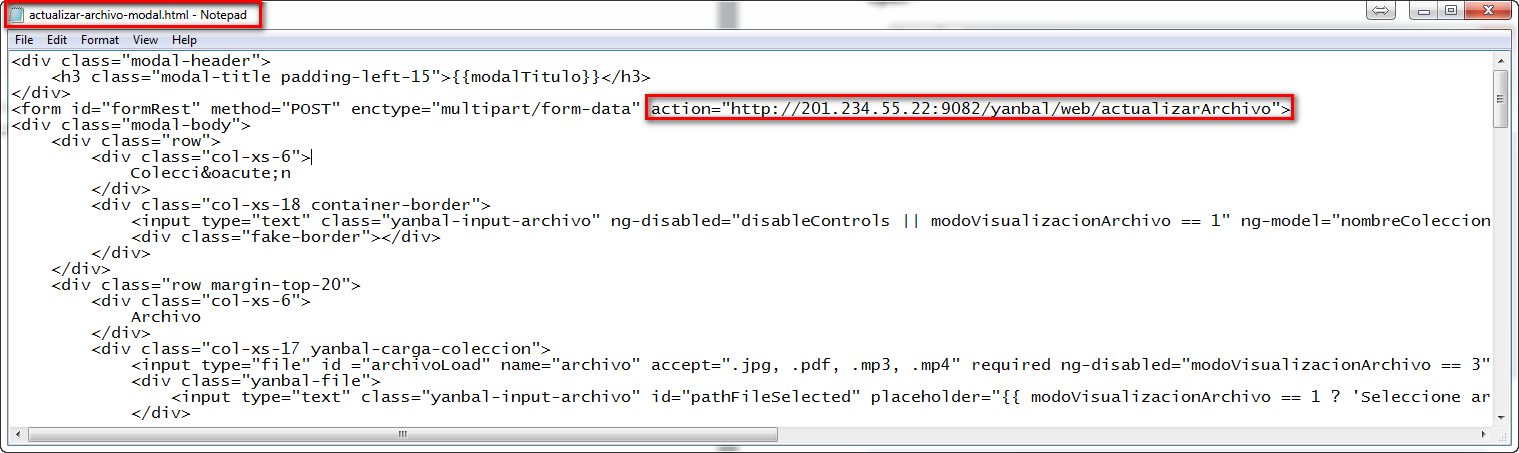
Sobre el editor de texto ubicamos el atributo "**urlServidor**" y lo modificamos con el valor del url donde se encuentra desplegado nuestro servicio web. Luego de modificar esta constante guardar los cambios y cerrar el editor de texto.



Luego buscamos los archivos “**insertar-archivo-modal.html**” y “**actualizar-archivo-modal.html**” en la ruta “**\components\coleccion\views**” y los abrimos del mismo modo que el archivo “**constants.js**”.

Ubicamos la línea donde se encuentra el tag “**action=**” y le asignamos el url donde se encuentra nuestro servicio backend, debería ser el mismo servidor al cual el archivo “**constants.js**” fue configurado.





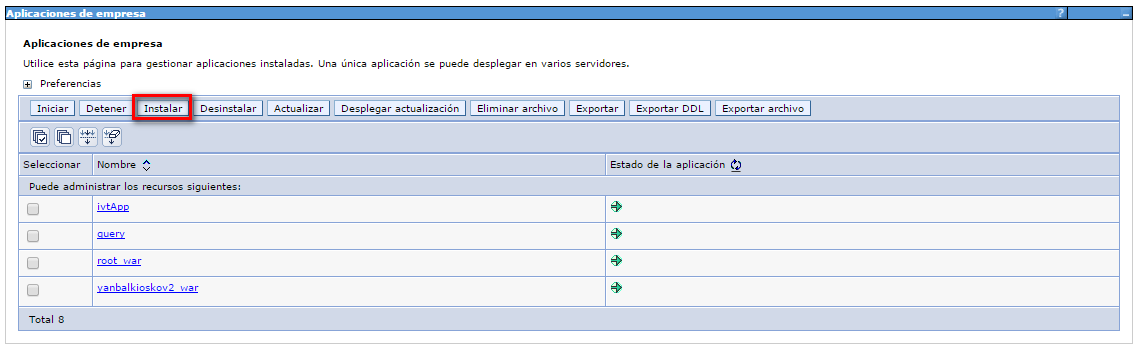
Luego de hacer las modificaciones en estos archivos guardamos los cambios como en los casos anteriores.

# Instalación del sitio web.

Seleccionamos la opción **“Aplicaciones” 🡪 “Tipo de Aplicación”🡪”Aplicaciones de empresa de WebSphere”** como se muestra en la imagen siguiente:



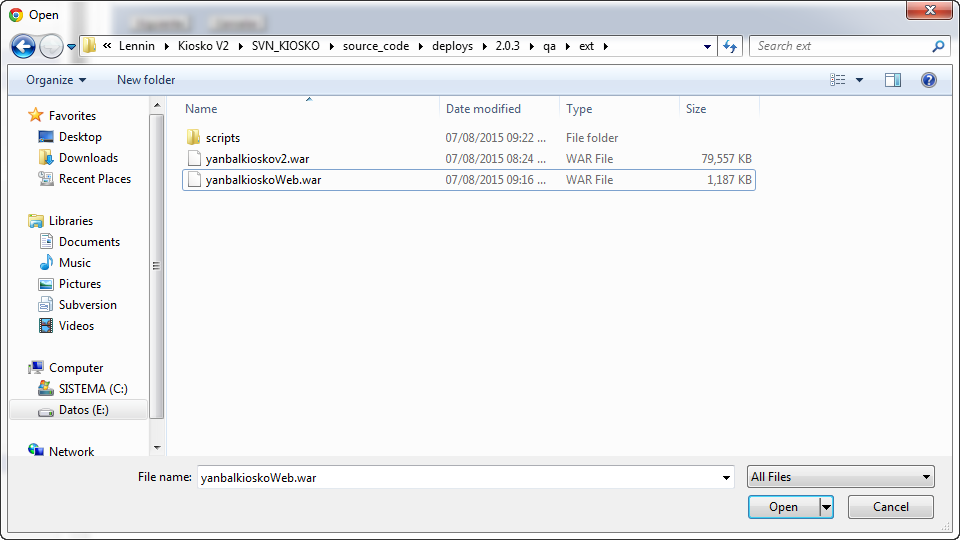
Nos mostrara la siguiente pantalla, en esta pantalla presionamos el botón instalar. (Si ya tenemos una instalación previa de nuestro WebSite, será necesario desinstalarlo, para ello seguiremos los pasos descritos mas abajo siguiendo el [siguiente enlace](#_Desinstalación_del_sitio)).



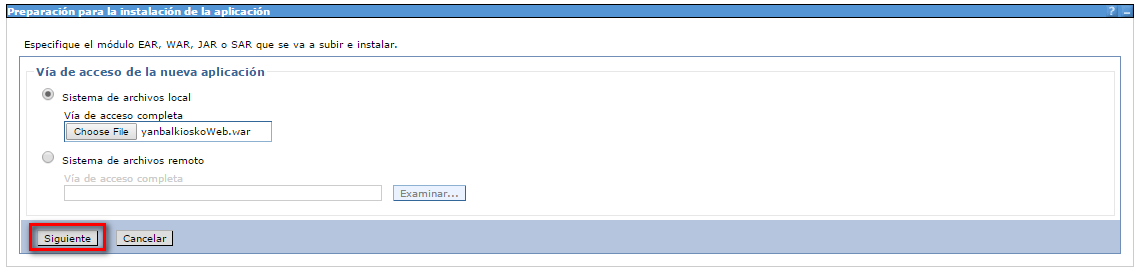
Nos presentara la siguiente pantalla donde seleccionaremos nuestro “**war**” o “**ear**” del sistema, seleccionamos la opción “**Choose File**”.

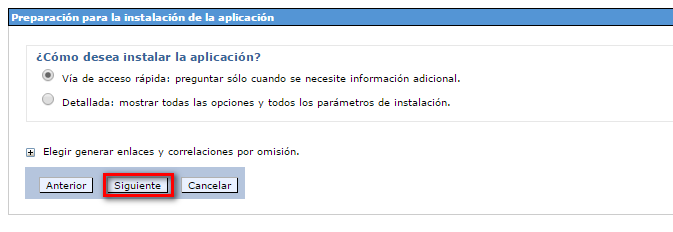


Nos presentara la siguiente pantalla del browser donde buscaremos nuestro archivo “**war**” o “**ear**” que deseamos desplegar, lo seleccionamos y presionamos el botón “**Open**”.

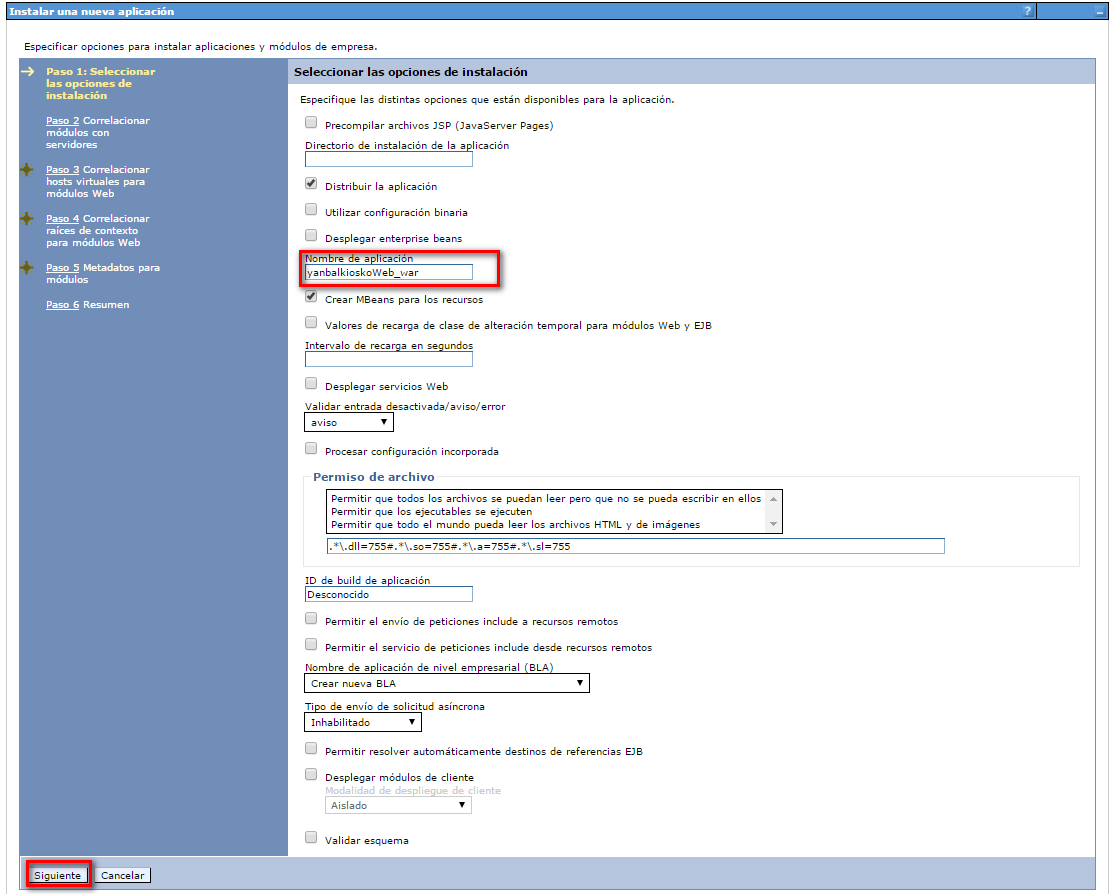


Nos mostrara la siguiente pantalla, seleccionamos el botón “**Siguiente**”.



Nos mostrara la siguiente pantalla, seleccionamos el botón “**Siguiente**”.

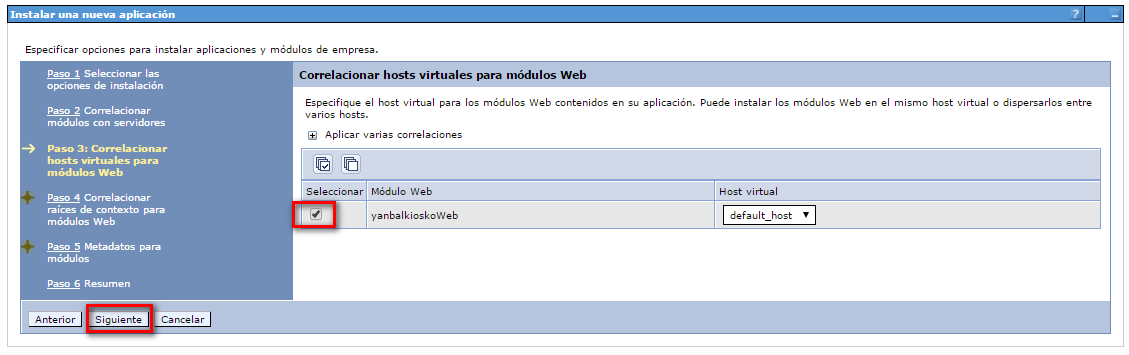
Se nos presentara la siguiente ventana, en ella podemos cambiar el nombre del Aplicativo en el campo “**Nombre de la aplicación**”, en el despliegue de ejemplo lo dejaremos con su nombre original por defecto y presionamos el botón “**Siguiente**”.



Se nos presenta la siguiente ventana, seleccionamos y activamos la casilla de verificación y presionamos el botón “**Siguiente**”.



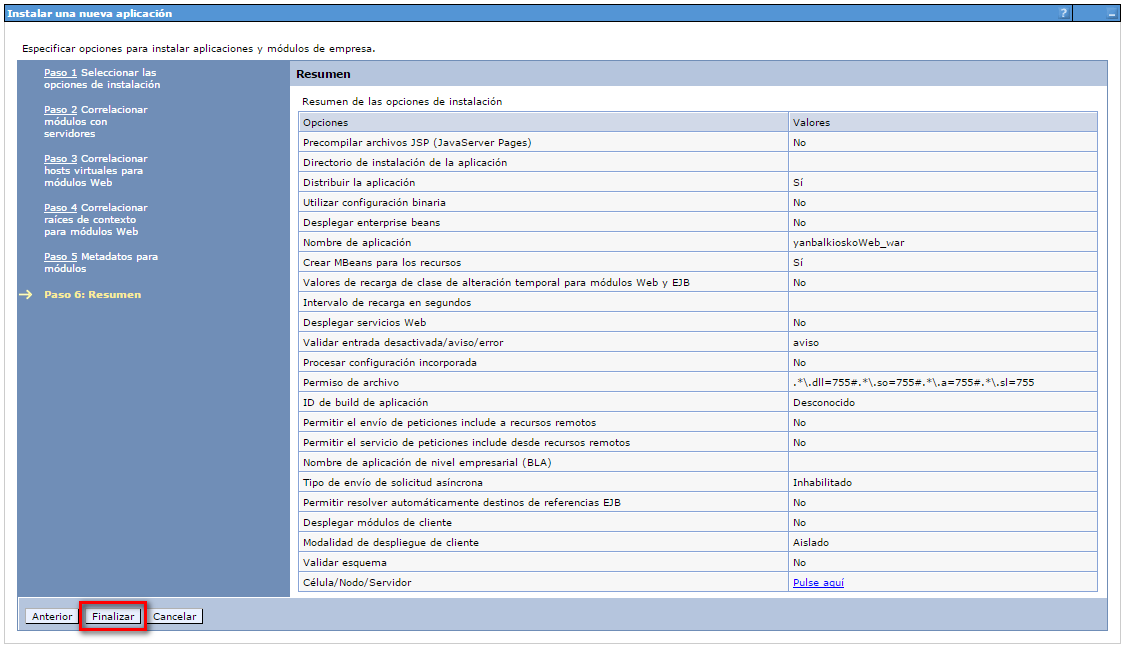
Se nos presenta la siguiente ventana, seleccionamos y activamos la casilla de verificación y presionamos el botón “**Siguiente**”.



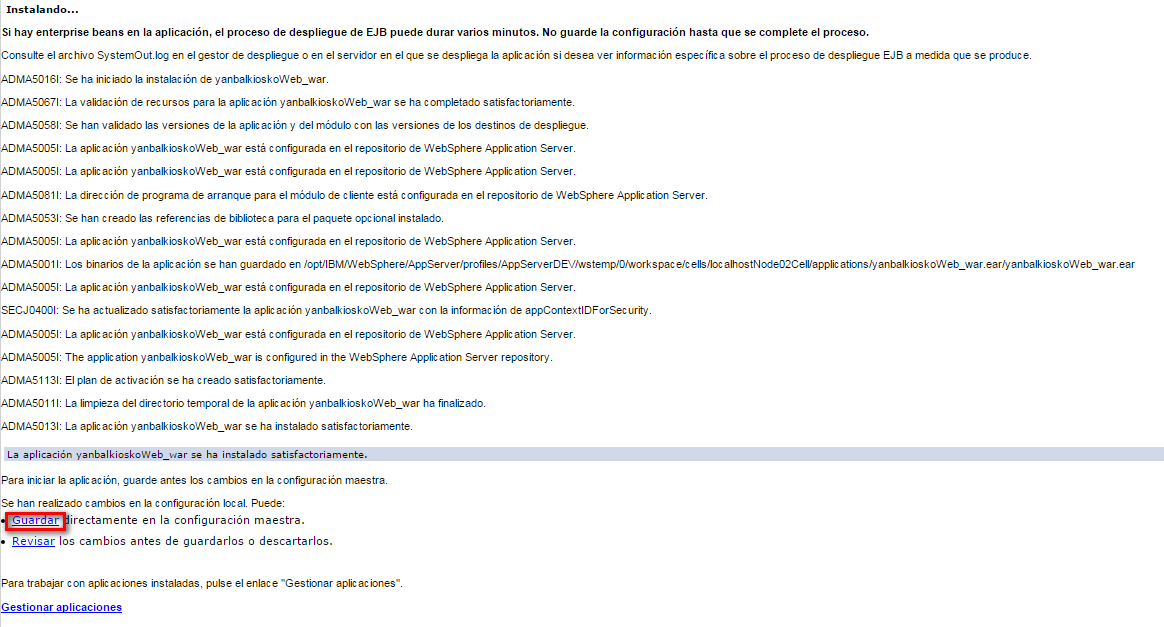
Se nos presenta la siguiente ventana, ingresamos un nombre para el contexto, en el ejemplo asignaremos el nombre de “**/site**” y presionamos el botón “**Siguiente**”.



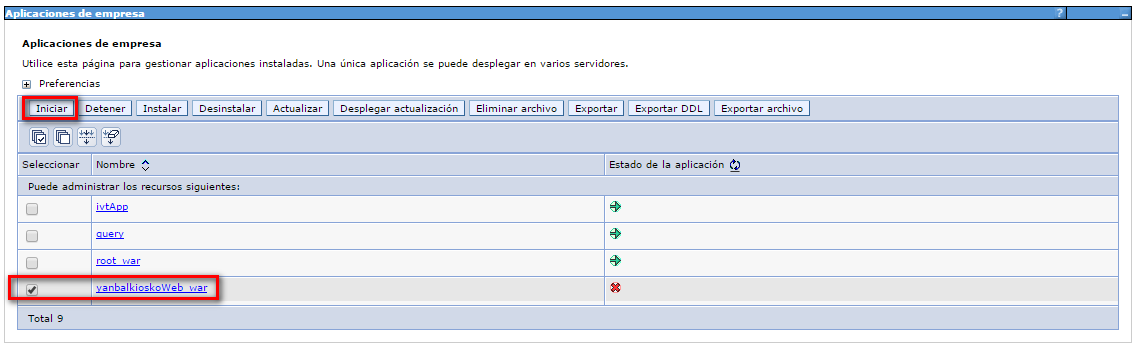
Nos muestra una pantalla con el resumen de los atributos del despliegue, presionamos el botón “**Finalizar**”.



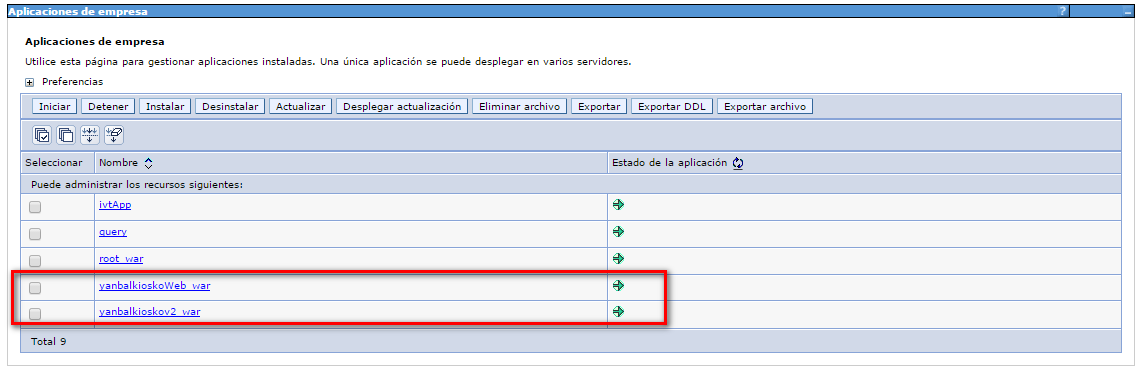
Nos muestra la pantalla con la información del proceso de instalación, cuando finaliza presionamos la opción “**Guardar**” para confirmar la instalación.



Nos muestra el listado de aplicaciones instaladas en nuestro servidor, marcamos la casilla de verificación de nuestro sistema y presionamos el botón “**Iniciar**”.



Luego de unos instantes veremos nuestro sistema ya iniciado.

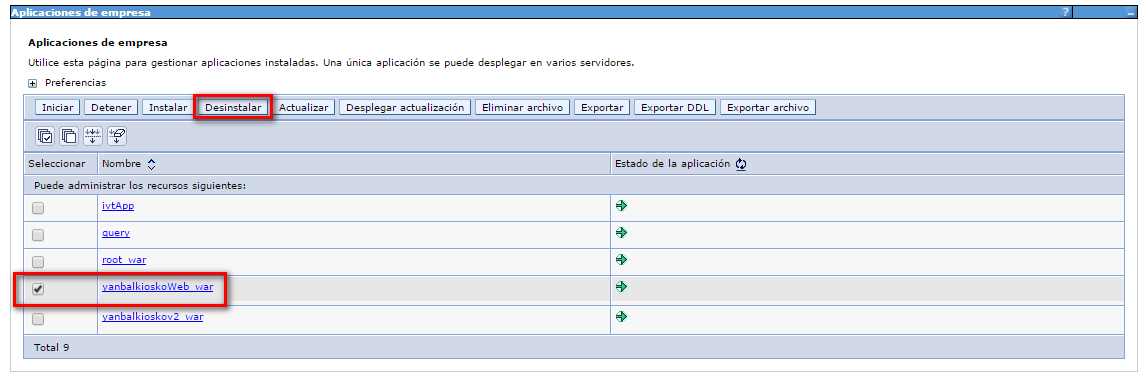


# Desinstalación del sitio web.

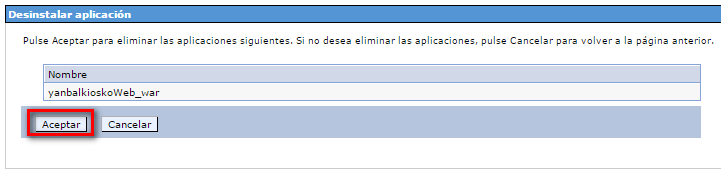
Seleccionamos la opción **“Aplicaciones”🡪 “Tipo de Aplicación”🡪”Aplicaciones de empresa de WebSphere”** como se muestra en la imagen siguiente:



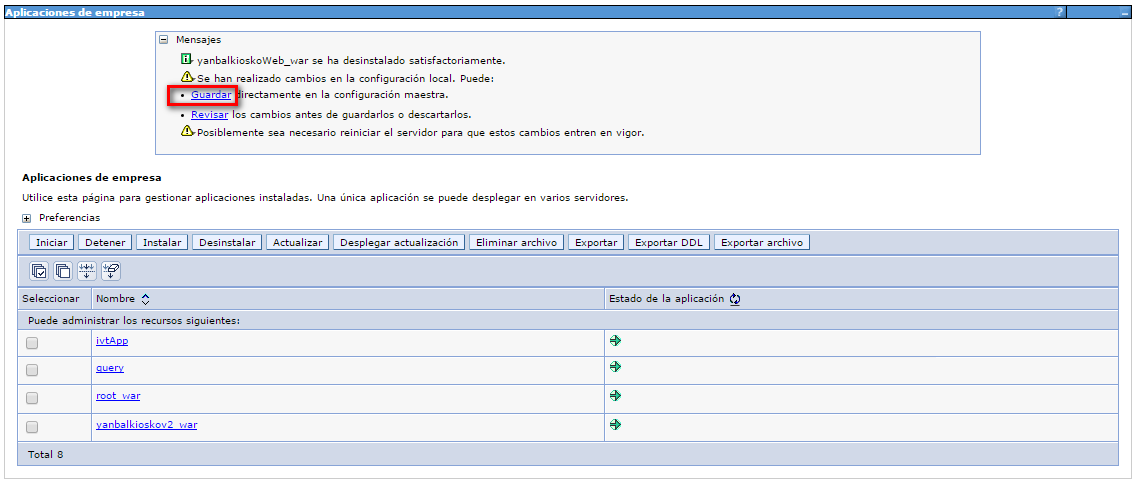
Nos mostrara la pantalla con la lista de aplicaciones instaladas, ubicamos nuestra aplicación y la seleccionamos activando la casilla de verificación y luego presionamos el botón “**Desinstalar**” como se muestra en la imagen.



Nos mostrara la pantalla siguiente antes de continuar con la desinstalación del sitio web, presionamos el botón “**Aceptar**”.



Luego pedirá confirmar los cambios, seleccionamos la opción “**Guardar**” para confirmar la desinstalación del sitio web.



Luego de confirmar la desinstalación, veremos la pantalla de listado de aplicaciones sin nuestro sitio web, como se muestra en la pantalla siguiente.

