**Manual para el desarrollador**

**Ejercicio práctico donde se describe cómo diseñar, exponer y consumir un servicio web de datos.**



**Autora:**

Ing. Gloria Raquel Leyva Jerez

**Contenido**

[**Introducción** 3](#_Toc498035664)

[**Instalación y configuración de las herramientas** 4](#_Toc498035665)

[1. Creación de la base de datos en PostgreSQL 4](#_Toc498035666)

[2. Desplegar el Servidor de Servicios de Datos (DSS) 4](#_Toc498035667)

[3. Instalar el plugin “*wso2-developer-studio\_3.8.0*” al Eclipse 5](#_Toc498035668)

[**Creación y despliegue del servicio** 7](#_Toc498035669)

[1. Creación de servicios en el DSS 8](#_Toc498035670)

[1.1 Adicionar los datasources 8](#_Toc498035671)

[1.2 Adicionar las consultas 11](#_Toc498035672)

[1.3 Adicionar las operaciones 14](#_Toc498035673)

[1.4 Exponer el servicio 16](#_Toc498035674)

[1.5 Probar el servicio desplegado. 17](#_Toc498035675)

[2. Generar servicios en el DSS 18](#_Toc498035676)

[3. Diseñar el servicio en el IDE Eclipse 20](#_Toc498035677)

[3.1 Adicionar las consultas 22](#_Toc498035678)

[3.2 Adicionar las operaciones 24](#_Toc498035679)

[3.3 Desplegar el servicio desde el IDE 25](#_Toc498035680)

[3.4 Desplegar el servicio desde la plataforma DSS 27](#_Toc498035681)

[**Consumir el servicio por otras aplicaciones** 27](#_Toc498035682)

[1. Generar un cliente de servicio web Axis2 28](#_Toc498035683)

[2. Consumir el servicio 31](#_Toc498035684)

[**Conclusiones** 34](#_Toc498035685)

# **Introducción**

La creciente informatización de los procesos en los diversos sectores de la sociedad ha propiciado el aumento y constante manejo de la información, promoviendo la utilización de Sistemas de Información (SI). Los SI operan sobre una colección de datos estructurados de acuerdo con las necesidades de una empresa, recopilan, elaboran y distribuyen la información que las empresas necesitan. Este proceso de distribución de la información es común en diversos sistemas, es decir, múltiples sistemas necesitan actualizar sus datos en las Bases de Datos (BD) donde pueden compartir la misma información. Por tanto, al crear un servicio de datos se facilita que pueda ser utilizado por múltiples aplicaciones o SI, sin tener que invertir tiempo y esfuerzo en desarrollar estas funcionalidades en cada SI. La implementación de servicios web aporta interoperabilidad entre aplicaciones de software, independientemente de sus propiedades o de las plataformas sobre las que se instalen; y permiten que servicios y software de diferentes compañías ubicadas en diferentes lugares geográficos puedan ser combinados fácilmente, para proveer servicios integrados.

En el presente trabajo se explica primeramente la instalación y configuración de las herramientas, y luego se describe de diversas formas cómo se diseña, expone y consume un servicio de datos, mediante un ejemplo práctico de operaciones sobre una tabla Paciente e Historia Clínica de una BD Hospital. Para explicar el proceso de cómo las aplicaciones pueden consumir el servicio se desarrolla una aplicación que gestiona los datos de los pacientes y las historias clínicas, ejemplificando algunas consultas DML[[1]](#footnote-1) y DDL[[2]](#footnote-2) que puedan realizarse sobre la BD Hospital.

A continuación, se muestra el diagrama de despliegue para visualizar la distribución de los componentes físicos que intervienen en el ejemplo práctico que se explicará en el desarrollo con un manual para el desarrollador.

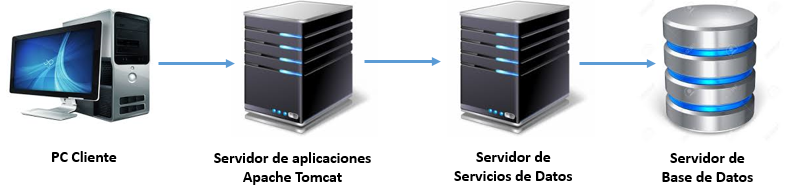


Figura 1 Diagrama de despliegue

Fuente: elaboración propia.

Los usuarios de las PC Cliente interactúan con la aplicación de gestión de información que se encuentra en el Servidor de Aplicaciones, donde pueden listar, insertar, eliminar y actualizar datos mediante el consumo de los servicios expuestos en el Servidor de Servicios de Datos, el cual a su vez ejecuta las consultas en las Base de Datos (BD). El servicio expuesto puede ser consumido por otras aplicaciones.

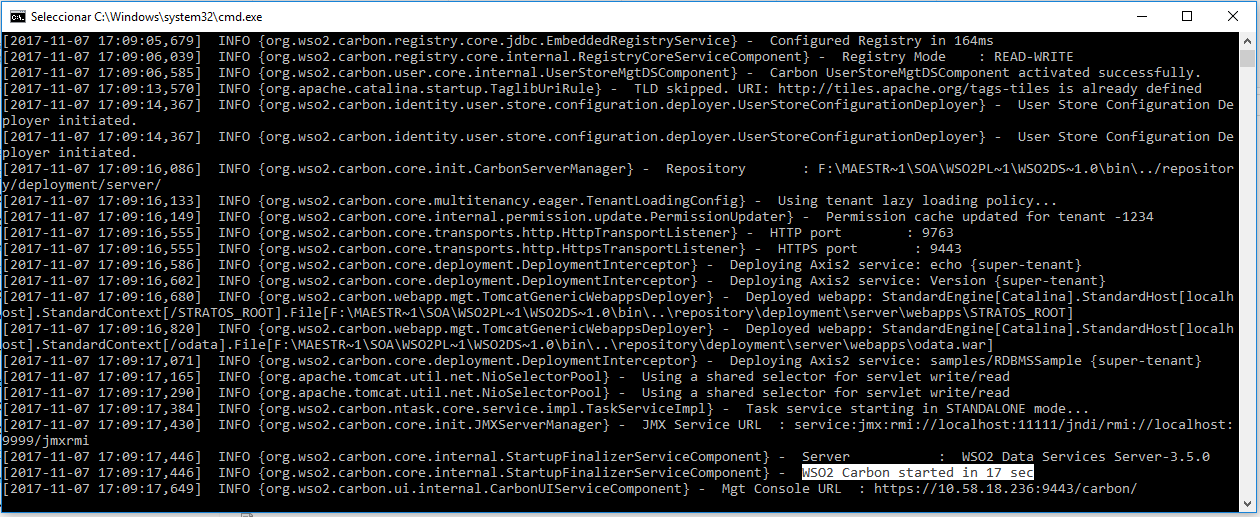
# **Instalación y configuración de las herramientas**

## **Creación de la base de datos en PostgreSQL**

La base de datos del ejemplo que se explica en el presente trabajo se encuentra en el directorio “.\Herramientas”con el nombre “hospital”. Debe instalar el motor de base de datos PostgreSQL y mediante la opción “*Restore*” adicionar “*hospital.backup*”.

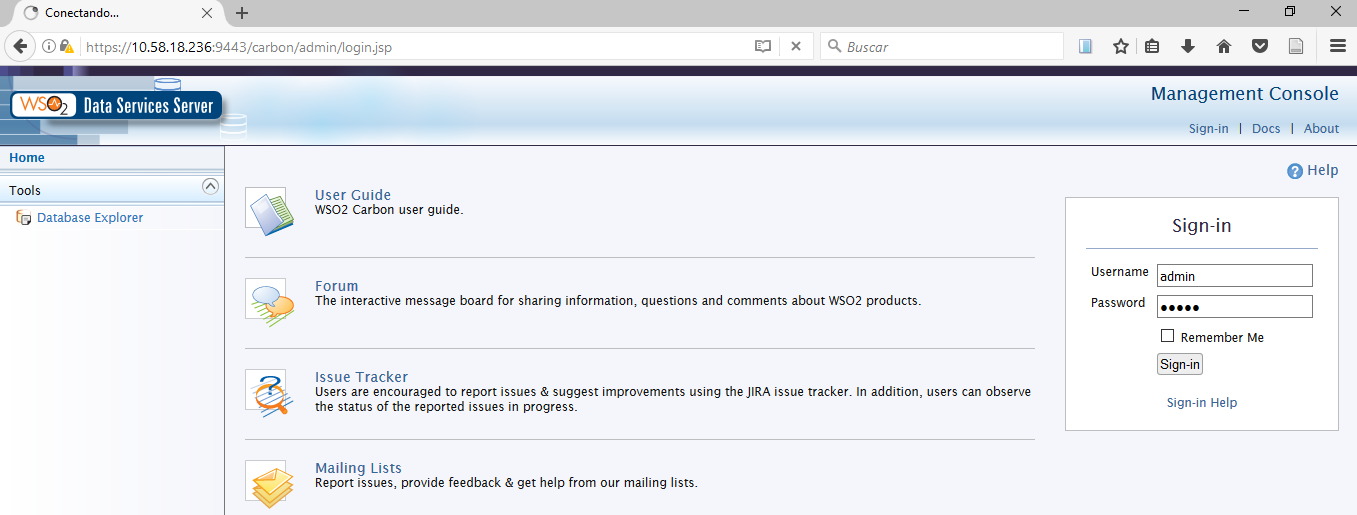
## **Desplegar el Servidor de Servicios de Datos (DSS)**

Para desplegar el servidor sobre WINDOWS se ejecuta el archivo “*wso2server.bat*”, y sobre LINUX “*wso2server.sh*”, que se encuentran en el directorio “.\Herramientas\wso2 plataform\wso2dss-3.5.0\bin”.



A- Cuando se muestre en consola que el servidor ha iniciado, se debe acceder al navegador y abrir el enlace provisto por *Mgt Console URL*.

B- Las credenciales para autenticarse por defecto en el DSS son usuario y contraseña “*admin*”.



**BA**

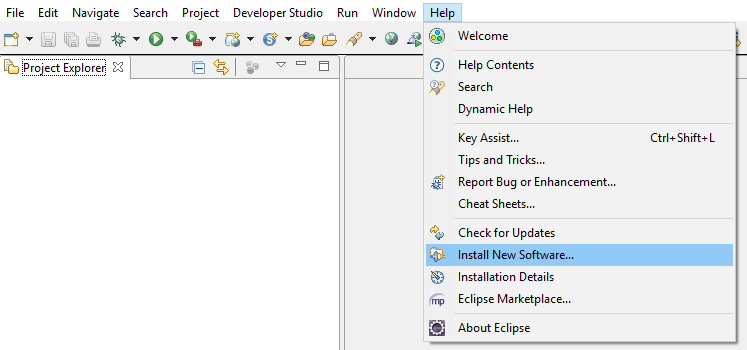
**AA**

## **Instalar el plugin “*wso2-developer-studio\_3.8.0*” al Eclipse**

Para implementar el servicio en el Eclipse se debe instalar el plugin “*wso2-developer-studio\_3.8.0*” que se encuentra en el directorio “*\Herramientas\wso2 plataform”*. Este plugin brinda la facilidad de crear los diferentes tipos de proyecto que brinda la plataforma del wso2.

A- Acceder a la opción “*Help”*.

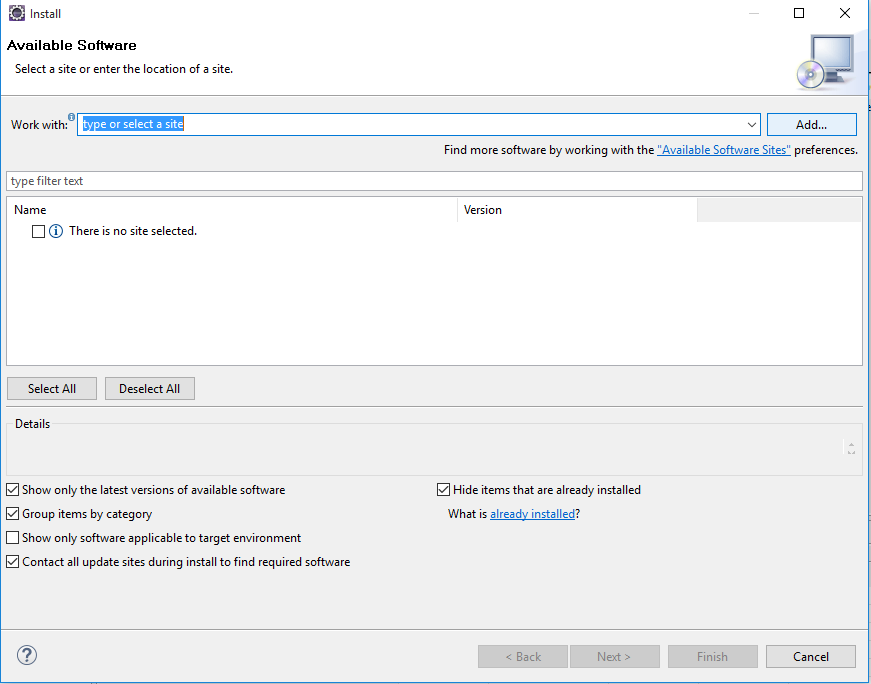
B- Para instalar el plugin debe elegir “*Install new software”*.



**B**

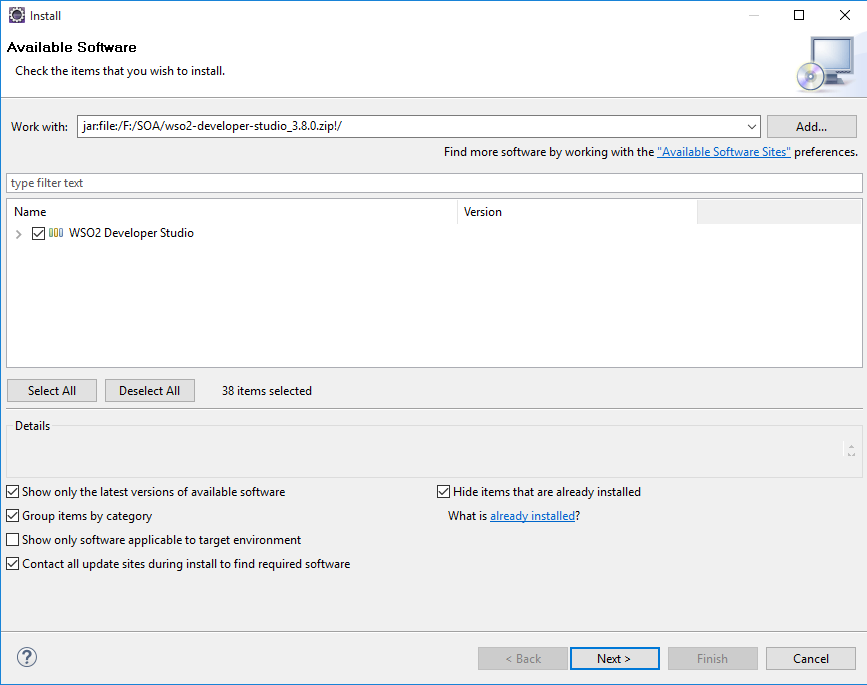
**AA**

C- Especificar la ruta donde se encuentra el plugin.



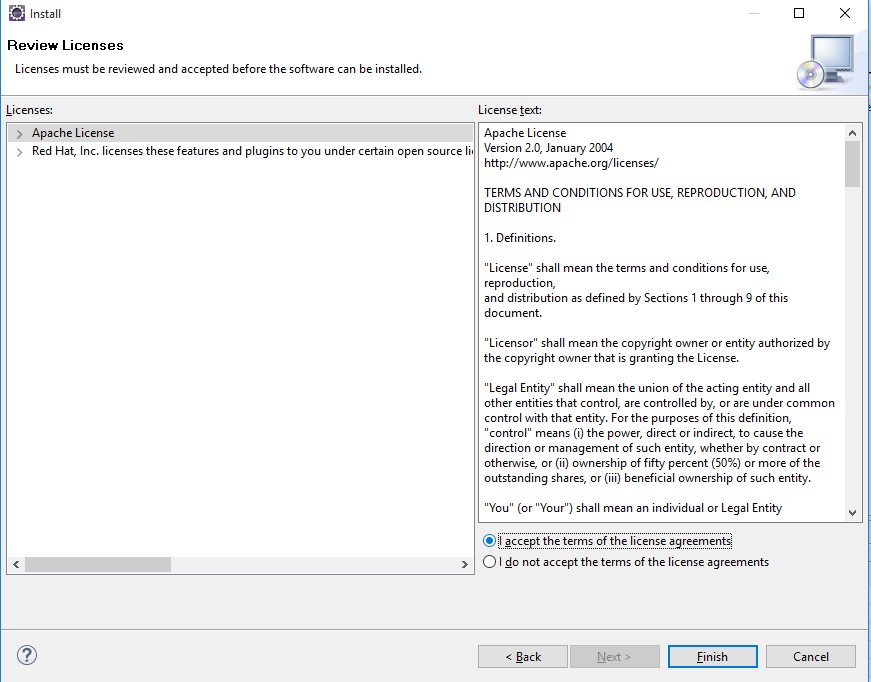
**C**

D- Marcar la opción indicada



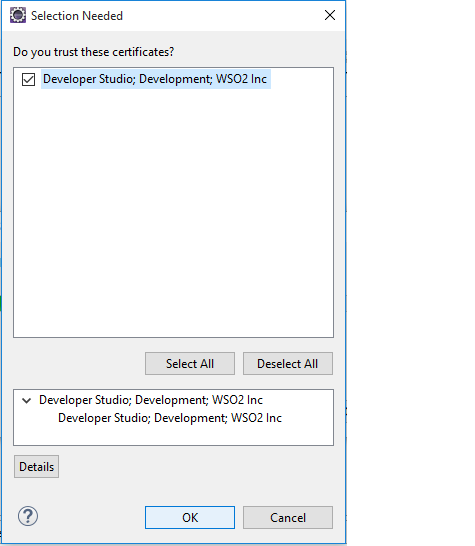
**D**

E- Aceptar los términos de licencia.



**E**

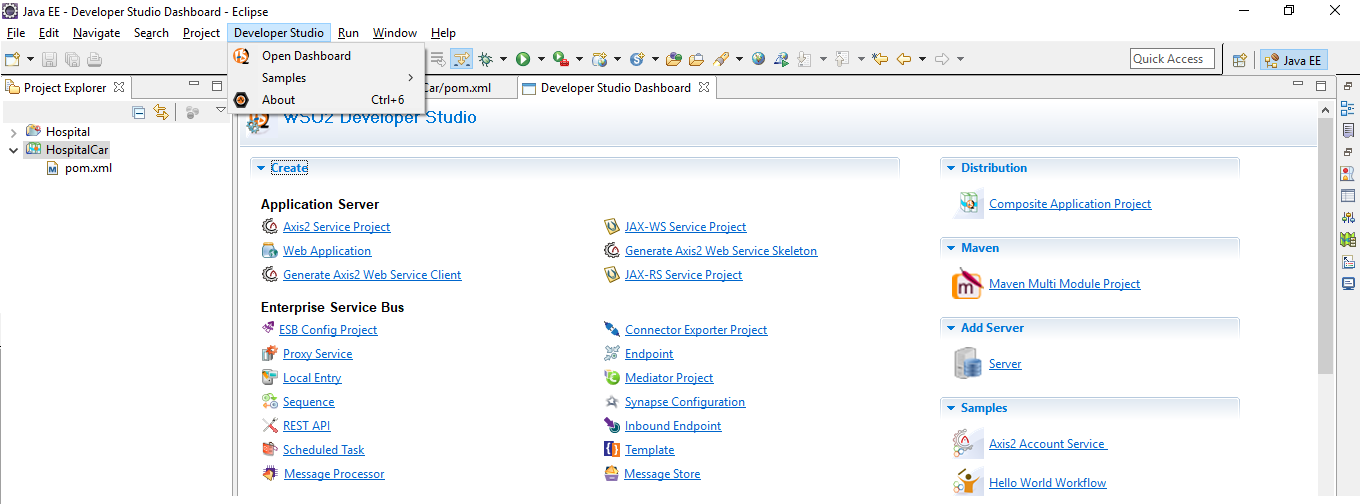
F- Marcar el certificado.



**F**

Al culminar el proceso de instalación del certificado se reinicia el Eclipse y se puede observar como en el menú se muestra la opción “*Developer Studio*” (G)

**G**



# **Creación y despliegue del servicio**

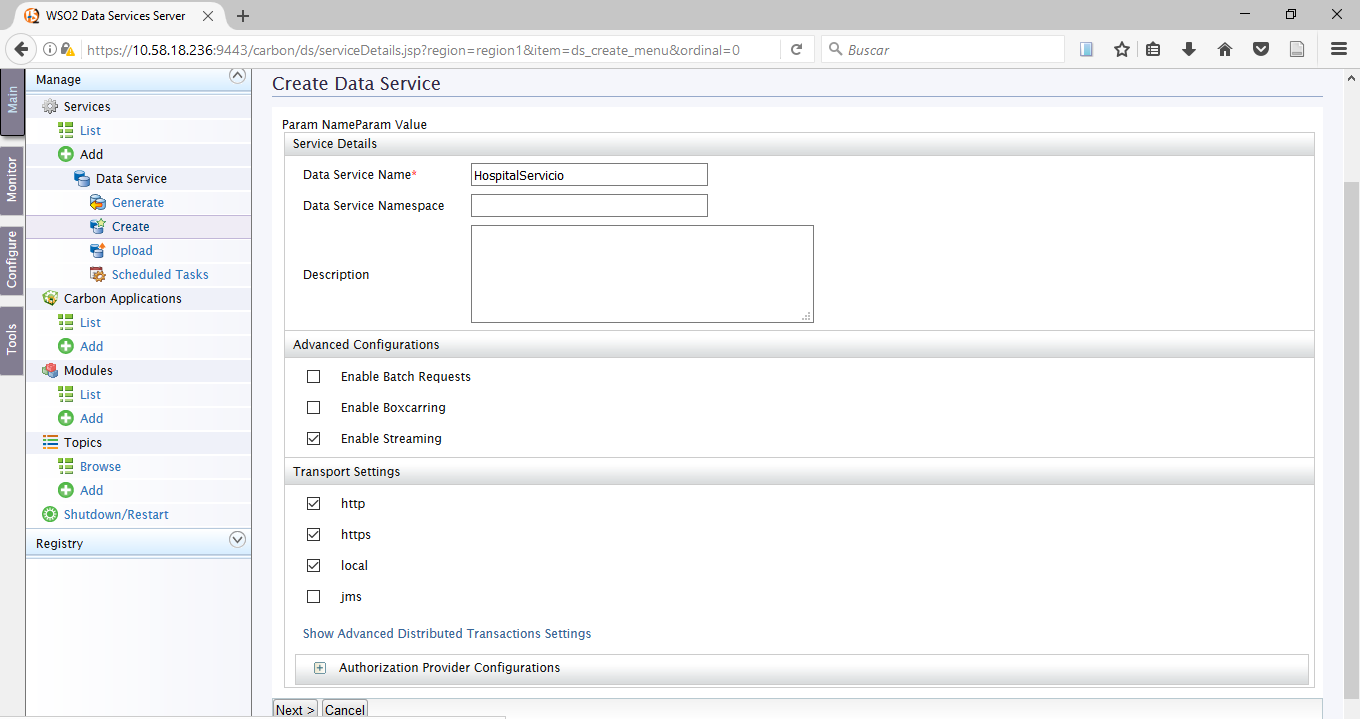
Los servicios pueden ser diseñados en un IDE[[3]](#footnote-3) y ser expuesto con el DSS. También con el DSS es posible crear, generar o subir servicios diseñados en un IDE de manera que al exponerlos puedan ser consumidos por cualquier aplicación.

A continuación, se describe cada una de estas formas a través de las cuales los servicios son expuestos para ser consumidos por aplicaciones.

## **Creación de servicios en el DSS**

A- Mediante la opción “*Create*” se inicia la creación del servicio.

B- Se especifica el nombre del servicio.

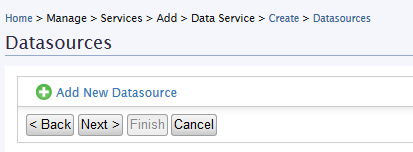


**B**

**A**

### **1.1 Adicionar los datasources**

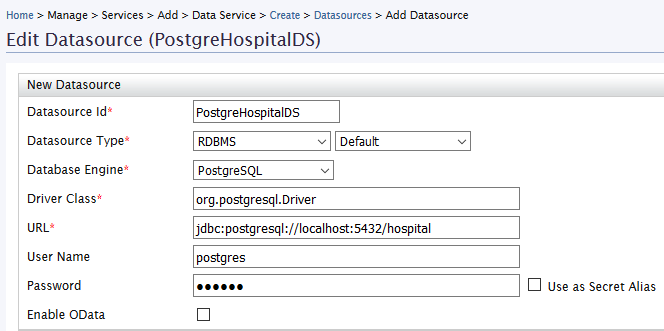
A- Adicionar el *Datasources*. El *Datasource* es donde se encuentra la configuración de la base de datos.



**A**

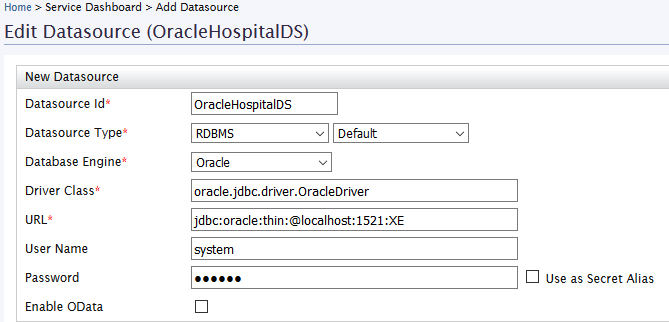
Se especifican los detalles de conexión a la base de datos. A continuación, se especifican los parámetros de conexión para los motores de base de datos PostgreSQL, Oracle, MySQL y Microsoft SQL Server.

B- Adicionar *Datasource* para PostgreSQL



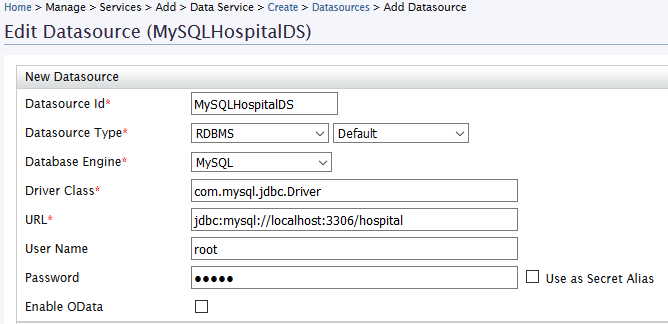
**B**

B- Adicionar *Datasource* para Oracle



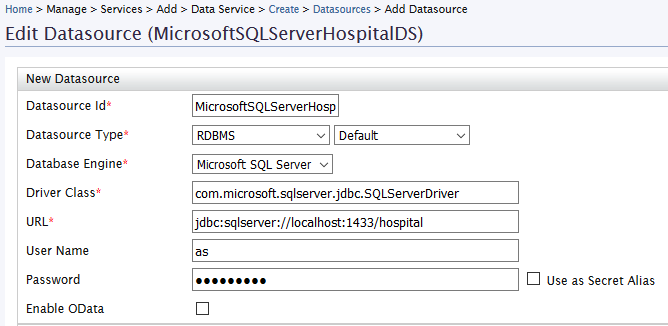
**B**

B- Adicionar Datasource para MySQL



**B**

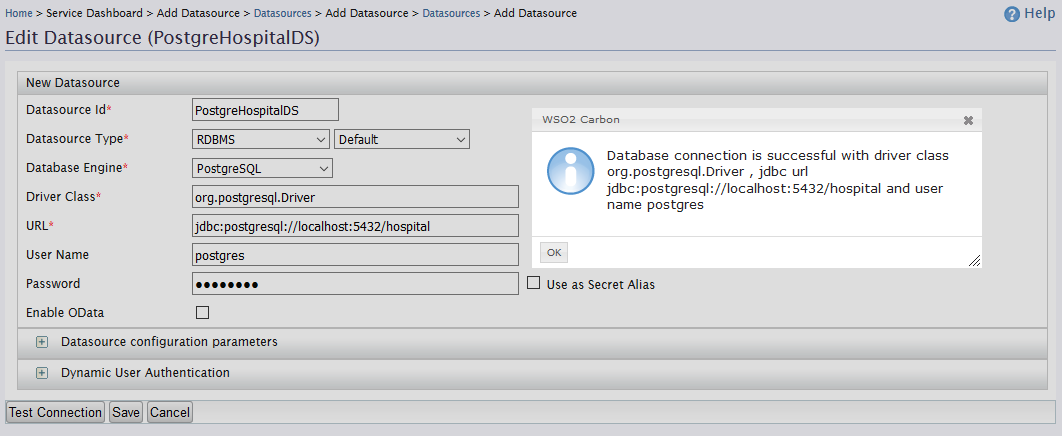
B- Adicionar *Datasource* para Microsoft SQLServer



**B**

C- Mediante la opción “*Test Connection*” se validan los parámetros establecidos

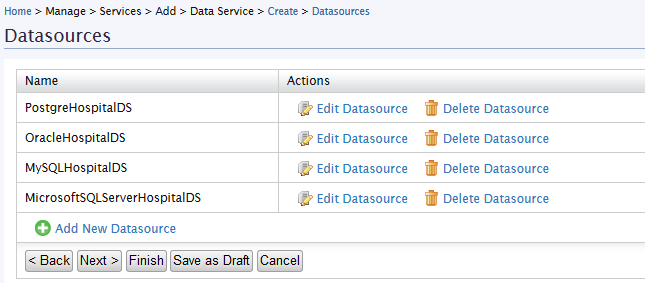
D- Se guardan los cambios mediante la opción “*Save*”.



**D**

**C**

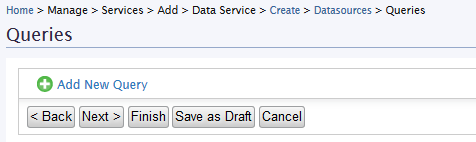
A continuación, se muestran todos los *datasources* adicionados.



Una vez adicionados todos los *datasources* se continúa mediante la opción “*Next*”.

### **1.2 Adicionar las consultas**

A- Adicionar las consultas que se realizarán sobre las bases de datos mediante la opción “*Add New Query*”. Se debe tener en cuenta que es necesario adicionar las consultas específicas para cada tipo de *datasource*.



**A**

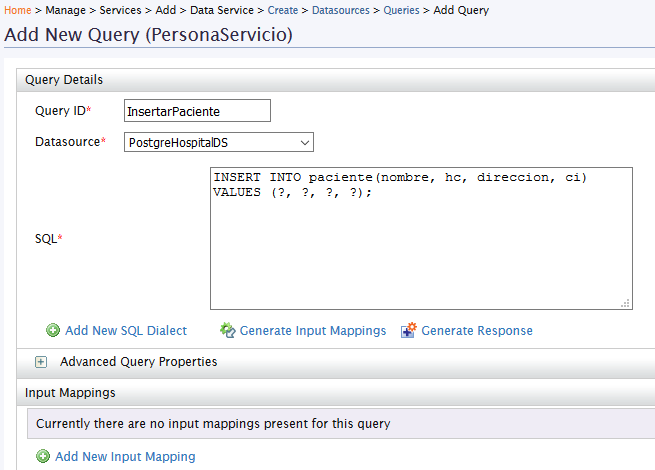
Se adicionan las consultas de *insert*, *update* y *delete* para realizar las acciones DML y también las consultas *create*, *alter* y *drop* para las acciones DDL

B- Se especifica un nombre para la acción de inserción.

C- Se selecciona el *datasource* de PostgreSQL.

D- Insertar el SQL de la consulta.

E- Esta consulta requiere cuatro parámetros de entrada los cuales se deben especificar mediante la opción “*[Add New Input Mapping](javascript:%20document.dataForm.submit();)*”.

Z

**E**

**C**

**B**

**DA**

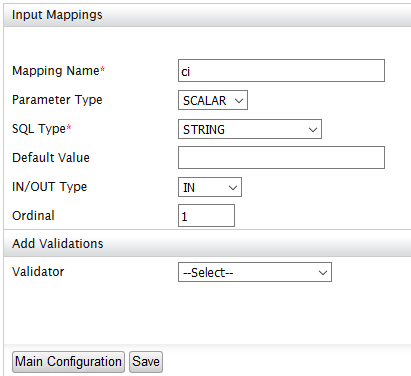
Al elegir la opción “*Add New Input Mapping*” se brinda la posibilidad de definir los parámetros de entrada de cada consulta. En el proceso se definir el parámetro de entrada del “*ci*” debe:

F- Especificar el nombre del parámetro de entrada.

G- Seleccionar tipo de datos.

H- Especificar la posición que ocupa en el *sql.*

I- Adicionar cada parámetro mediante la opción “*Add.*



**I**

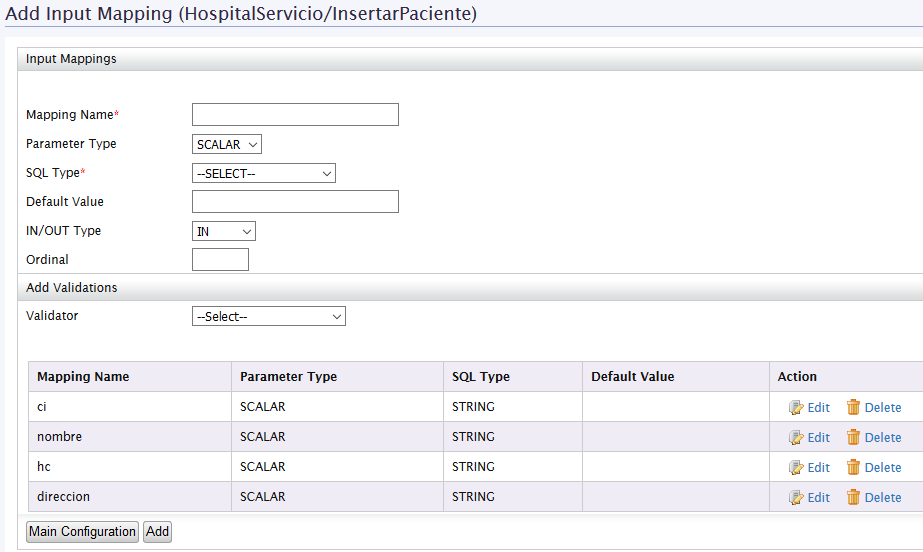
**H**

**G**

**F**

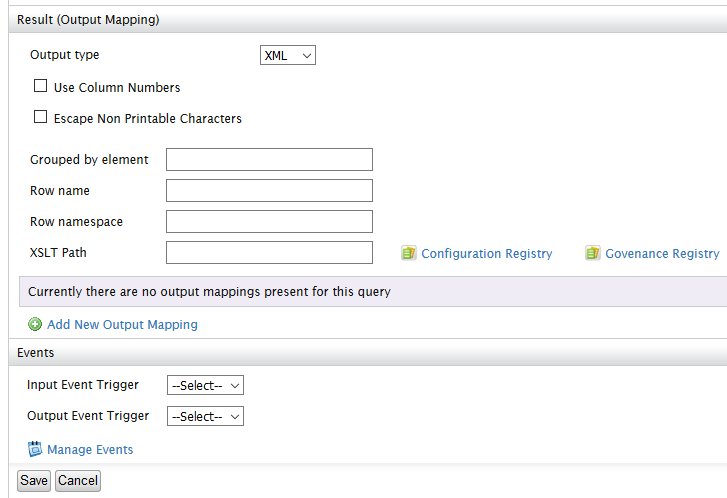
Como la tabla paciente en la acción de inserción tiene otros campos “*nombre*”, “*hc*” y “*dirección*”, se debe realizar el mismo procedimiento para especificar los restantes parámetros de entrada, quedando de la siguiente forma.

J- Al adicionar todos los parámetros se selecciona la opción “*Main Configuration*” para continuar con la especificación de la consulta.



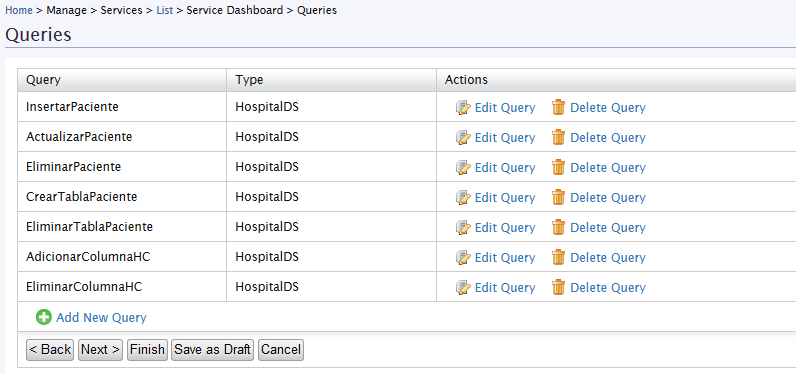
**J**

K- Para salvar la consulta se selecciona la opción “*Save*”.



**K**

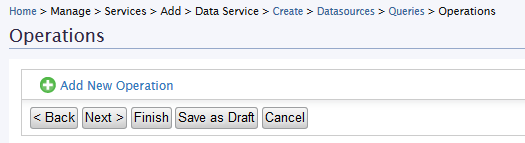
L- Una vez adicionadas todas las consultas se continúa mediante la opción “*Next*”.



**L**

### **1.3 Adicionar las operaciones**

A- Se deben adicionar las operaciones correspondientes a cada consulta adicionada anteriormente mediante la opción “*Add New Operation*”.



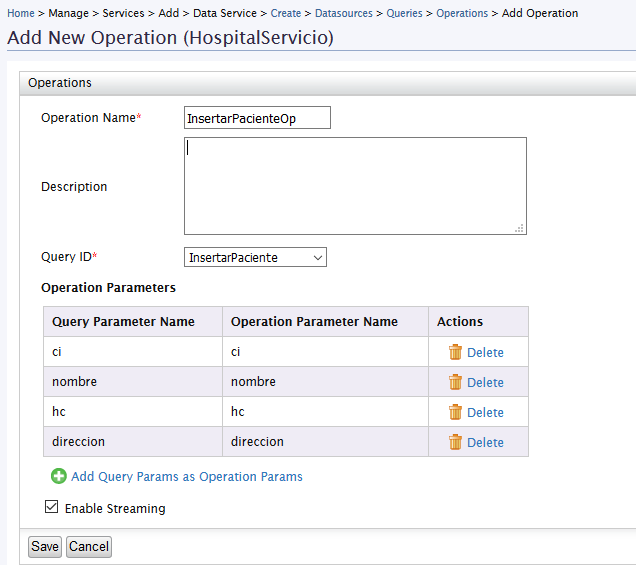
**A**

A continuación, se ejemplifica la creación de la operación encargada de ejecutar la consulta “*InsertarPaciente*”, adicionada anteriormente. Para lo cual debe:

B- Especificar el nombre de la operación.

C- Seleccionar la consulta a la cual estará asociada. Automáticamente se mostrarán todos los parámetros que contiene la consulta y cómo estos serán mapeados por la operación.

D- Se selecciona la opción “*Save*”, para adicionar la operación.

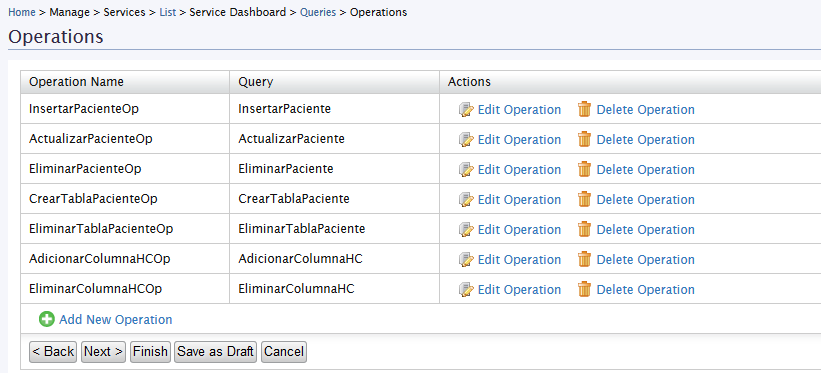


**D**

**C**

**B**

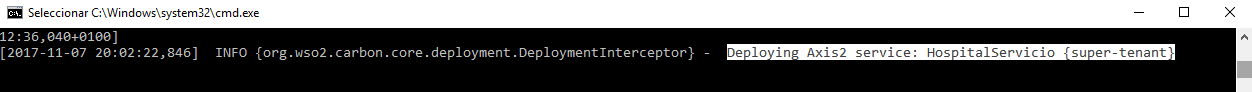
E- Cuando se adicionen todas las operaciones, se selecciona la opción “*Finish*” para terminar el proceso de creación del servicio de datos.



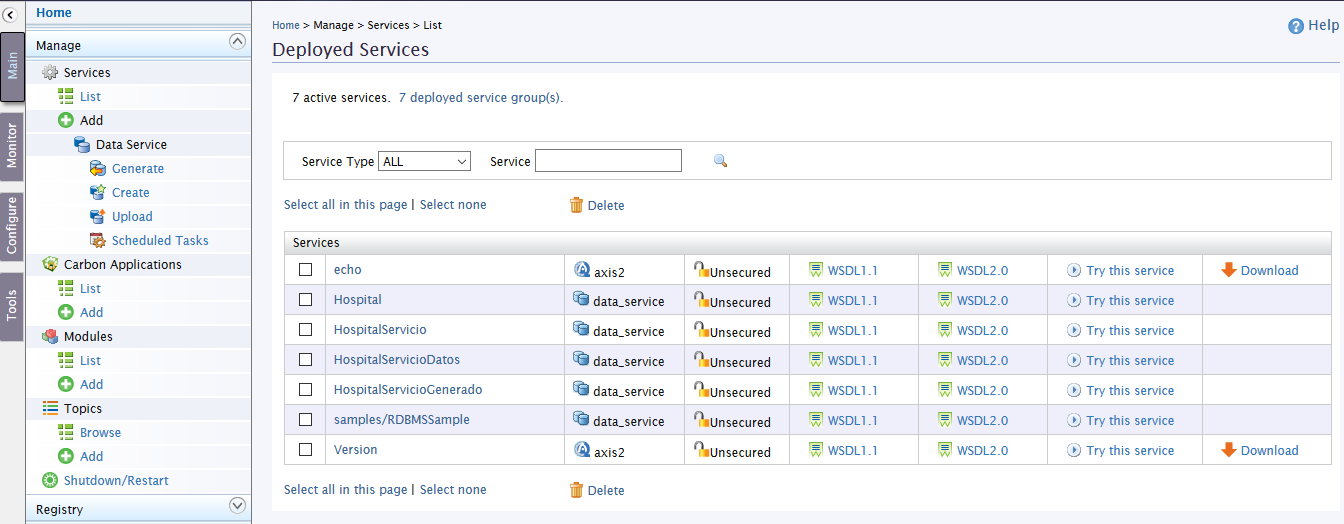
**E**

### **1.4 Exponer el servicio**

Se puede verificar en consola como el DSS expone el servicio cuando culmina el proceso anteriormente descrito.

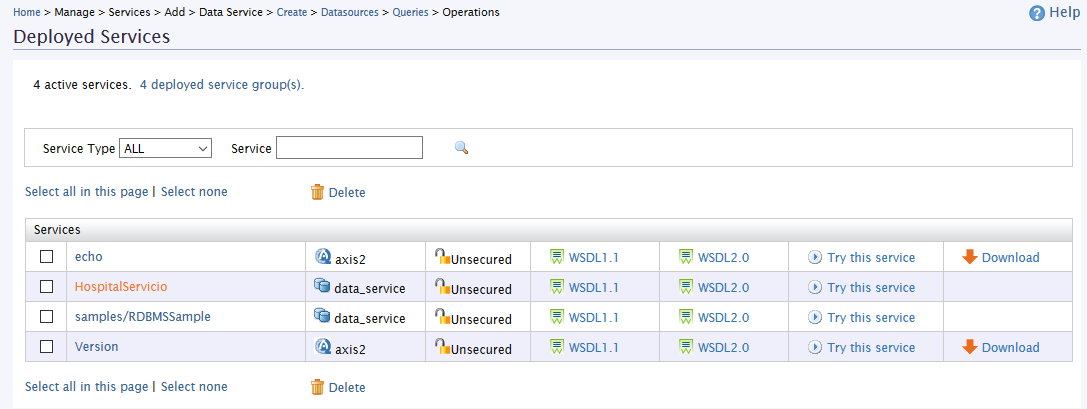


A- Cuando finalice, acceda al listado de servicios desplegados. El servicio se mostrará en el listado de servicios desplegados.



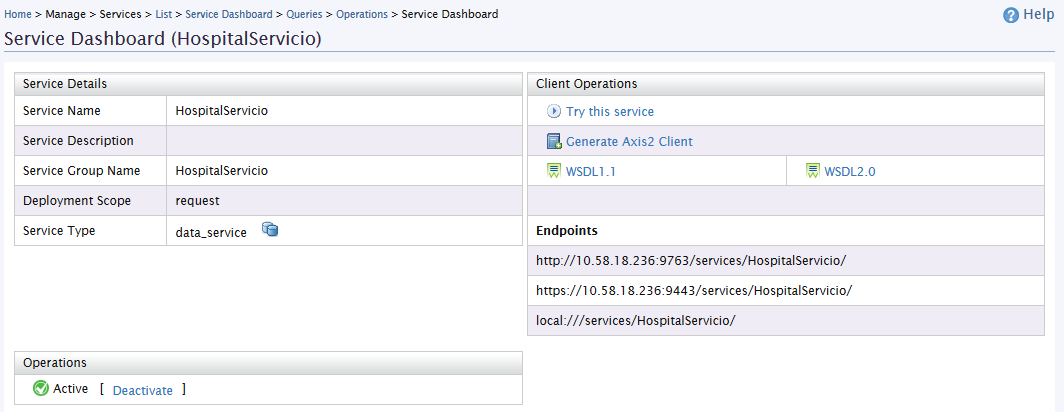
**A**

B- Seleccionar el servicio para observar los detalles.



**B**

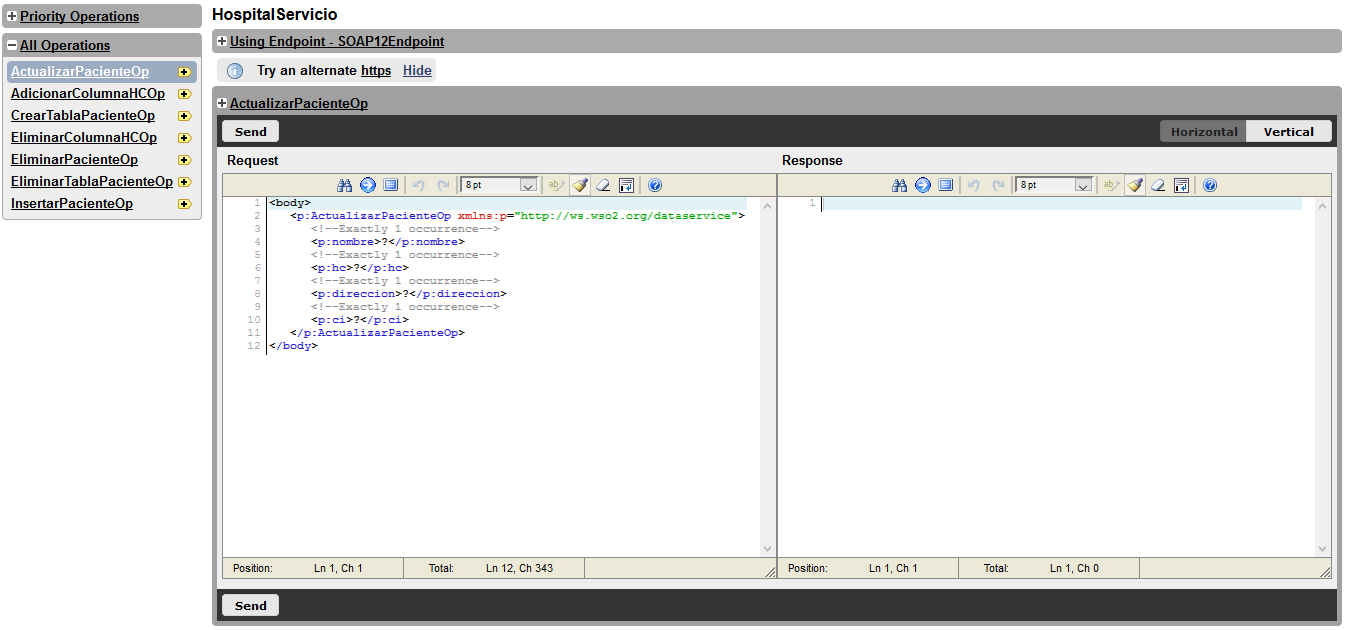
C- Para probar el funcionamiento del servicio se selecciona la opción “*Try this service*”.



**C**

D- Al seleccionar esta opción se abrirá otra pestaña donde se podrán observar las operaciones definidas durante la creación del servicio.

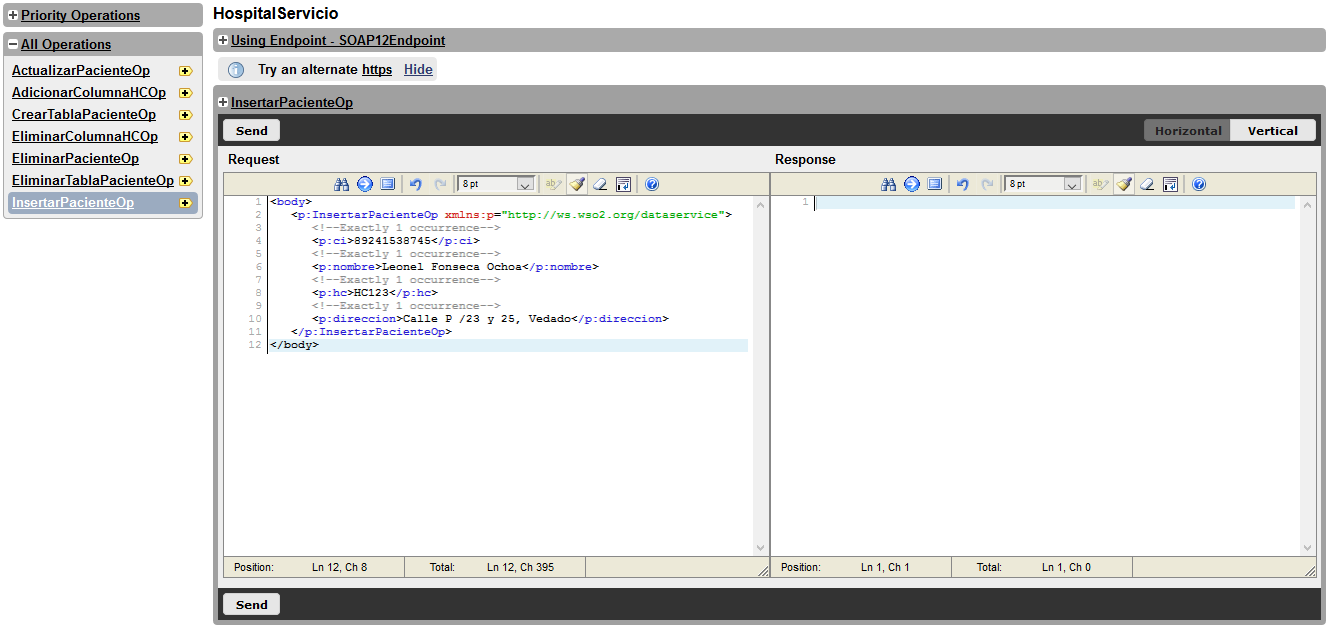
**D**



### **1.5 Probar el servicio desplegado.**

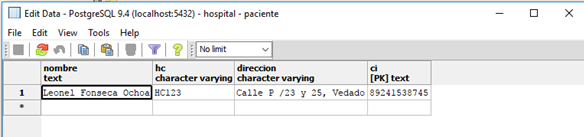
A continuación, se muestra la vista del servicio desplegado donde se prueba la operación “*InsertarPacienteOp*” que ejecuta la consulta de inserción del paciente con los datos especificados en cada parámetro.

A- Para ejecutar la operación se selecciona la opción “*Send*” luego de especificar los datos de entrada.



**A**

B- Se puede verificar en la base datos como el paciente ha sido insertado correctamente.

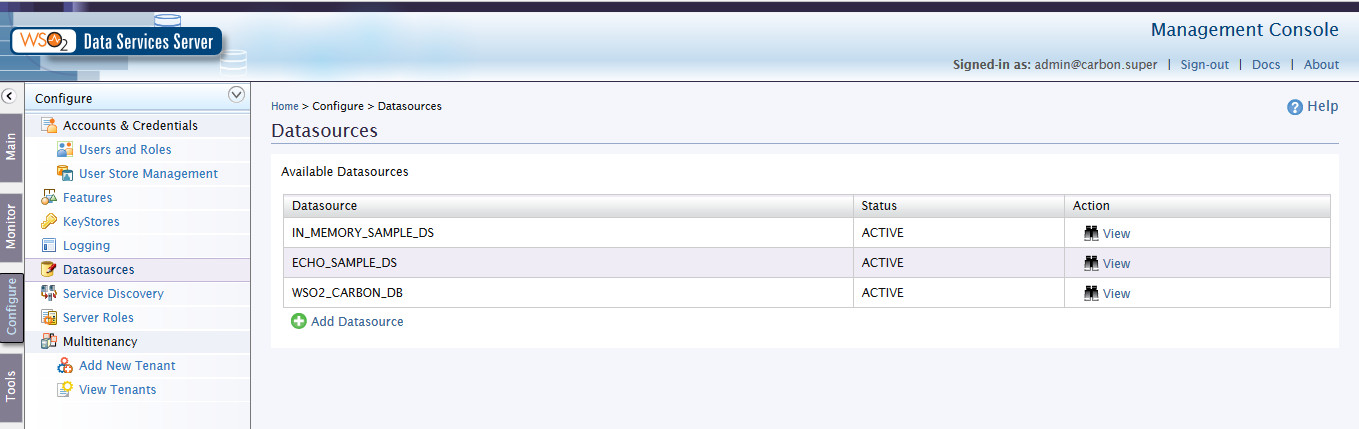


**B**

## **Generar servicios en el DSS**

A- Crear los *datasources* mediante la opción “*Datasources*” del menú “*Configure*”.

B- Adicionar cada *datasource* mediante la opción “*Add Datasource*”.

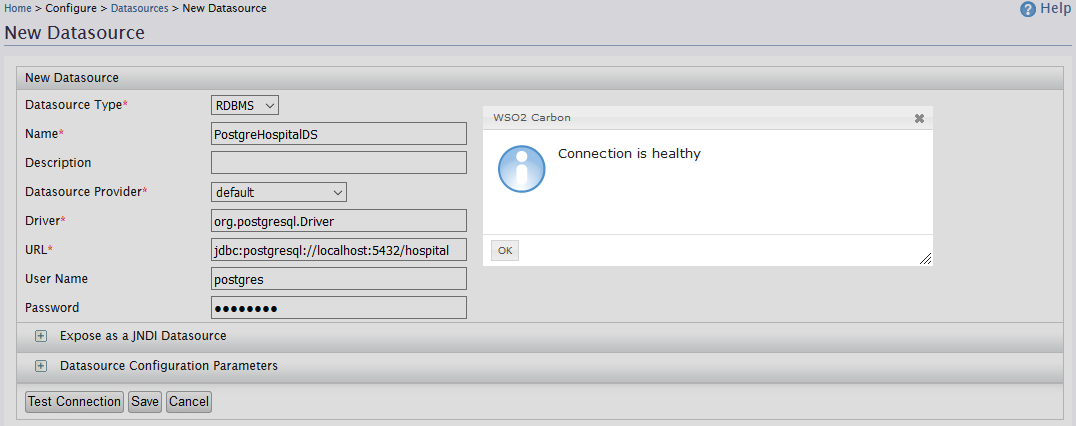


**B**

**AZ**

Adicionar cada *datasource* con la información descrita en el epígrafe 1.1 de la sección anterior. A continuación, se ejemplifica el *datasource* para PostgreSQL.

C- Mediante la opción “*Test Connection*” se validan los parámetros establecidos y finalmente se guardan los cambios mediante la opción “*Save*”.

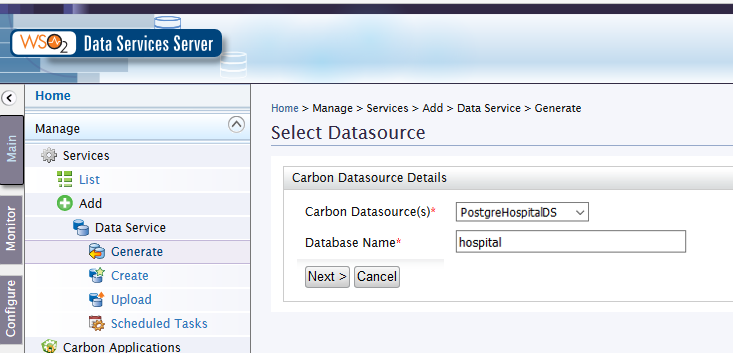


**C**

D- Generar el servicio mediante la opción “*Generate*” del submenú “*Data Service*” del menú “*Main*”.

E- Seleccionar el *datasource* adicionado anteriormente.

F- Especificar el nombre de la base de datos.

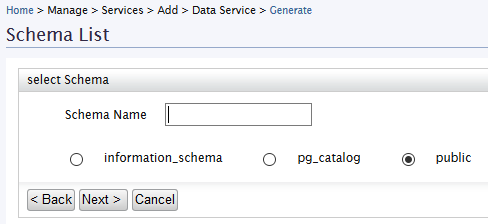


**F**

**E**

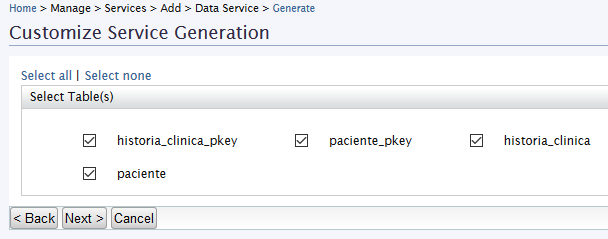
**D**

G- Seleccionar el esquema en el que están las tablas con las cuales se realizarán las consultas.



**G**

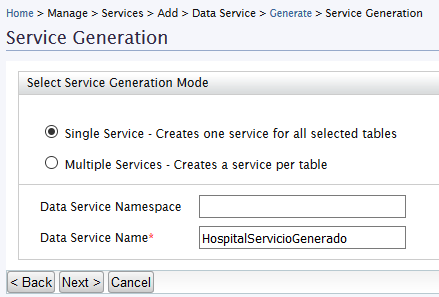
H- Seleccionar las tablas sobre las cuales se realizarán las consultas.



**H**

I- Seleccionar el modo de generación del servicio.

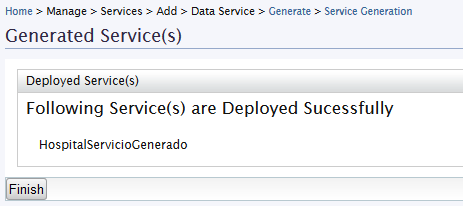
J- Especificar un nombre para el servicio.



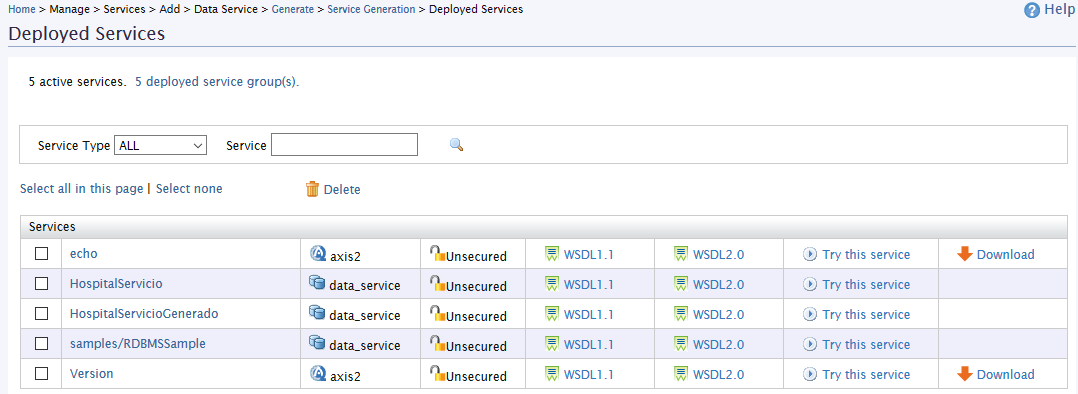
**J**

**I**

Si el servicio se despliega de manera satisfactoria se muestra la siguiente interfaz:

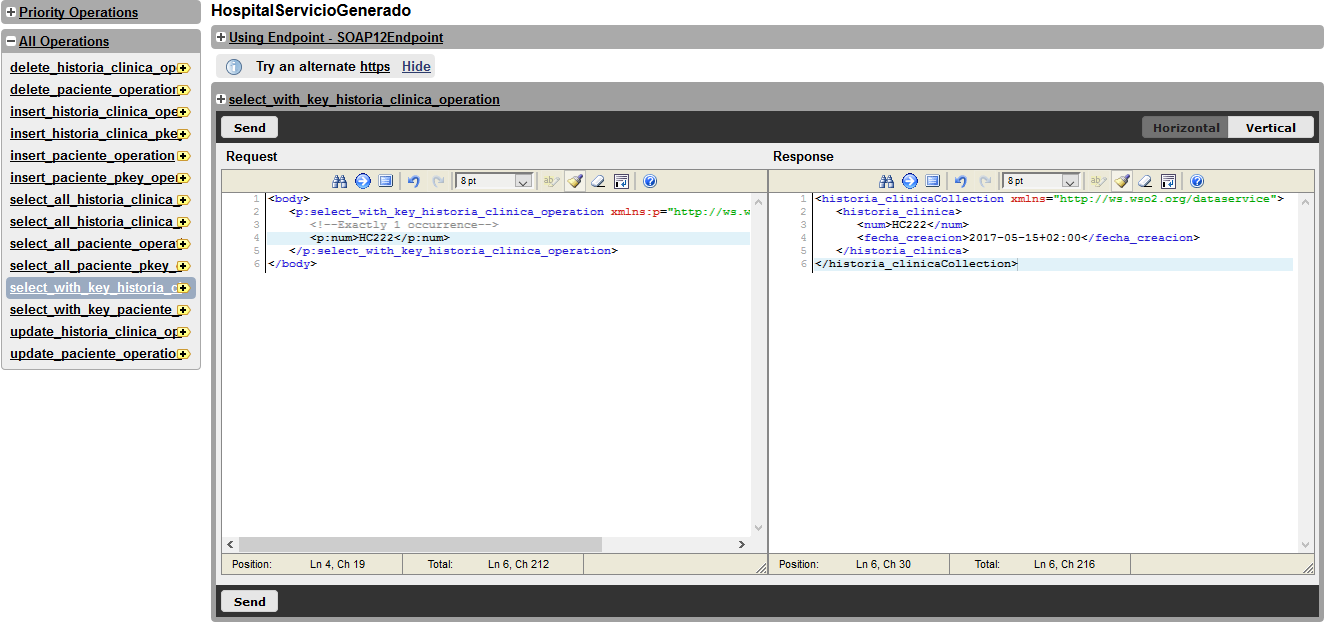


K- Al finalizar este proceso se accede al listado de todos los servicios desplegados donde se muestra el servicio generado.



**K**

Se generan automáticamente todas las operaciones, las cuales pueden probarse de la misma forma, descrita en el epígrafe 1.5 de la sección anterior.

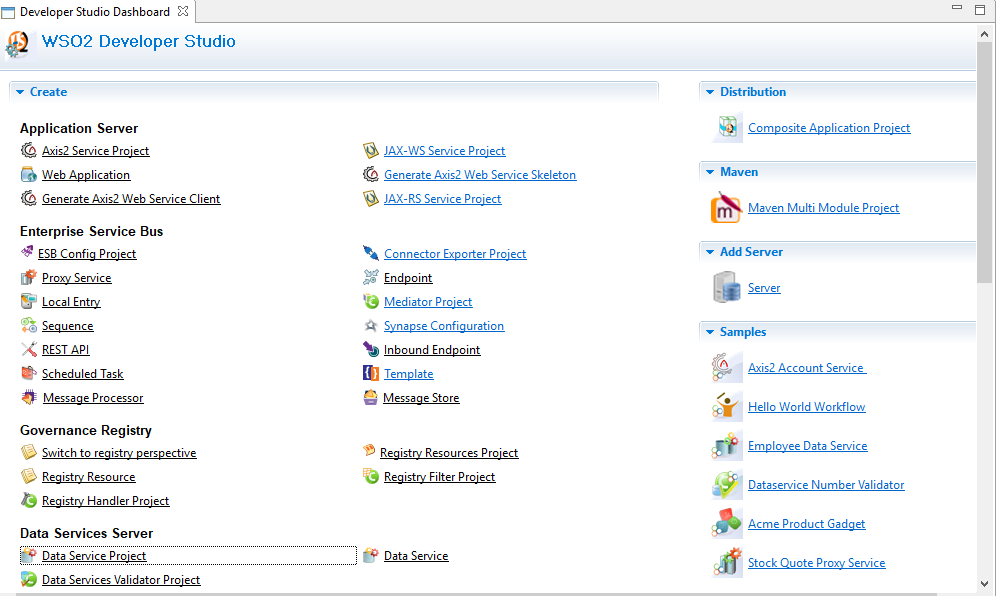


## **Diseñar el servicio en el IDE Eclipse**

Se debe crear un servicio donde se adicionan las consultas, a cada consulta se le especifica el *sql* y se define una operación para ejecutar cada consulta.

A- Crear un proyecto para servicio de datos mediante la opción “*Data Service Project*”. En el proceso de creación se especifica un nombre para el proyecto y los datos para Maven.

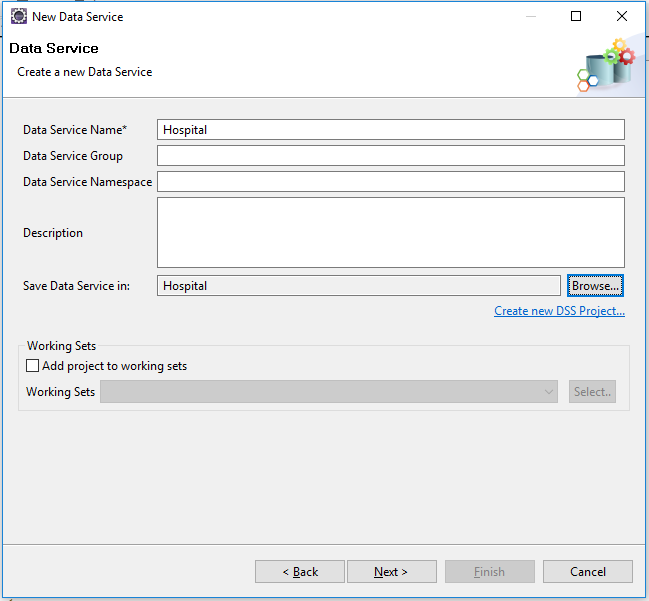
B- Crear un servicio de dato, ya que en la versión que se trabaja estas funciones están independientes. Se selecciona la opción “*Data Service*”, posteriormente se marca la opción “*Create a New Data Service”.*



**B**

**A**

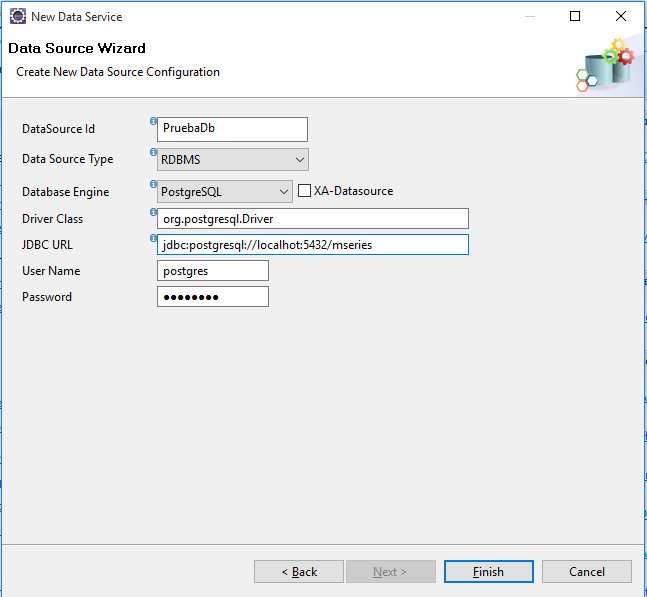
1. Luego se especifica un nombre.
2. En el campo “*Save Data Service in*” mediante la opción “*Browse*” se busca el proyecto de servicio de datos anteriormente creado.



**D**

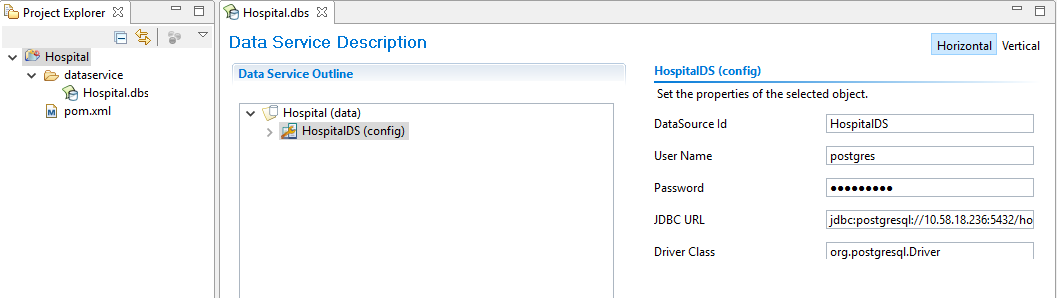
**C**

E- Configurar los parámetros del *Datasource* con la misma información descrita anteriormente para los gestores Oracle, PostgreSQL, MySQL o Microsoft SQL Server.



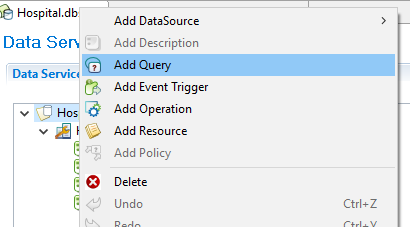
**E**

Cuando se finaliza, el proyecto tiene la siguiente estructura:



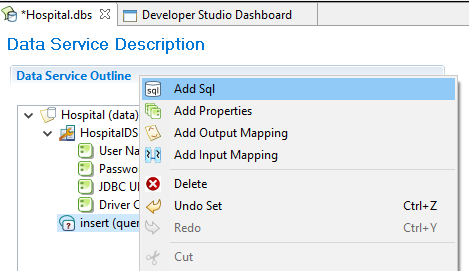
### **3.1 Adicionar las consultas**

A- Se adiciona la consulta mediante la opción “*Add Query*”. Se especifica un nombre, en el ejemplo, se realiza la consulta “*insert*”.

****

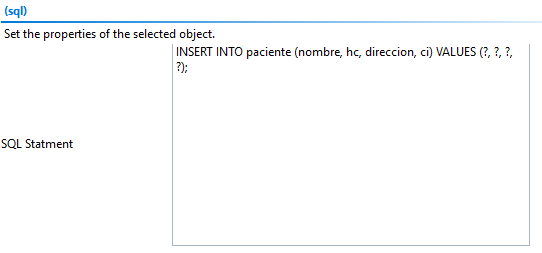
**A**

B- Adicionar el *sql* correspondinte a la consulta que se crea, mediante la opción “*Add Sql*”.



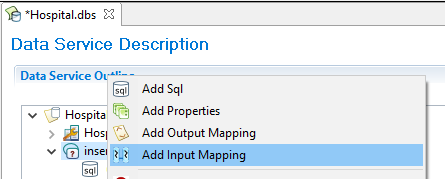
**B**

C- Adicionar la sentencia para este tipo de acción DML.



**C**

D- Adicionar los parámetros de entrada mediante la opción “*Add Input Mapping*”. En este caso la acción *insert* tiene 4 parámetros de entrada.



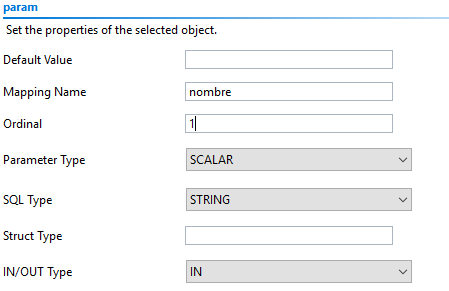
**D**

Para cada parámetro se debe:

E- Especificar el nombre.

F- Especificar la posición que ocupa en el *sql.*

G- Especificar el tipo de dato del campo de entrada.



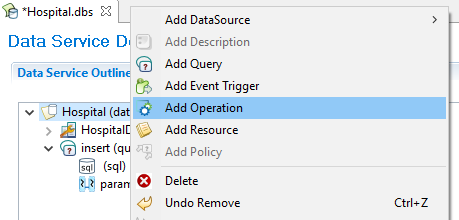
**F**

**G**

**E**

### **3.2 Adicionar las operaciones**

A- Adicionar una operación para cada consulta mediante la opción “*Add Operation*”.



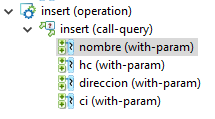
**A**

Para cada operación se debe:

B- Especificar el nombre.

C- Seleccionar la consulta asociada mediante la opción “*Call query*”

**B**



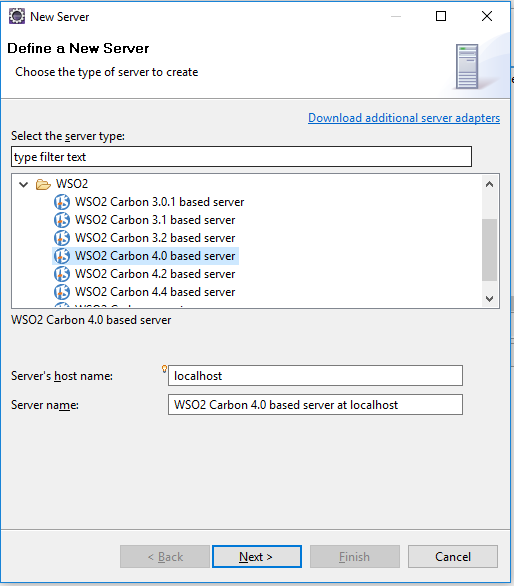
**C**

Los parámetros de entrada se deben generar automáticamente, de lo contrario se definen mediante la opción “*Add Input Mapping*”. A continuación se muestra la vista con las consultas y operaciones definidas en el servicio:



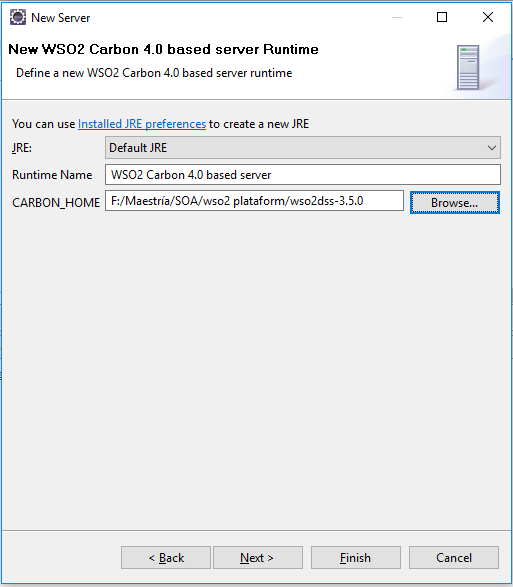
### **3.3 Desplegar el servicio desde el IDE**

A- Se crear un servidor carbón del WSO2.



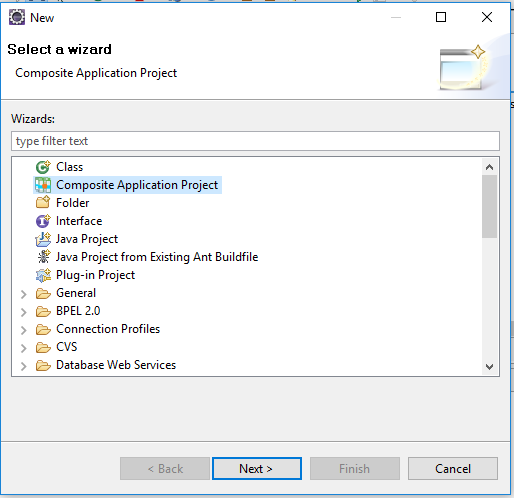
**A**

B- Se especifica la ruta del servidor DSS que se encuentra en el directorio “wso2 plataform\wso2dss-3.5.0”.



**B**

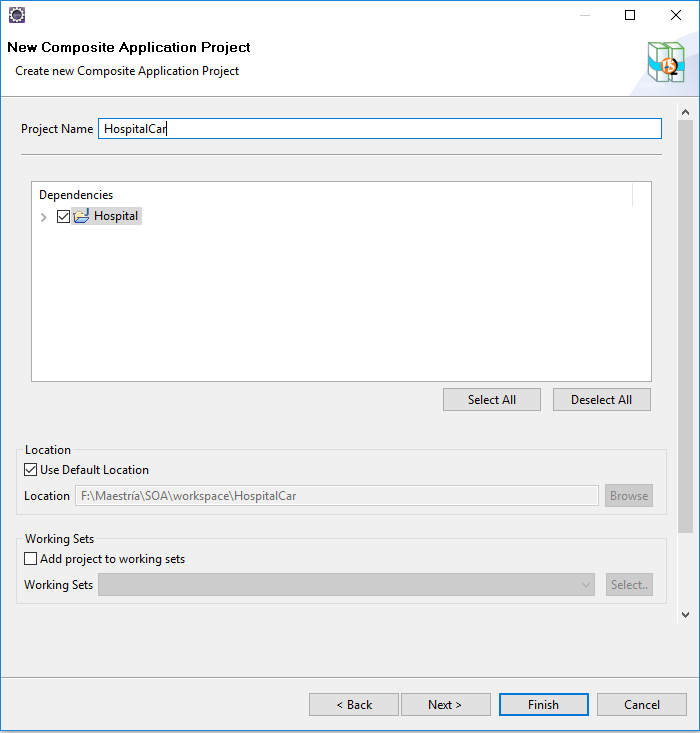
C- Se crea un proyecto “*Composite Application project*” para desplegar el servicio en el DSS.



**C**

D- Se le especifica un nombre.

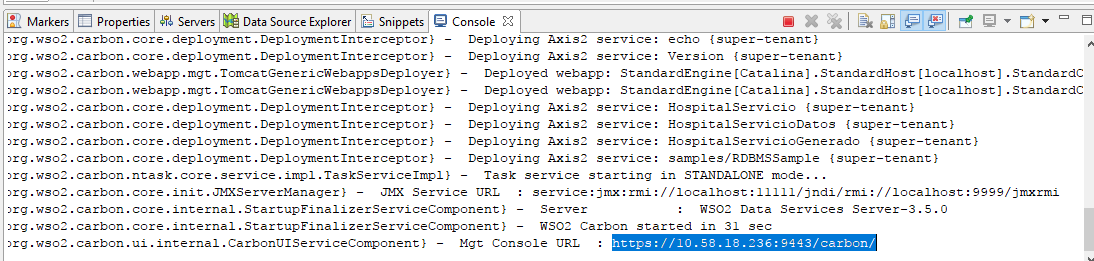
E- Se selecciona el servicio creado.



**E**

**D**

Se corre el proyecto y la *url* mostrada en la consola *(“https://10.58.18.236:9443/carbon/”*) se abre en el navegador para ver el servicio desplegado.

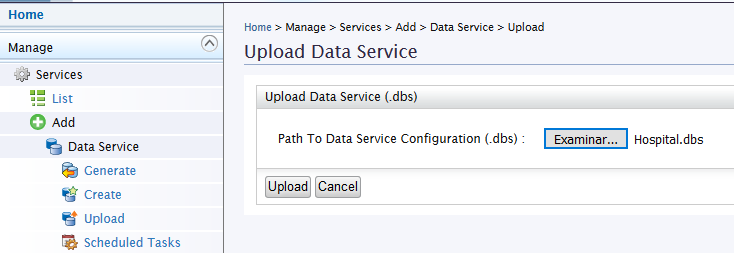


### **3.4 Desplegar el servicio desde la plataforma DSS**

Una vez que el DSS está corriendo, tal y como se explica en el epígrafe 2 del proceso de instalación de las herramientas, para desplegar el servicio previamente diseñado debe:

A- Seleccionar la opción “*Upload*” en el DSS.

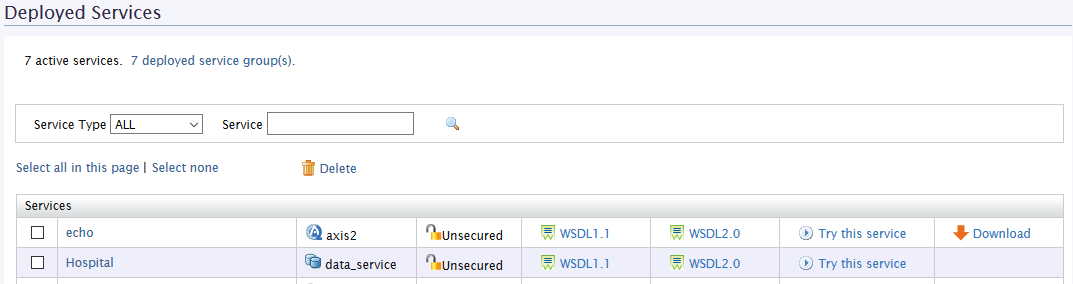
B- Importar el archivo del servicio “*Hospital.dbs*”.



**A**

**B**

C- Al subirlo, se podrá observar en el listado de servicios desplegados.



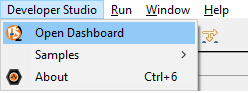
**C**

# **Consumir el servicio por otras aplicaciones**

Se debe generar todo el código fuente del servicio en un paquete de clases determinado. De manera que dicho código pueda ser reutilizado por cualquier sistema que necesite consumir las operaciones definidas por el servicio.

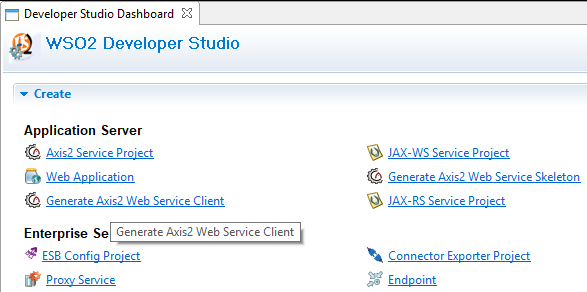
## **Generar un cliente de servicio web Axis2**

A- Primeramente, se debe abrir el *Dashboard* del plugin del *Developer Studio* instalado en el Eclipse y explicado en el epígrafe 3 del proceso de instalación de las herramientas.



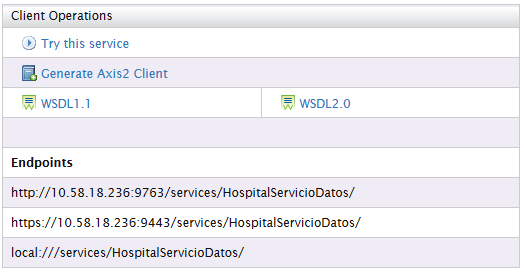
**A**

B- Seleccionar la opción “*Generate Axis2 Web Service Client*”.



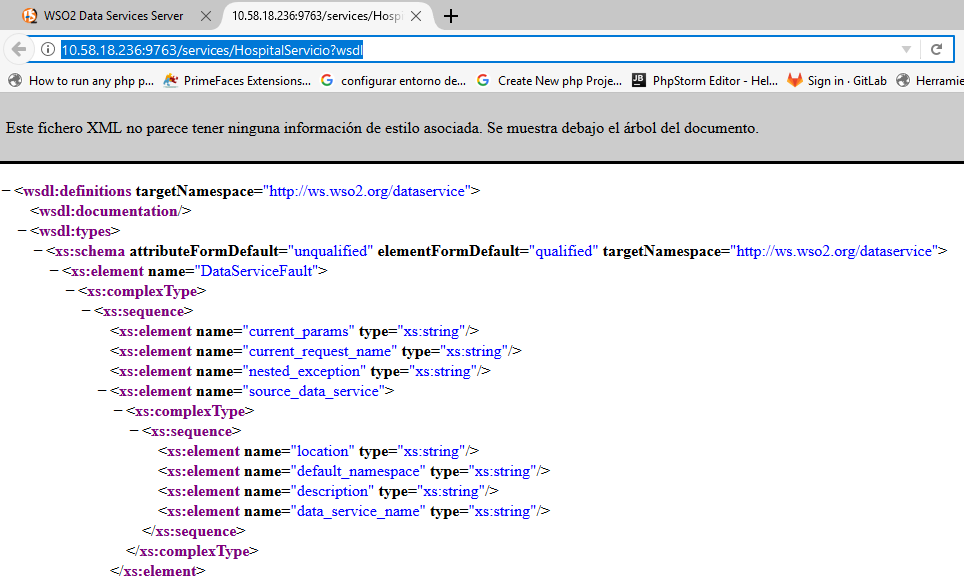
**B**

C- Para especificar el campo *WSDL URI*, se accede a las características del servicio y se selecciona en las operaciones del cliente el *WSDL1.1*. El *WSDL* es el contrato del servicio que fue generado automáticamente por el DSS.



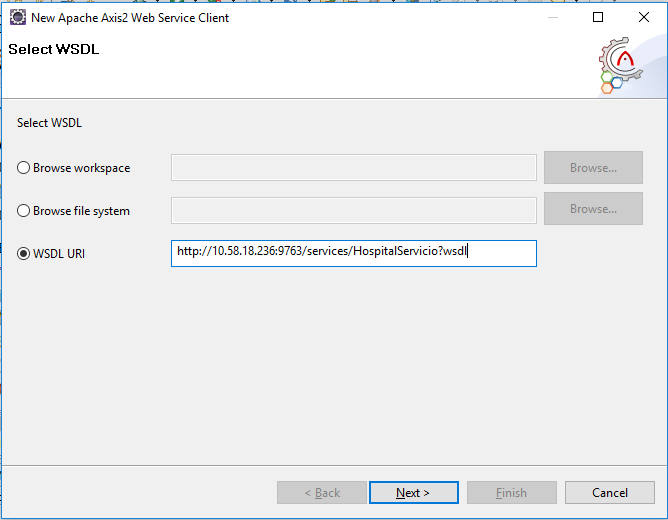
**C**

D- Se copia la dirección donde el DSS tiene expuesto el contrato del servicio.



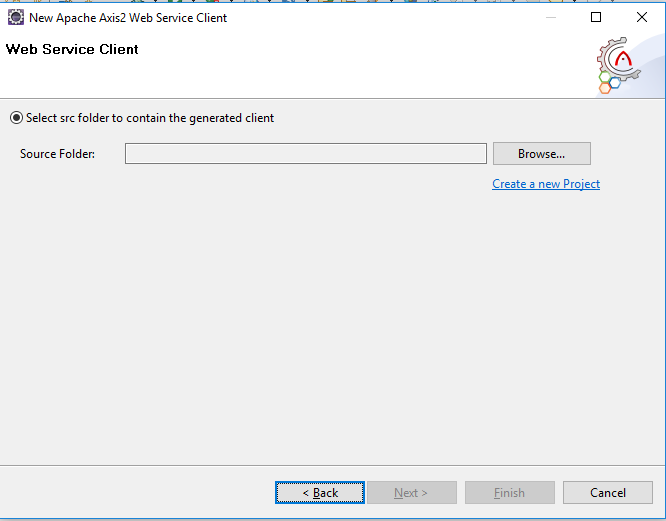
**D**

E- Se copia la dirección en el campo *WSDL URI* y se selecciona la opción “*Next*”.



**E**

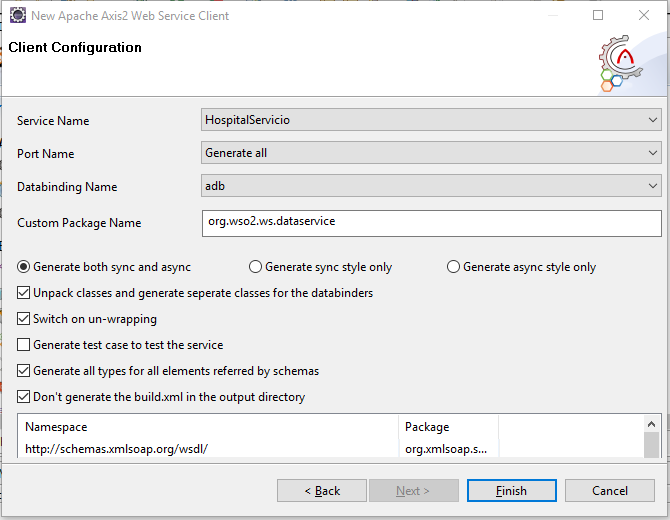
F- Seleccionar la opción “*Create a new Project*”. La creación de este nuevo proyecto es necesaria para especificar una ruta donde va a ser generado todo el código fuente del cliente del servicio.



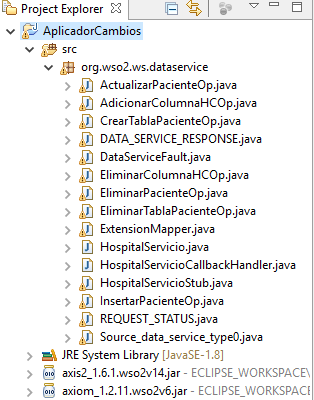
**F**

Se define un nombre al proyecto y al seleccionar la opción “*Next*” se finaliza la creación de proyecto mediante la opción “*Finish*”

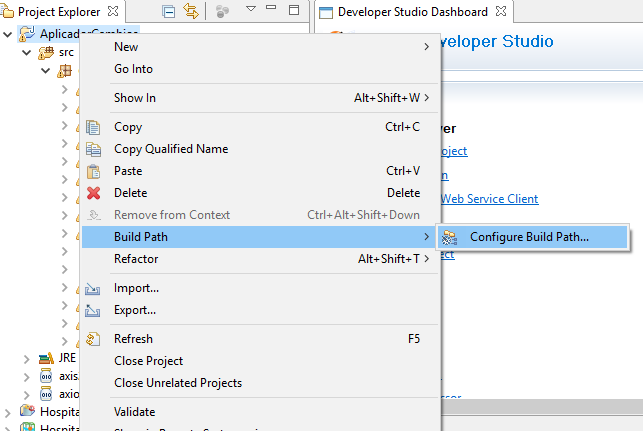
En la ventana de Configuración del cliente deben seleccionarse las opciones tal y como se muestra en la siguiente imagen:



Al finalizar la generación del cliente del servicio se puede observar el código fuente generado en la carpeta *src* del proyecto creado.

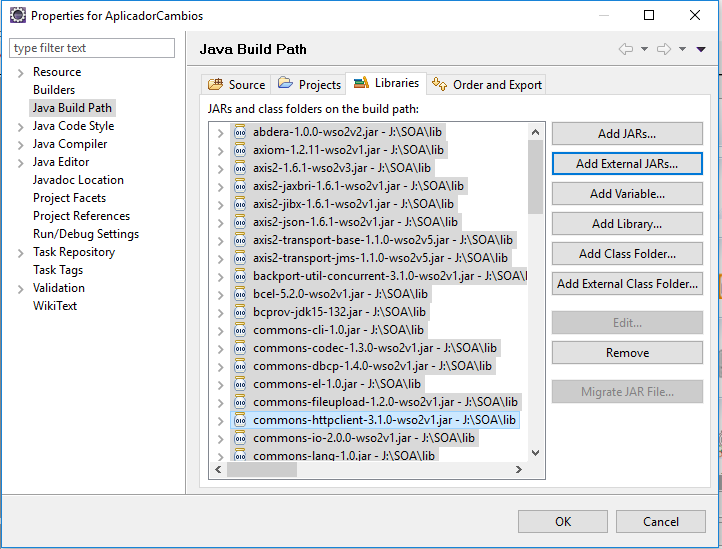


G- Adicionar las librerías necesarias al *Build Path* del proyecto.



**G**

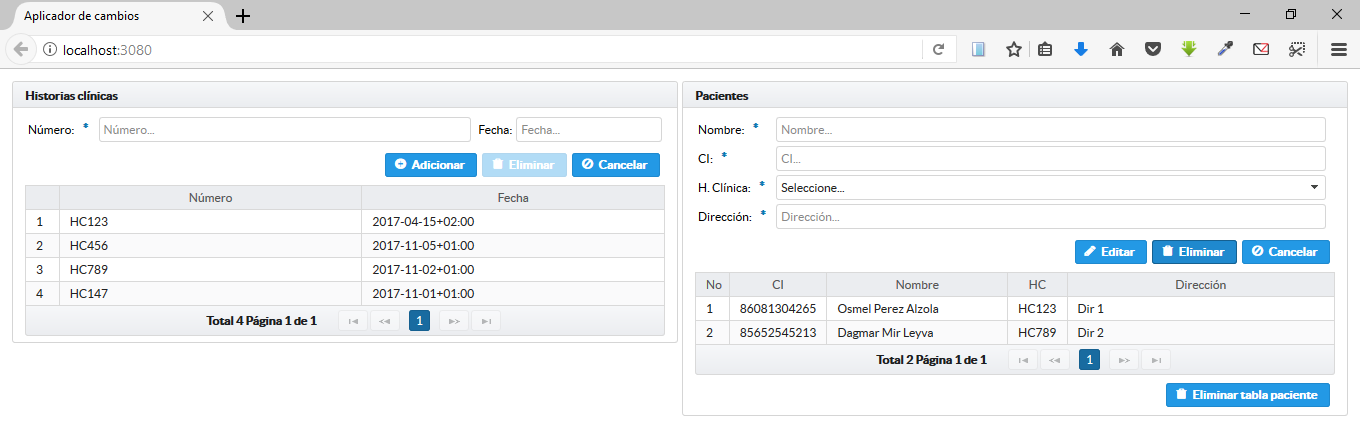
H- Mediante la opción “*Add External JARs*” se adicionan las librerías necesarias para poder consumir el servicio. Estas librerías se encuentran en la carpeta “Herramientas\lib” del presente trabajo.



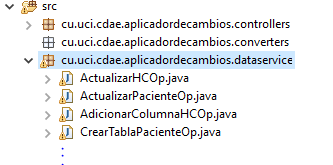
**H**

## **Consumir el servicio**

En el presente trabajo se desarrolló una aplicación web que es la encargada de gestionar los procesos de un hospital. La aplicación consume el servicio creado anteriormente, de manera que todas las operaciones DML y DDL se realizan a través del servicio. A continuación, se explica cómo se consume el servicio por esta aplicación.

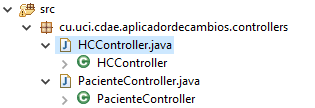


A- En la aplicación se debe utilizar el código fuente del cliente del servicio generado, añadiendo previamente las librerías necesarias tal y como se explicó anteriormente.



**A**

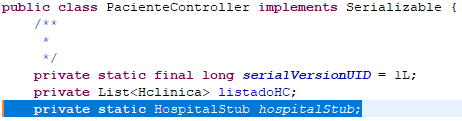
B- Crear la o las clases encargadas de acceder a todas las operaciones definidas en el servicio.



**B**

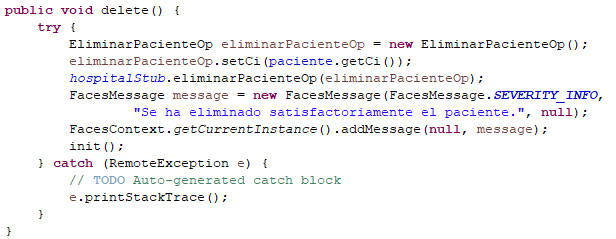
Todo servicio tiene una clase *Stub* generada automáticamente. Esta clase es la controladora del servicio en general, es decir, para ejecutar las operaciones definidas en el servicio debe ser a través de ella. En el caso del servicio que se presenta como ejemplo, una de las operaciones consiste en eliminar un paciente.

C- Como se muestra en la siguiente imagen se crea una instancia de la clase *Stub.*



**C**

D- Para luego ser utilizada en la operación de eliminación de un paciente.

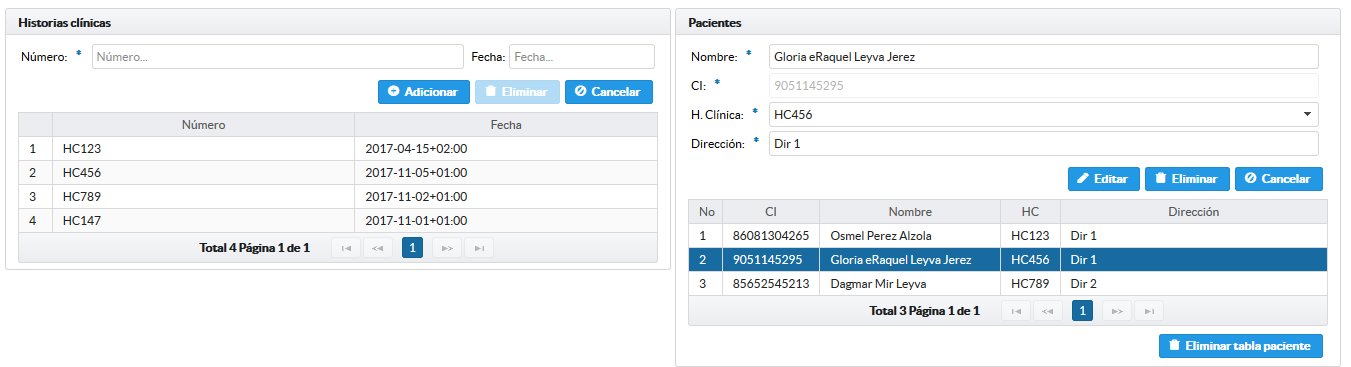


**D**

De esta manera se consumirían el resto de las operaciones definidas en el servicio. A continuación, se muestra el flujo de eliminación de un paciente desde la aplicación consumiendo el servicio.

E- Se selecciona el paciente del listado.

F- Se elige la opción “*Eliminar*”.

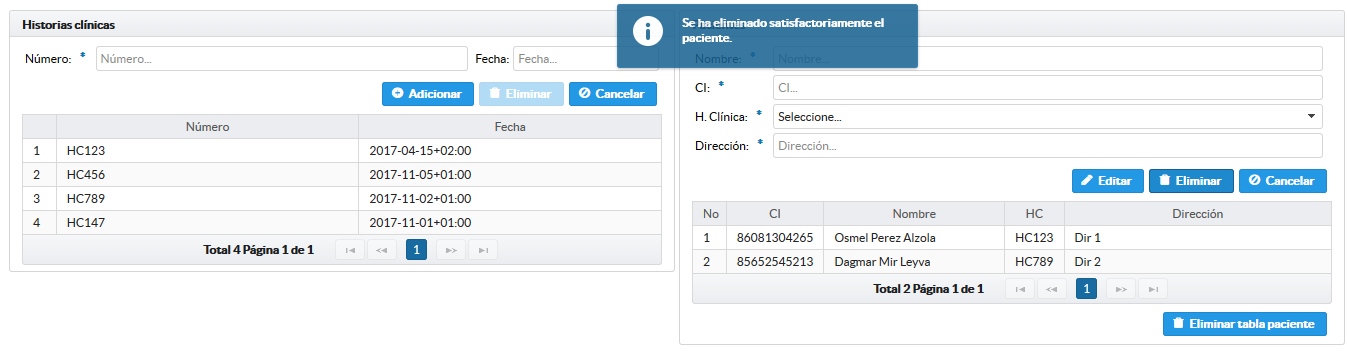


**F**

**E**

G- Al consumir la operación del servicio correspondiente a eliminar un paciente, se observa como es eliminado satisfactoriamente.

**G**



# **Conclusiones**

Al finalizar el presente trabajo se arribaron a las siguientes conclusiones:

* Documentar el proceso de creación del servicio web de datos mediante el manual que se describe en el presente trabajo, facilita el desarrollo de servicios web a los programadores.
* La implementación y apropiación de este servicio web para la replicación de datos, específicamente para la aplicación de cambios permite el desarrollo de sistemas cada vez más robustos e interoperables, facilitando las tareas de integración de datos cuando existen diferentes motores de base de datos involucrados, sin incurrir en los altos costos que hoy las compañías de software sostienen.

1. Lenguaje de Manipulación de Datos, del inglés *Data Manipulation Language*. Se especializa en la utilización de la BD y permite realizar consultas mediante los comandos *insert, update, select y delete.* [↑](#footnote-ref-1)
2. Lenguaje de Definición de Datos, *del inglés Data Definition Language.* Se especializa en la escritura de esquemas, es decir, en la descripción de la BD y permite realizar las operaciones básicas de *create, alter y drop.* [↑](#footnote-ref-2)
3. Entorno de Desarrollo Integrado, del inglés *Integrated Development Environment*. Es un programa compuesto por un conjunto de herramientas de programación que permite escribir código fuente, compilarlo y ejecutarlo sin necesidad de cambiar de aplicación. [↑](#footnote-ref-3)