TRABAJO DE TITULACIÓN:

“AGENTES INTELIGENTES PARA LA GESTIÓN DE RECURSOS DIGITALES DE APRENDIZAJE”

**MANUAL DE INSTALACIÓN**

**smaweb**

**Versión 1.0**

|  |  |
| --- | --- |
| **DATOS GENERALES** | |
| **Código:** | 002 |
| **Versión:** | 1.0 |
| **Fecha de la versión:** | 25 de septiembre de 2019 |
| **Páginas:** | 16 |
| **Creado por:** | Alex Condoy |
| **Nivel de confidencialidad:** | Medio |

|  |  |  |  |
| --- | --- | --- | --- |
| **CONTROL DE VERSIONES** | | | |
| **Código** | **Versión** | **Fecha** | **Responsable** |
| 002 | 1.0 | 25 de septiembre de 2019 | Alex Rubén Condoy Carrión |
|  |  |  |  |
|  |  |  |  |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **CONTROL DE MODIFICACIONES** | | | | |
| **Código** | **Versión** | **Fecha** | **Responsable** | **Descripción** |
| 002 | 1.0 | 25 de septiembre de 2019 | Alex Rubén Condoy Carrion | Creación |
|  |  |  |  |  |
|  |  |  |  |  |

**TABLA DE CONTENIDO**

[1. INTRODUCCIÓN 5](#_Toc20391854)

[2. DIRIGIDO A USUARIOS 5](#_Toc20391855)

[3. OBJETIVO 5](#_Toc20391856)

[4. REQUERIMIENTOS PREVIOS. 5](#_Toc20391857)

[5. CONEXIÓN CON LA BASE DE DATOS 6](#_Toc20391858)

[6. CONEXIÓN CON OPENLINK VIRTUOSO 9](#_Toc20391859)

[7. INSTALACIÓN DEL SMA EN EL SERVIDOR WEB 14](#_Toc20391860)

[8. INGRESAR 16](#_Toc20391861)

**ÍNDICE DE FIGURAS**

[Figura 1. Arquitectura de Instalación. 5](#_Toc20391871)

[Figura 2. Creación de la BD. 6](#_Toc20391872)

[Figura 3. Script de BD. 7](#_Toc20391873)

[Figura 4. Tablas admin y usuarios. 8](#_Toc20391874)

[Figura 5. Creación del usuario para el SMA. 8](#_Toc20391875)

[Figura 6. Permisos de usuario de BD. 8](#_Toc20391876)

[Figura 7. Driver MySql en NetBeans. 9](#_Toc20391877)

[Figura 8. Conexión del proyecto con la BD. 9](#_Toc20391878)

[Figura 9. Instalación de OpenLink Virtuoso (Windows). 10](#_Toc20391879)

[Figura 10. OpenLink Virtuoso Server [vos] 10](#_Toc20391880)

[Figura 11. Panel de Administración de Virtuoso. 11](#_Toc20391881)

[Figura 12. Virtuoso Conductor. 11](#_Toc20391882)

[Figura 13. Linked Data. 11](#_Toc20391883)

[Figura 14. Creación de la Tabla (Graph) LearningObjects. 12](#_Toc20391884)

[Figura 15. Creación de la Tabla (Graph) Perfiles. 12](#_Toc20391885)

[Figura 16. Graphs de LeanringObjects y Perfiles creadas. 12](#_Toc20391886)

[Figura 17. Virtuoso JDBC 4 Driver JAR file. 13](#_Toc20391887)

[Figura 18. Dependencias de Virtuoso. 13](#_Toc20391888)

[Figura 19. Código de conexión con el servidor Virtuoso. 13](#_Toc20391889)

[Figura 20. Reglas de firewall para acceso de Virtuoso. 14](#_Toc20391890)

[Figura 21. Seleccionar sma\_web como el proyecto principal. 14](#_Toc20391891)

[Figura 22. Mensaje de confirmación al generar el archivo WAR. 15](#_Toc20391892)

[Figura 23. Archivo sma\_web.war. 15](#_Toc20391893)

[Figura 24. Descompresión del archivo WAR en webapps/ROOT. 15](#_Toc20391894)

[Figura 25. Despliegue del sistema en el servidor. 16](#_Toc20391895)

# INTRODUCCIÓN

El presente manual explica paso a paso y detalladamente el proceso para la instalación del sistema **SMAWEB,** que permite buscar objetos de aprendizaje (OAs) en base a un perfil inteligente.

# DIRIGIDO A USUARIOS

El presente manual está destinado al personal técnico y/o investigadores que pretendan realizar la instalación del Sistema Multiagente (SMA) desarrollado en el trabajo de titulación: “Agentes Inteligentes para la Gestión de Recursos Digitales de Aprendizaje”.

# OBJETIVO

Detallar el proceso necesario para la instalación del SMA, el mismo que permitirá realizar búsquedas de objetos de aprendizaje en base al perfil inteligente de un estudiante.

# REQUERIMIENTOS PREVIOS.

Para realizar la instalación del sistema multiagente son necesarios ciertos requisitos previos, los mismo que se deducen de la arquitectura de despliegue mostrada en Figura 1.

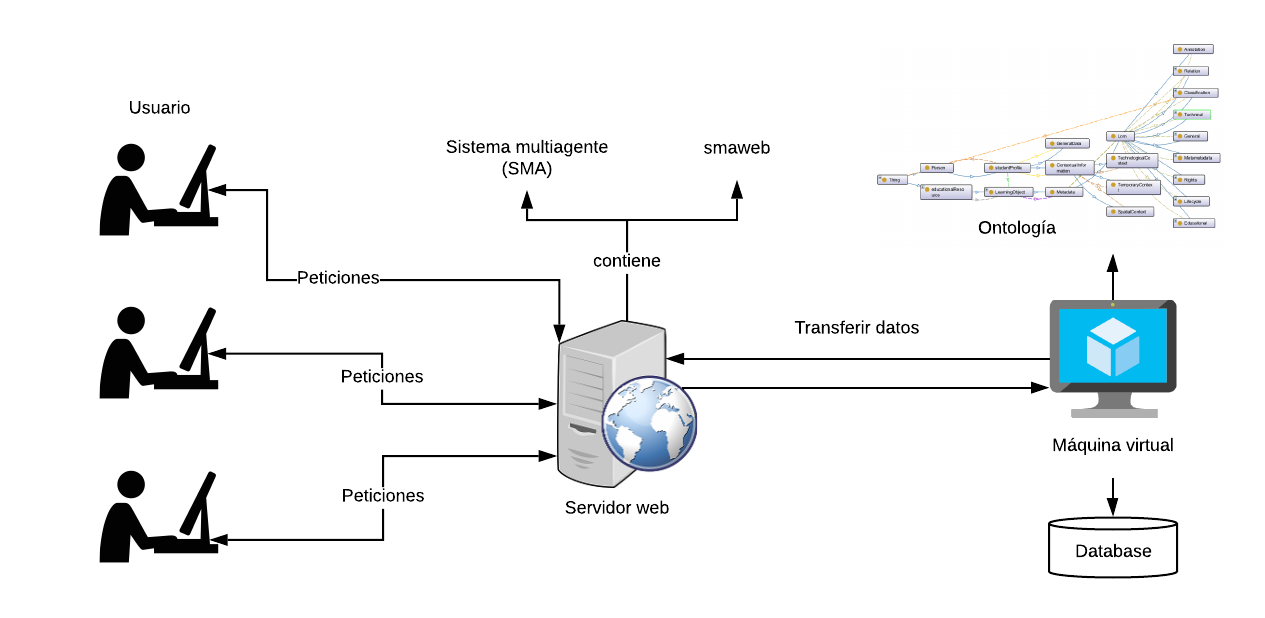


Figura 1. Arquitectura de Instalación.

**Requerimientos:**

* Servidor Web que contenga **Apache Tomcat 7.0** o superior.
* Una Máquina Virtual (VM) con mínimo: 1GB RAM – 1VCPU – SO: Windows server 2012 o Linux.
* Instancia de **XAMMP v.3.2.4** o superior en la VM.

# CONEXIÓN CON LA BASE DE DATOS

El SMA registra usuarios y su información, por lo cual debe ser conectado con una Base de Datos (BD); esta BD puede estar instalada en cualquier ubicación, pero para la presente implementación se encuentra en una máquina virtual dentro de una instancia de XAMMP.

1. Se debe crear una base de datos “**sma\_web**”.

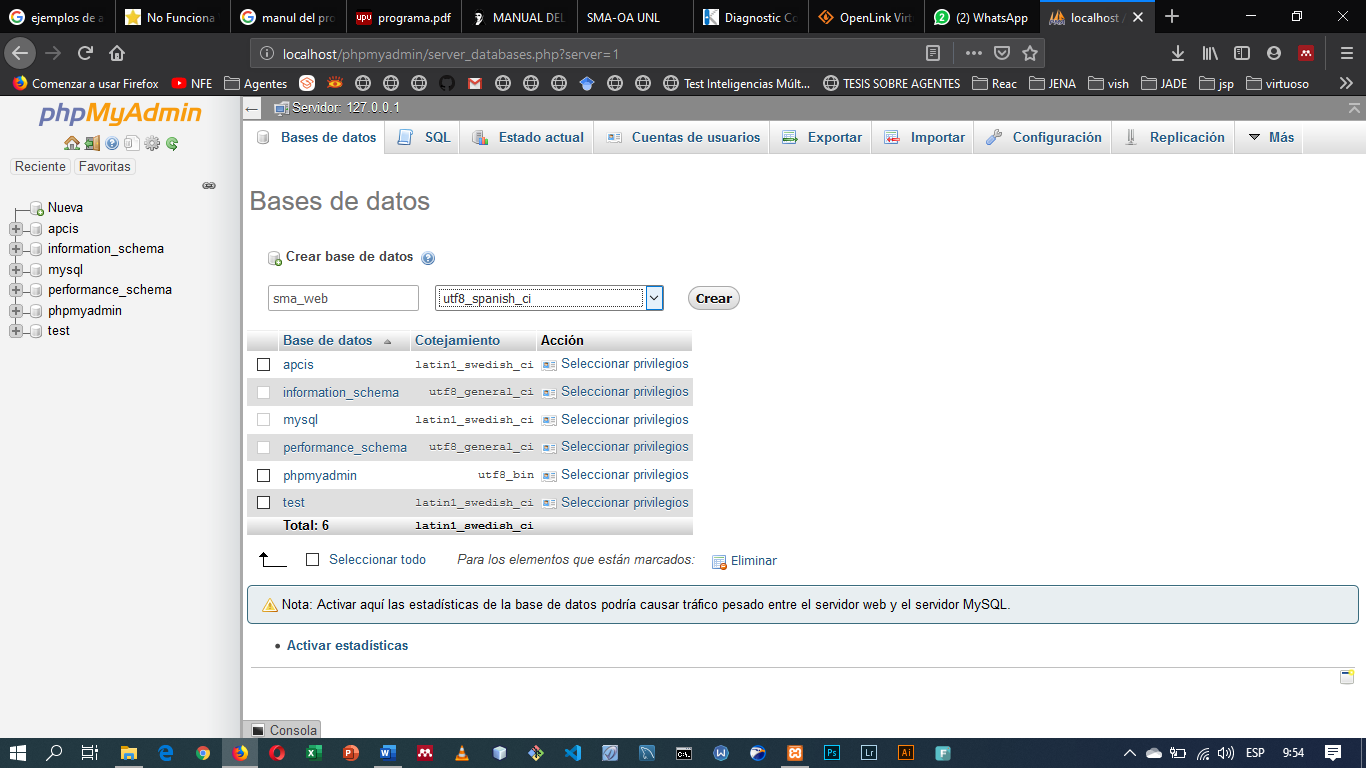


Figura 2. Creación de la BD.

1. Una vez creada, se deberá importar el script para crear las tablas de usuarios y administradores.

-- phpMyAdmin SQL Dump

-- version 4.8.3

-- https://www.phpmyadmin.net/

-- Servidor: 127.0.0.1

-- Tiempo de generación: 27-06-2019 a las 21:12:01

-- Versión del servidor: 10.1.37-MariaDB

-- Versión de PHP: 7.2.12

SET SQL\_MODE = "NO\_AUTO\_VALUE\_ON\_ZERO";

SET AUTOCOMMIT = 0;

START TRANSACTION;

SET time\_zone = "+00:00";

/\*!40101 SET @OLD\_CHARACTER\_SET\_CLIENT=@@CHARACTER\_SET\_CLIENT \*/;

/\*!40101 SET @OLD\_CHARACTER\_SET\_RESULTS=@@CHARACTER\_SET\_RESULTS \*/;

/\*!40101 SET @OLD\_COLLATION\_CONNECTION=@@COLLATION\_CONNECTION \*/;

/\*!40101 SET NAMES utf8mb4 \*/;

-- Base de datos: `sma\_web`

-- --------------------------------------------------------

-- Estructura de tabla para la tabla `admin`

CREATE TABLE `admin` (

`user` varchar(30) COLLATE utf8\_spanish\_ci NOT NULL,

`password` varchar(20) COLLATE utf8\_spanish\_ci NOT NULL

) ENGINE=InnoDB DEFAULT CHARSET=utf8 COLLATE=utf8\_spanish\_ci;

-- Estructura de tabla para la tabla `usuarios`

CREATE TABLE `usuarios` (

`id` int(11) NOT NULL,

`email` varchar(100) NOT NULL,

`password` varchar(50) NOT NULL,

`name` varchar(100) NOT NULL,

`registro\_google` tinyint(1) NOT NULL DEFAULT '0',

`inteligente` tinyint(1) NOT NULL,

`LearningStyles` text CHARACTER SET utf8 COLLATE utf8\_spanish\_ci

) ENGINE=MyISAM DEFAULT CHARSET=utf8;

-- Indices de la tabla `usuarios`

ALTER TABLE `usuarios`

ADD PRIMARY KEY (`id`);

-- AUTO\_INCREMENT de la tabla `usuarios`

ALTER TABLE `usuarios`

MODIFY `id` int(11) NOT NULL AUTO\_INCREMENT, AUTO\_INCREMENT=1;

COMMIT;

/\*!40101 SET CHARACTER\_SET\_CLIENT=@OLD\_CHARACTER\_SET\_CLIENT \*/;

/\*!40101 SET CHARACTER\_SET\_RESULTS=@OLD\_CHARACTER\_SET\_RESULTS \*/;

/\*!40101 SET COLLATION\_CONNECTION=@OLD\_COLLATION\_CONNECTION \*/;

Figura 3. Script de BD.

1. Se debe verificar que las tablas se hayan creado correctamente.

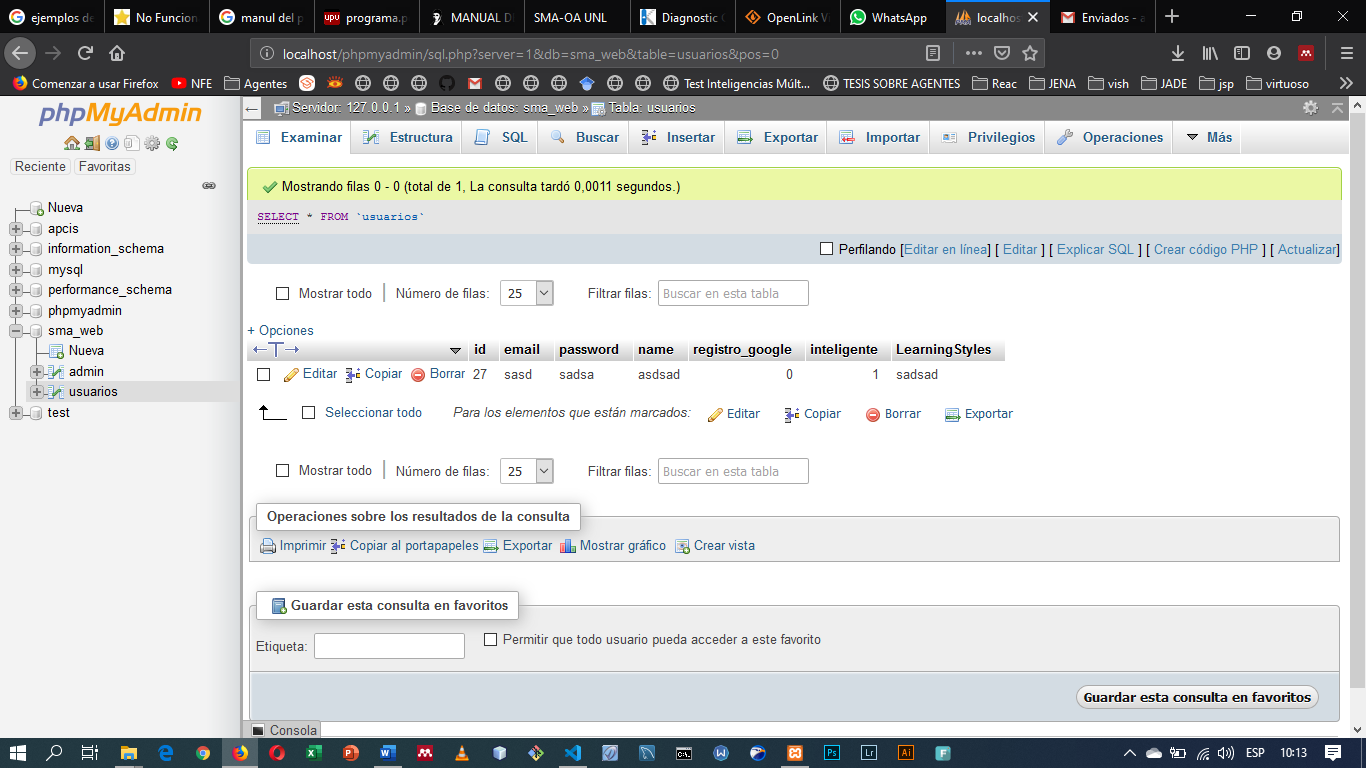


Figura 4. Tablas admin y usuarios.

1. Se crea un usuario y un administrador para que gestione la base de datos.

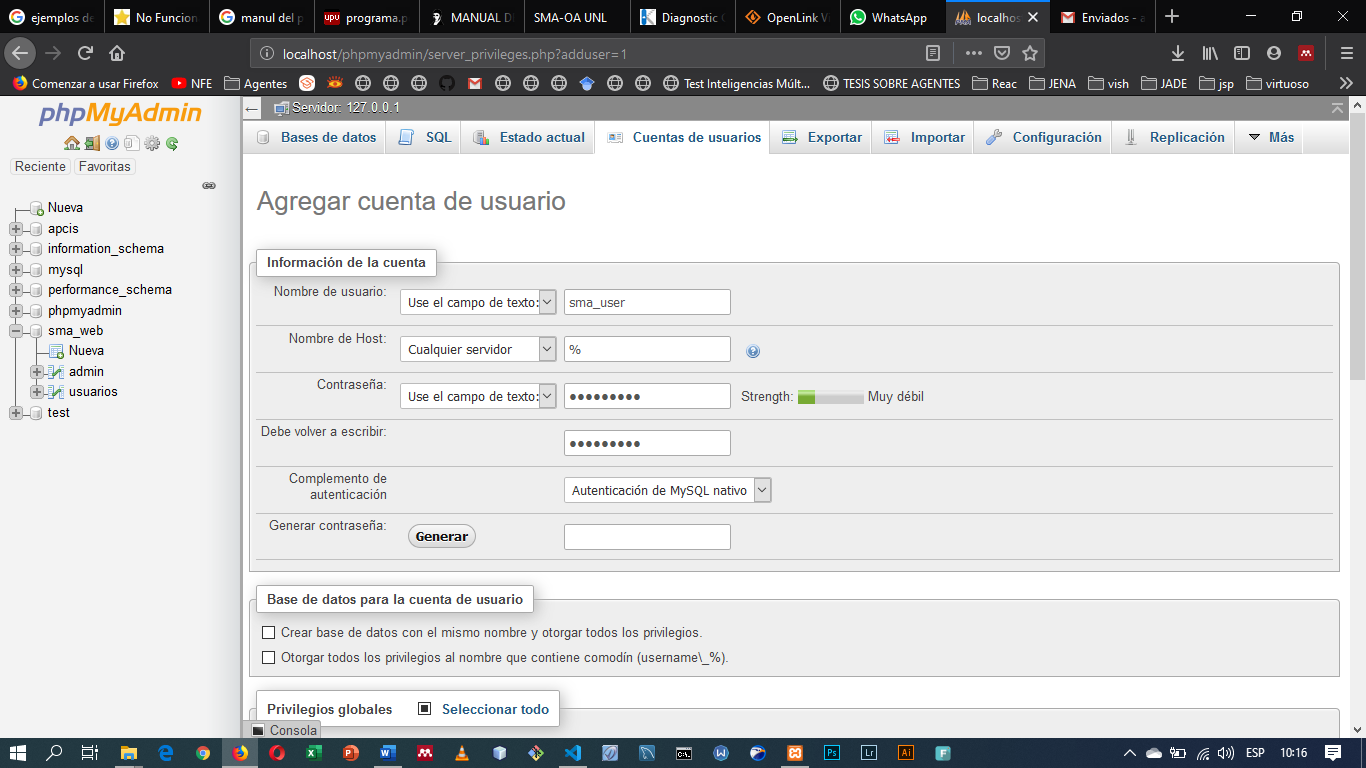


Figura 5. Creación del usuario para el SMA.

1. Se debe dar los permisos que se muestran en la Figura 6.

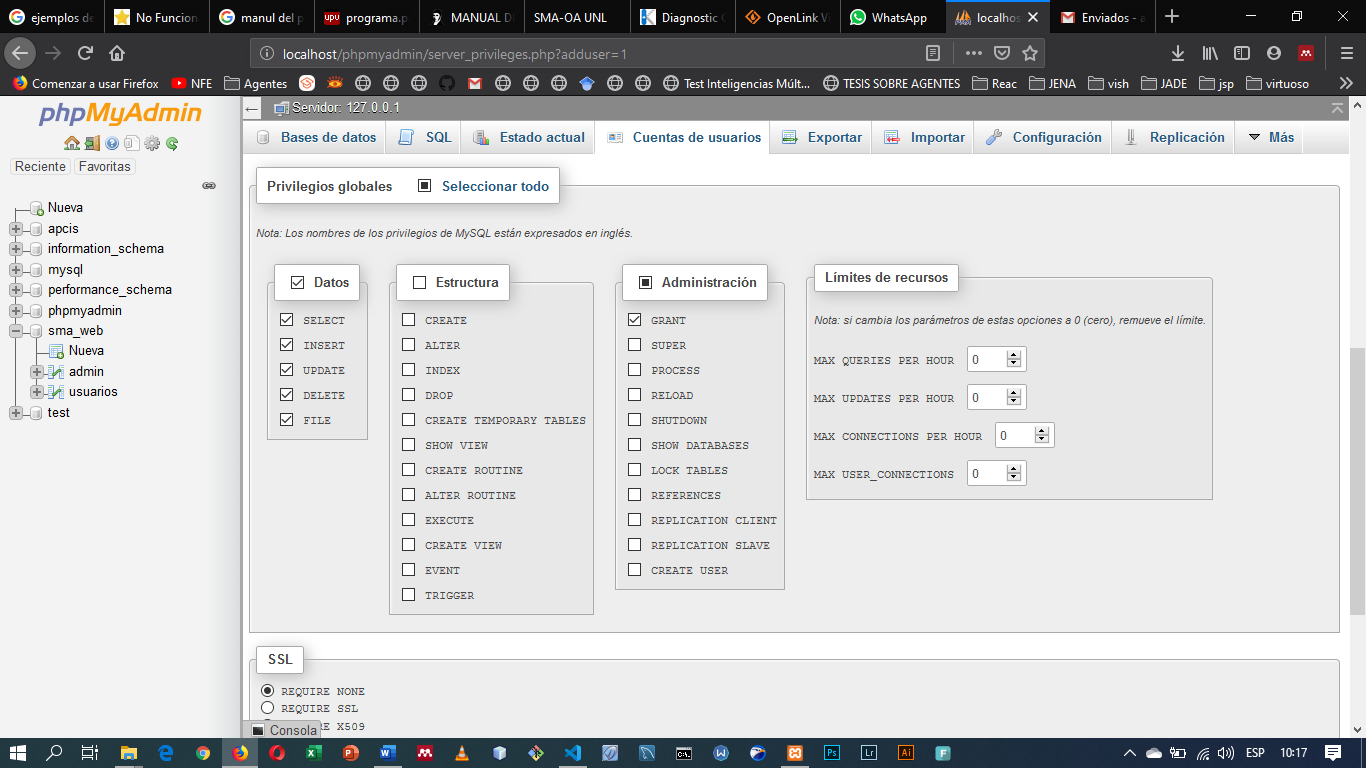


Figura 6. Permisos de usuario de BD.

1. El proyecto debe contar con la librería necesaria para realizar conexiones MySql, para la implementación del SMA se ha utilizado **mysql-connector-java-5.1.23-bin.jar**

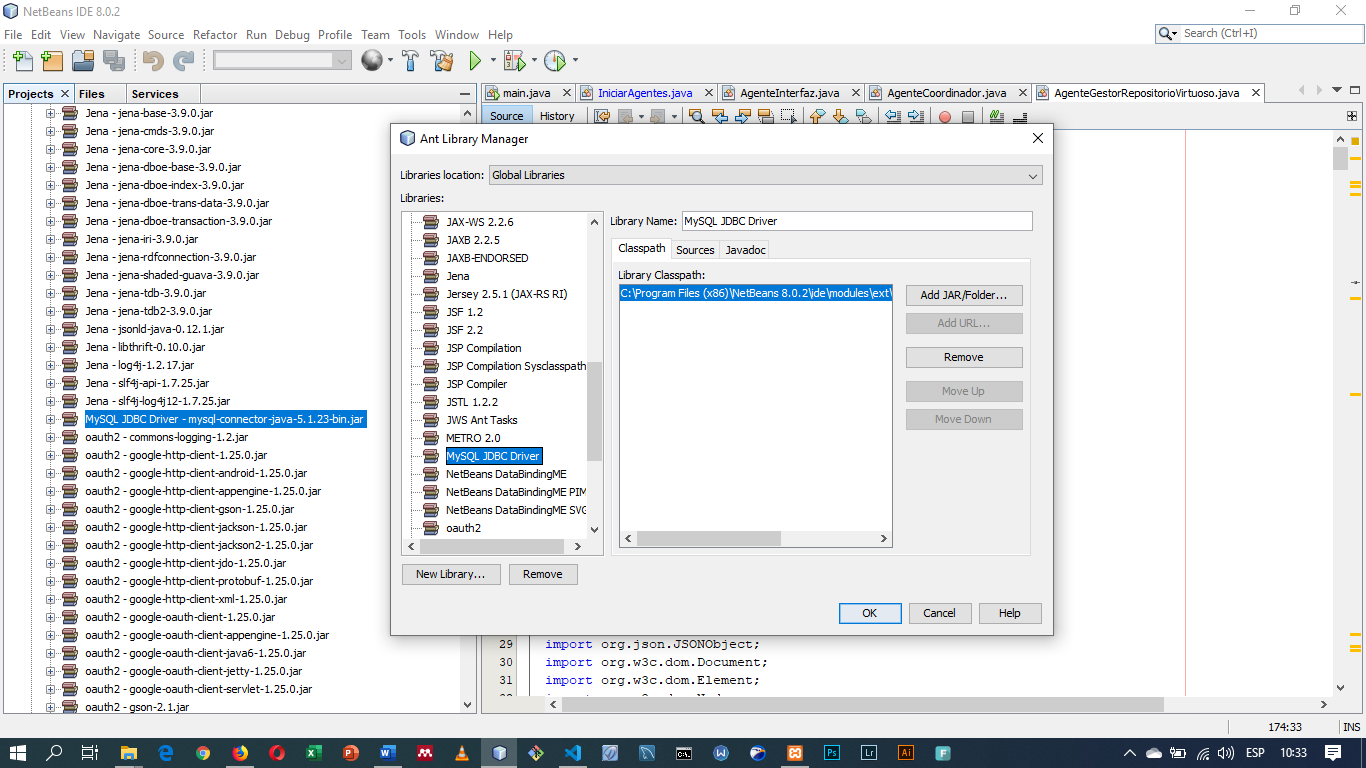


Figura 7. Driver MySql en NetBeans.

1. En el paquete **model,** agregamos la conexión dentro de la clase **Dao**, como lo muestra la Figura 8.

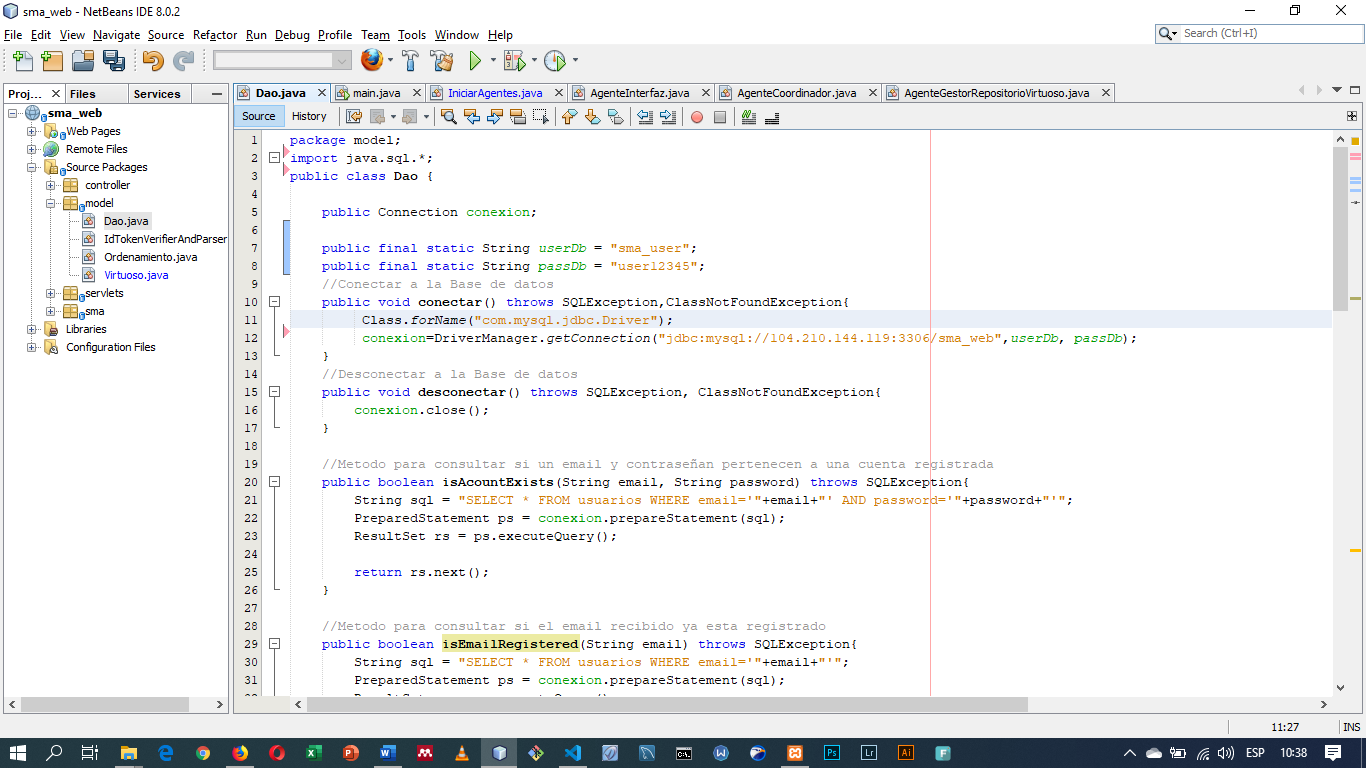


Figura 8. Conexión del proyecto con la BD.

1. Resulta importante mencionar que el puerto 3306 del servidor donde se encuentre la BD debe estar habilitado para los accesos por parte del exterior.

# CONEXIÓN CON OPENLINK VIRTUOSO

Para almacenar y recuperar objetos de aprendizaje es necesario instalar un gestor de ontologías, el cual contendrá la ontología con los individuos (instancias) de los objetos de aprendizaje y almacenará el perfil inteligente de los estudiantes.

Se puede utilizar cualquier motor para la gestión de ontologías, pero en este caso, se ha usado OpenLink Virtuoso en su versión libre. Por lo que en las siguientes líneas se describe detalladamente como realizar la vinculación de virtuoso con el sistema;

1. Lo primero a realizar es la instalación de Virtuoso, descargado de la siguiente dirección: <https://sourceforge.net/projects/virtuoso/>, esta instalación es sencilla en Windows y no necesita mayor explicación, para una instalación en Linux, se puede seguir el procedimiento explicado en:

<http://vos.openlinksw.com/owiki/wiki/VOS/VOSUbuntuNotes>.

1. Ejecutamos **Virtuoso Services Control**

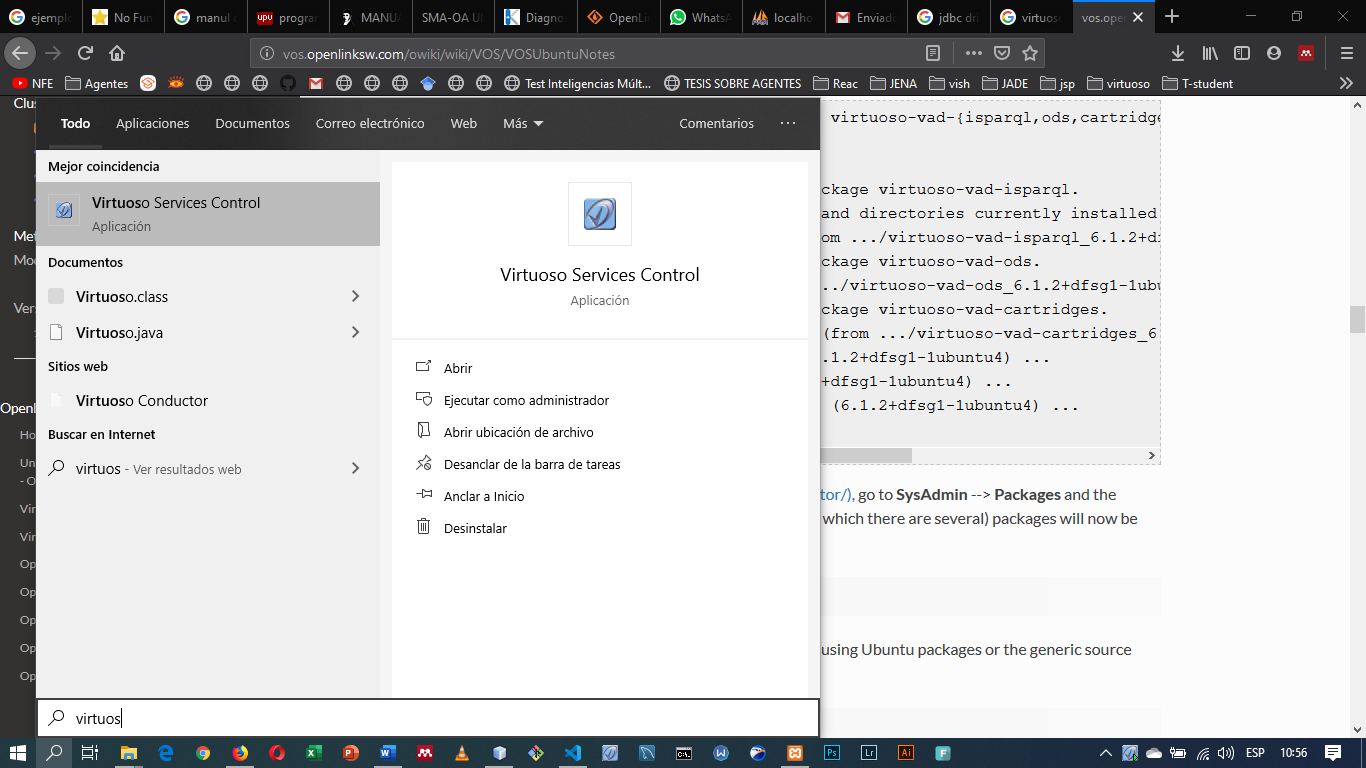


Figura 9. Instalación de OpenLink Virtuoso (Windows).

1. Seleccionamos el servicio creado por defecto (**OpenLink Virtuoso Server [vos]**) o en el que se hay creado con anterioridad de ser el caso. Una vez seleccionado el **[vos],** damos clic en **OpenLink Virtuoso Server[vos]- Start**

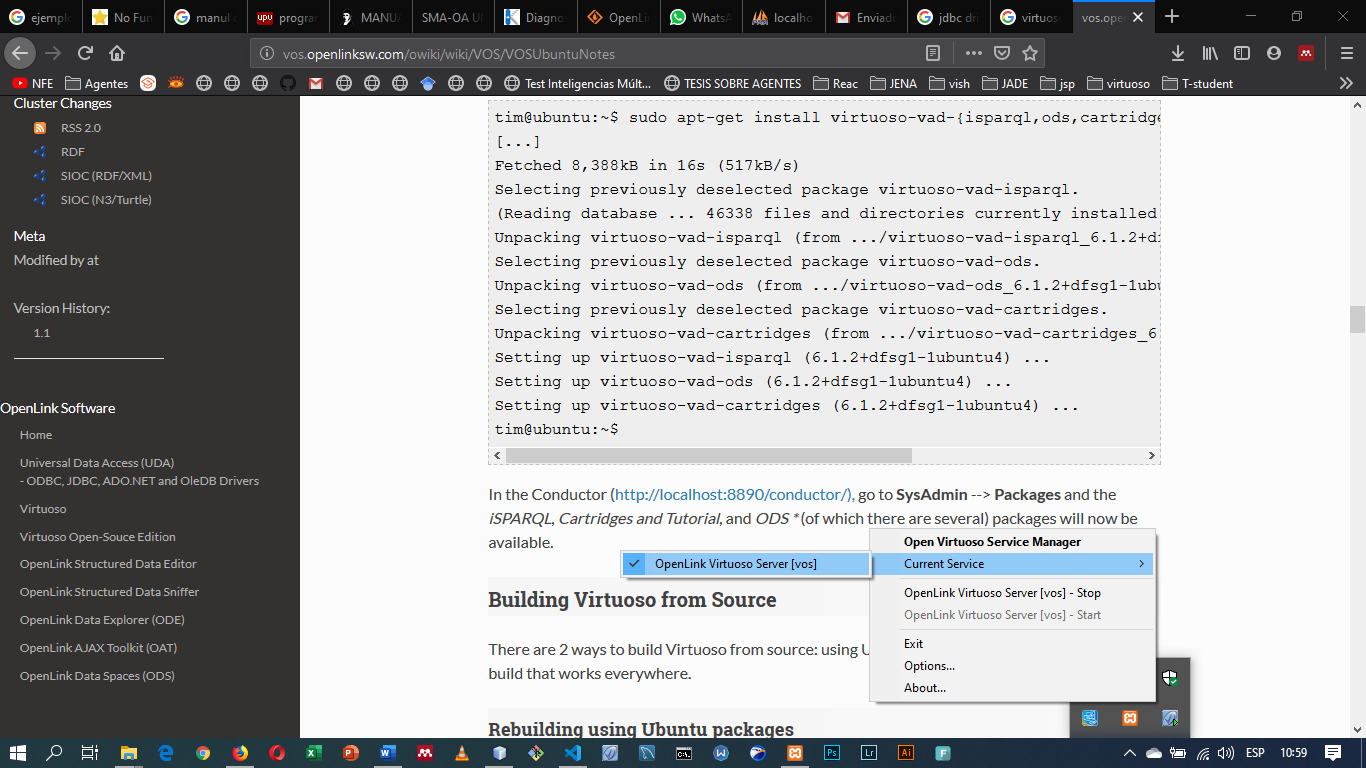


Figura 10. OpenLink Virtuoso Server [vos]

1. Una vez se inicia el servicio, ingresamos en nuestro navegador la dirección: <http://localhost:8890/> en la cual se ejecuta la instancia del servidor Virtuoso.

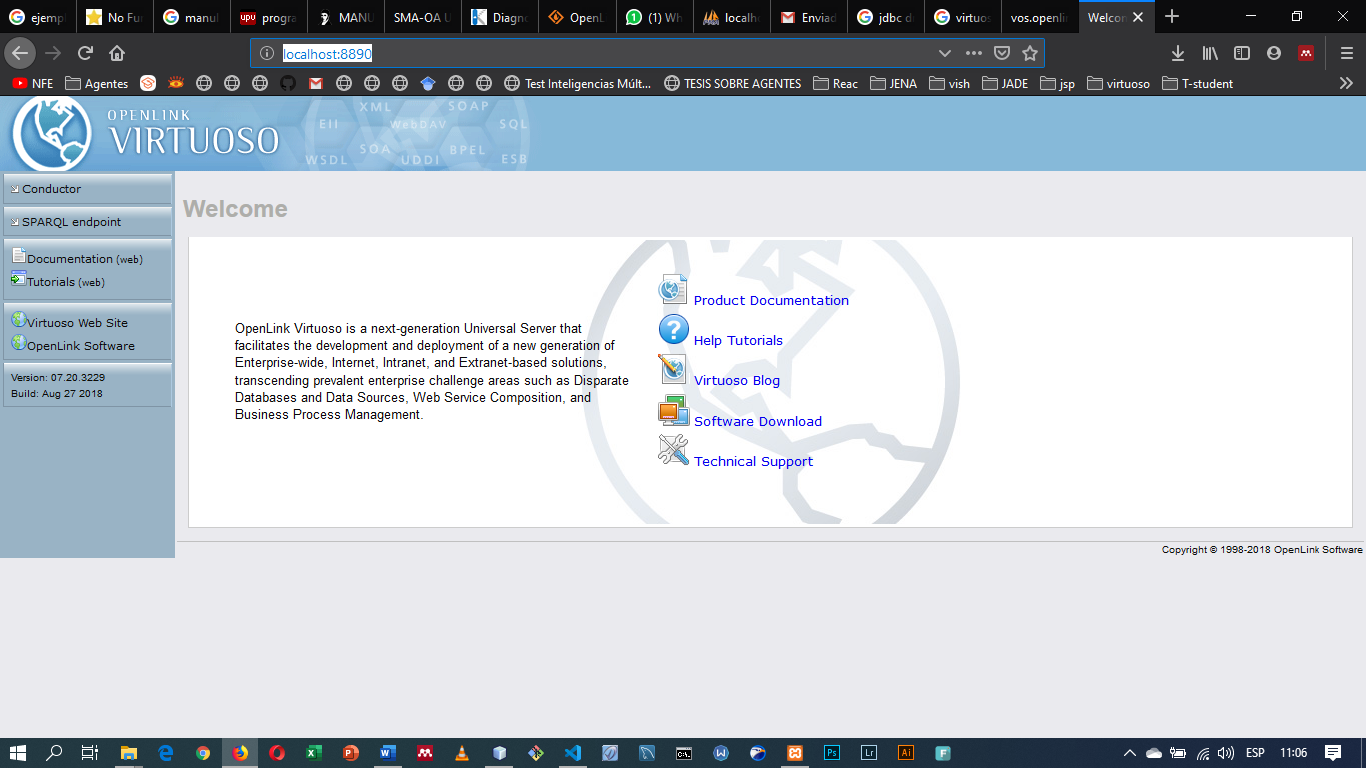


Figura 11. Panel de Administración de Virtuoso.

1. Clic en **Conductor,** el sistema pedirá un usuario y contraseña para el administrador; el usuario y contraseña por defecto es **dba** en ambos casos. Ingresamos y nos mostrará la ventana de administración.

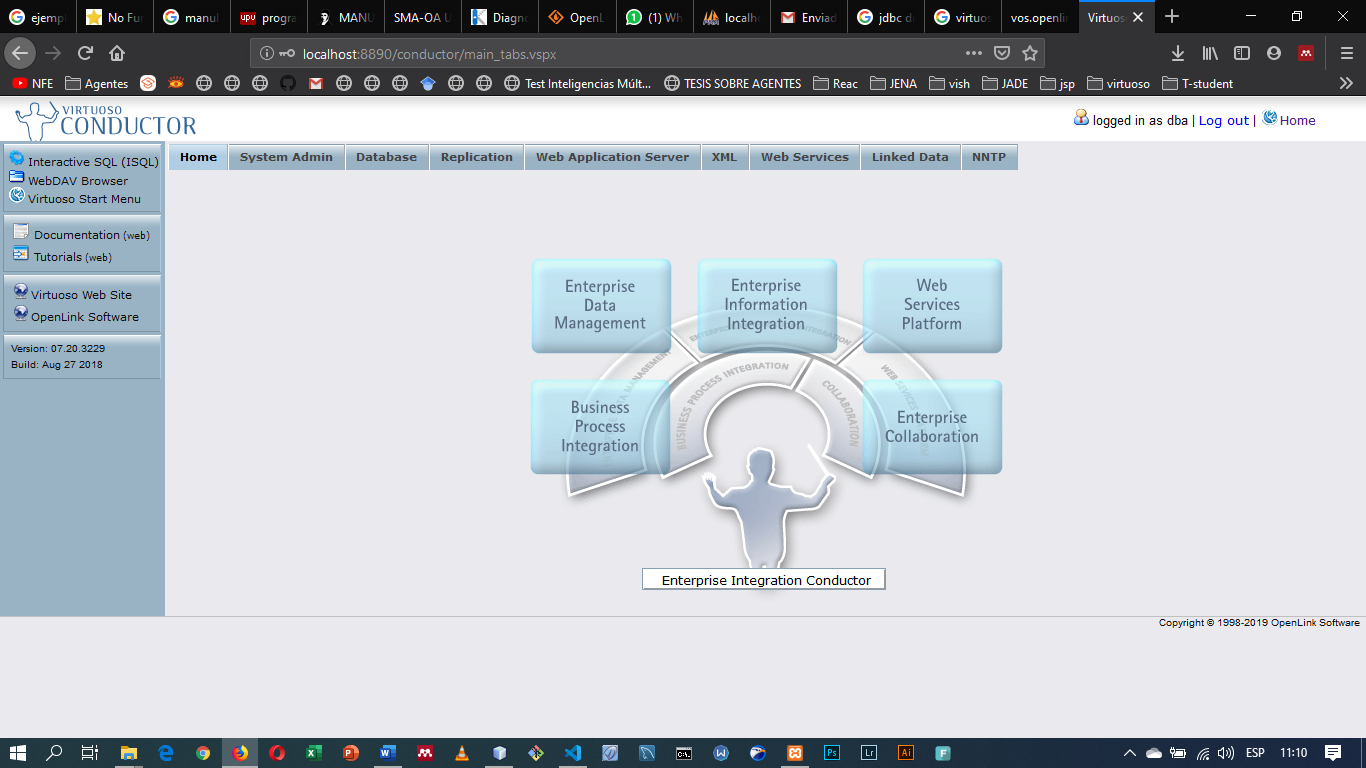


Figura 12. Virtuoso Conductor.

1. Nos dirigimos a la pestaña **Linked Data** y elegimos la opción **Quad Store Upload**.

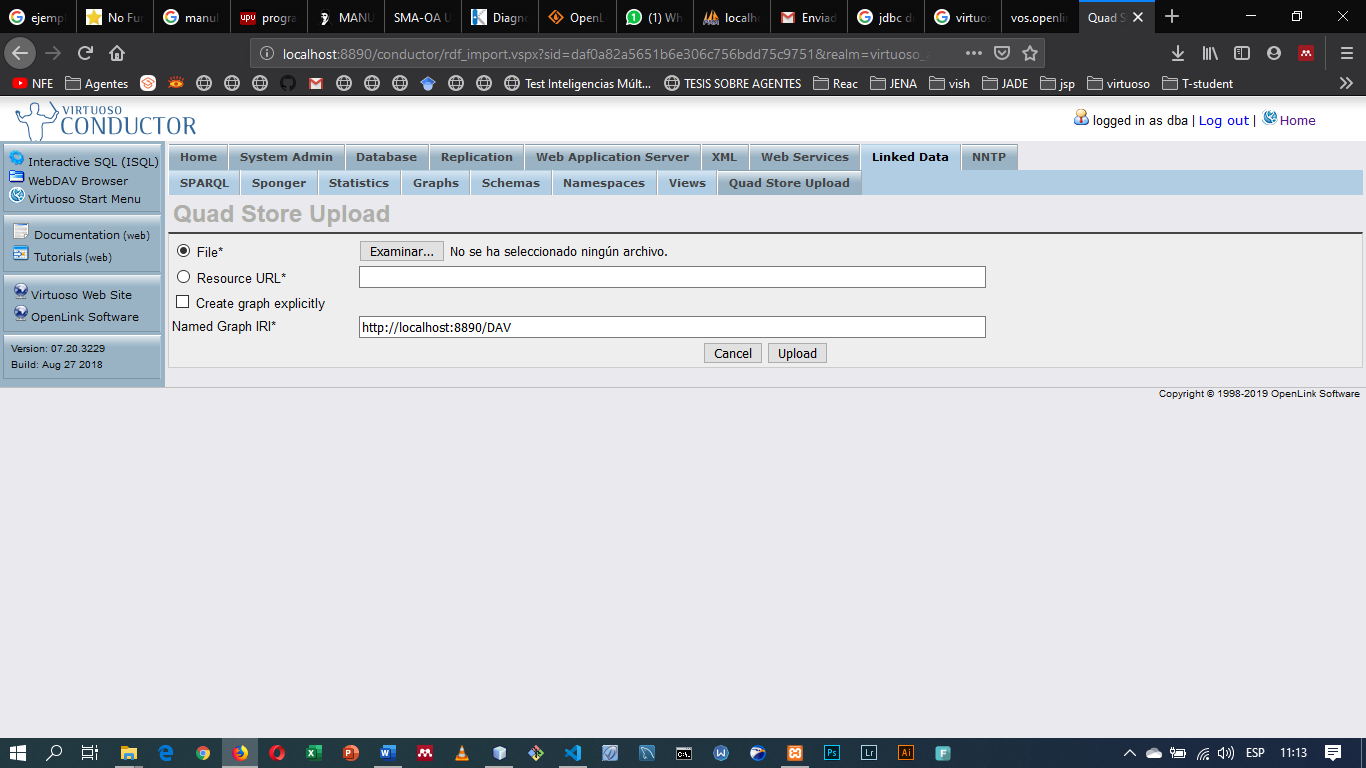


Figura 13. Linked Data.

1. Ahora se debe crear dos Graph o instancias de la ontología. Para los objetos de aprendizaje se carga el archivo **OntologíaBase.owl,** puede descargarla desde el directorio GitHub[[1]](#footnote-1) del Proyecto, seguidamente en **Named Graph IRI** se escribe: <http://LearningObjects> como lo muestra la Figura 14.

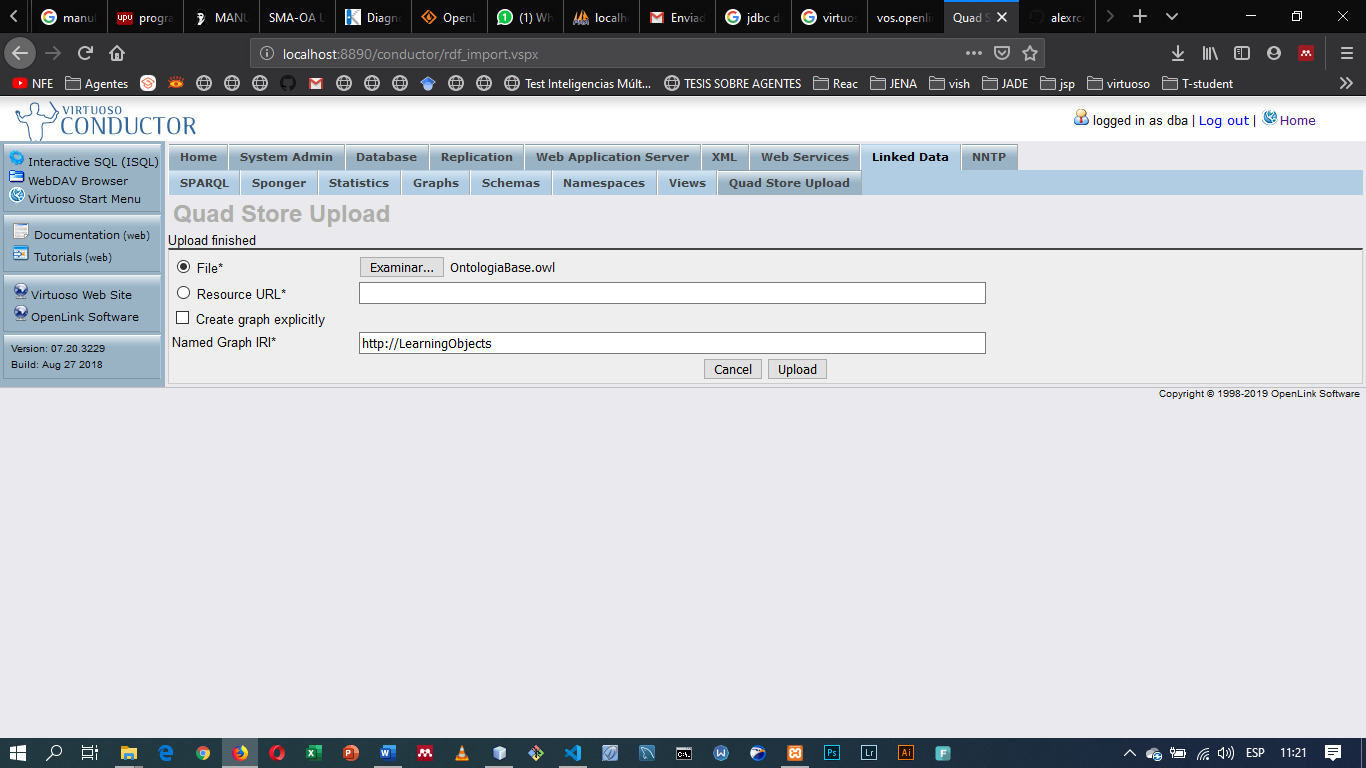


Figura 14. Creación de la Tabla (Graph) LearningObjects.

1. Para la Graph de los Perfiles se utiliza el mismo archivo (**OntologíaBase.owl**) y se lo nombre “**Perfiles**” como lo indica la Figura 15.

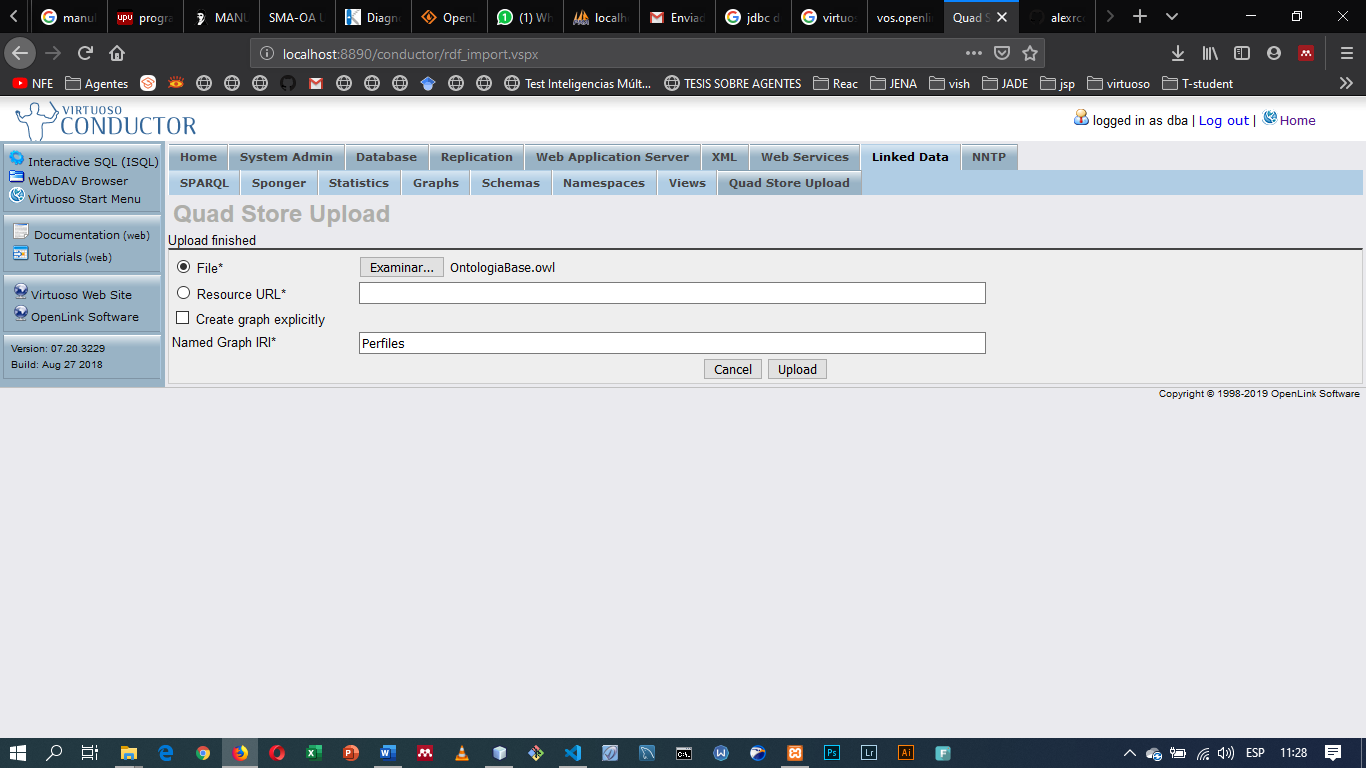


Figura 15. Creación de la Tabla (Graph) Perfiles.

1. Se revisa que las Graph se hayan creado, para ellos seleccionamos la pestaña **Graphs > Graph** y observamos que las dos gráficas creadas se encuentren en la lista.

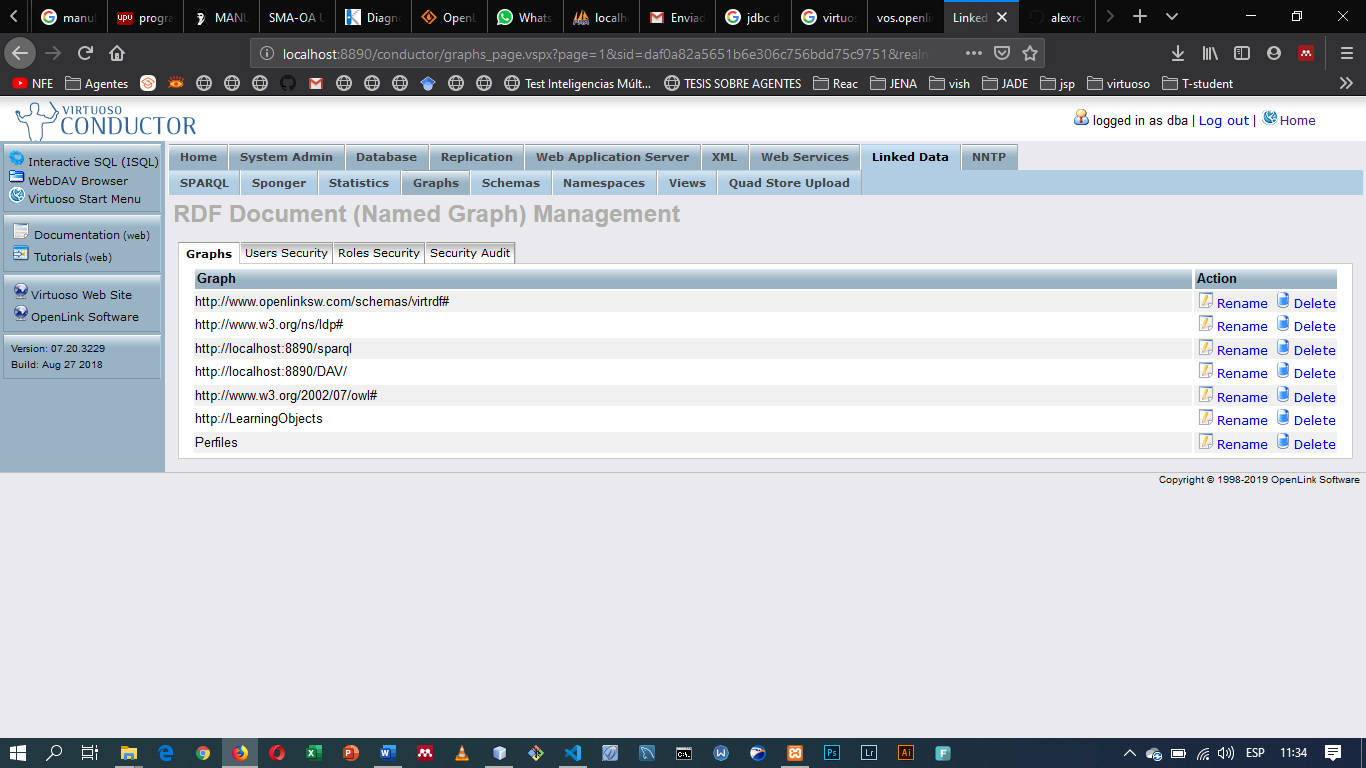


Figura 16. Graphs de LeanringObjects y Perfiles creadas.

1. Una vez creadas las Graphs, se debe realizar la conexión del proyecto con Virtuoso, esta conexión se realiza en la clase **Virtuoso** y es necesario descargar e instalar el Virtuoso JDBC 4 Driver JAR file, disponible en:

<http://vos.openlinksw.com/owiki/wiki/VOS/VOSDownload> .

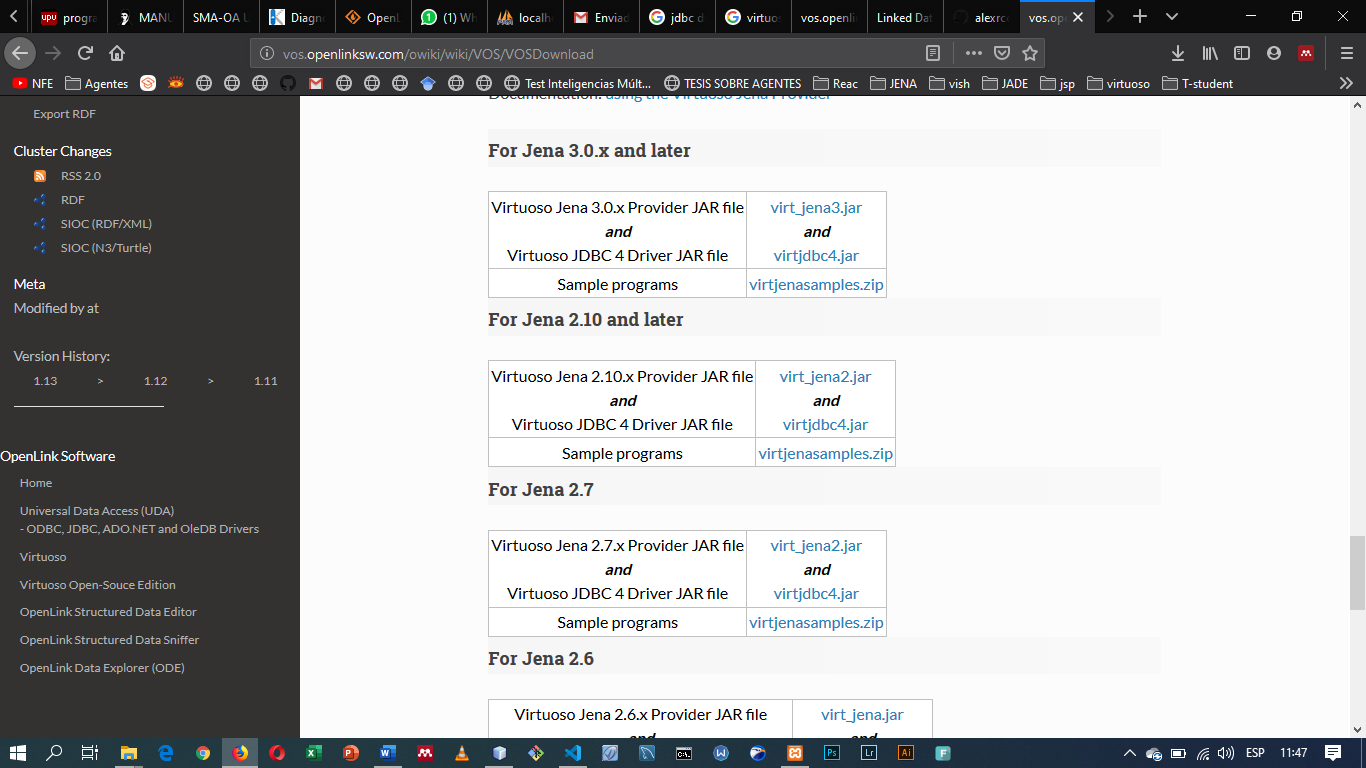


Figura 17. Virtuoso JDBC 4 Driver JAR file.

1. Instalamos los dos archivos descargados en el paso 10, en las librerías del proyecto.

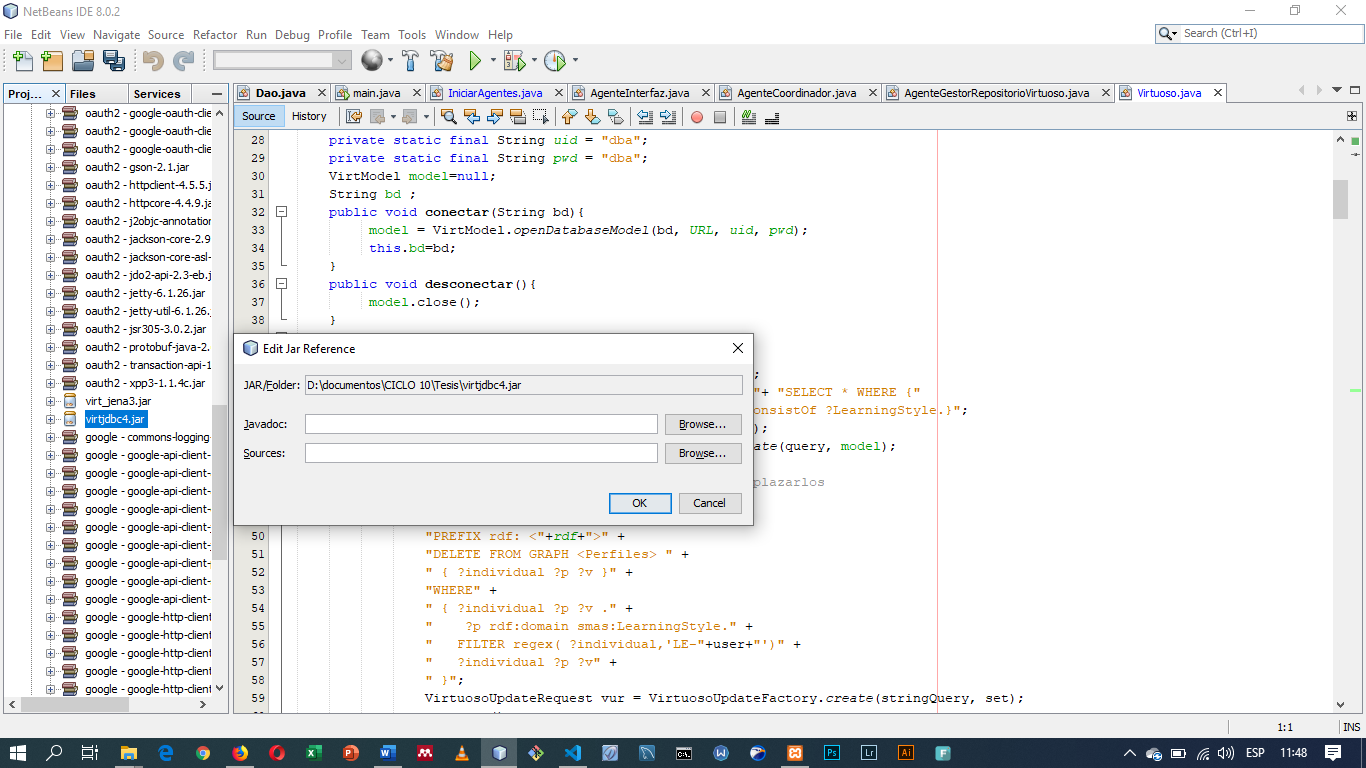


Figura 18. Dependencias de Virtuoso.

1. Ahora procedemos a realizar la conexión con el servidor en la case **Virtuoso,** como lo muestra la Figura 19.

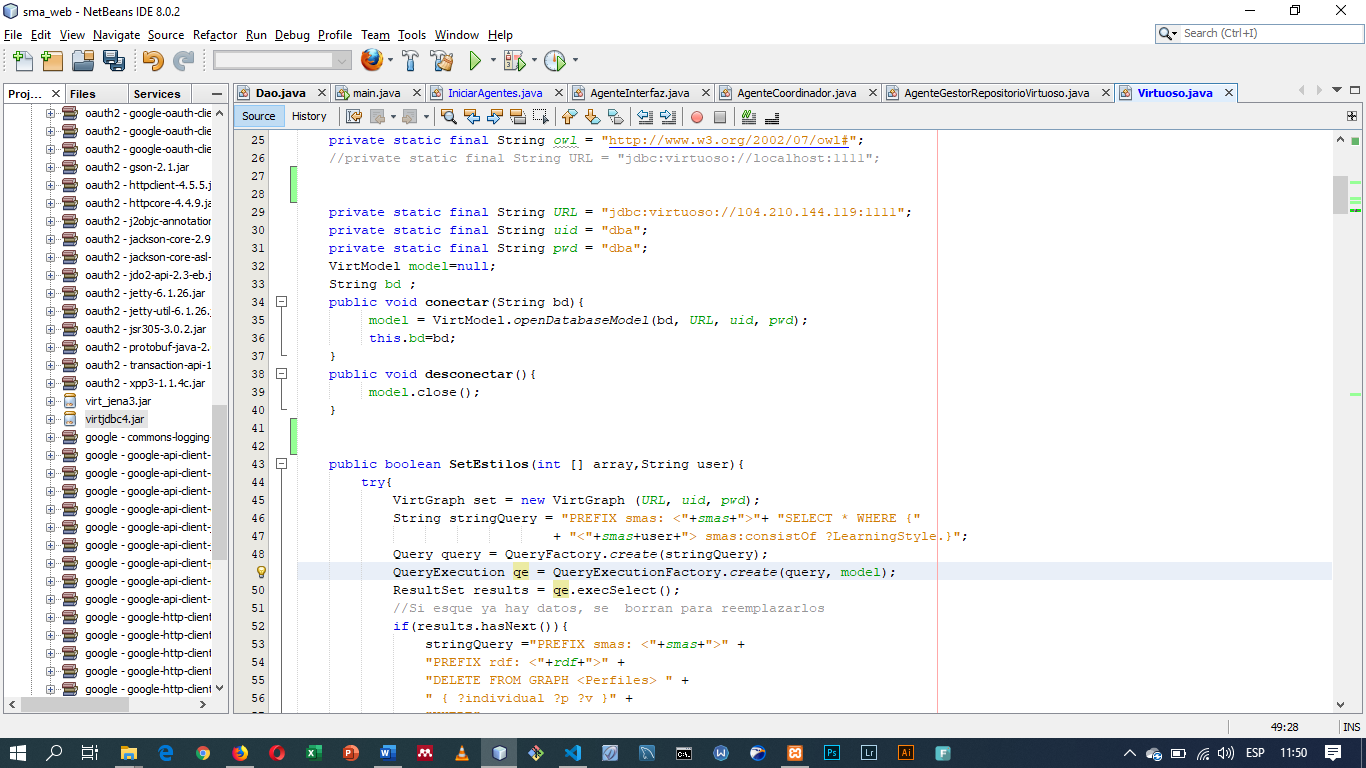


Figura 19. Código de conexión con el servidor Virtuoso.

1. Se debe mencionar que el servidor donde este instalado Virtuoso, debe tener habilitados los puertos 8890 y 1111, caso contrario no se podrá conectar.

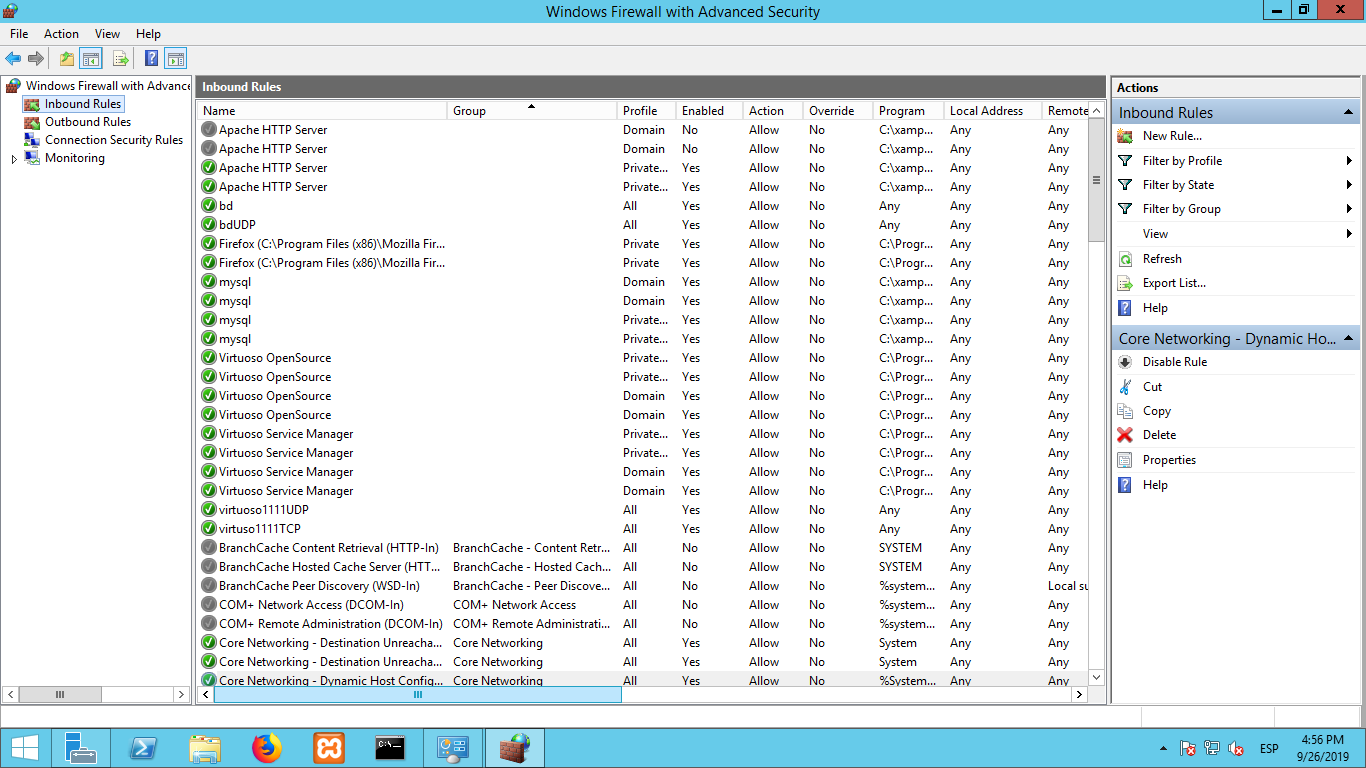


Figura 20. Reglas de firewall para acceso de Virtuoso.

# INSTALACIÓN DEL SMA EN EL SERVIDOR WEB

Como ya está especificado en los requerimientos previos, se debe disponer de un servidor que contenga Apache Tomcat, en el presente ejemplo se muestra la instalación en un servidor Web de Microsoft Azure (App Service):

1. Al ser un proyecto basado en el lenguaje de programación Java, se debe generar un archivo **.WAR**
2. Para generar el archivo WAR, en NetBeans, se debe abrir la pestaña **Run,** en la opción **Set Main Project** seleccionamos el nombre de nuestro proyecto (sma\_web en este caso específico);

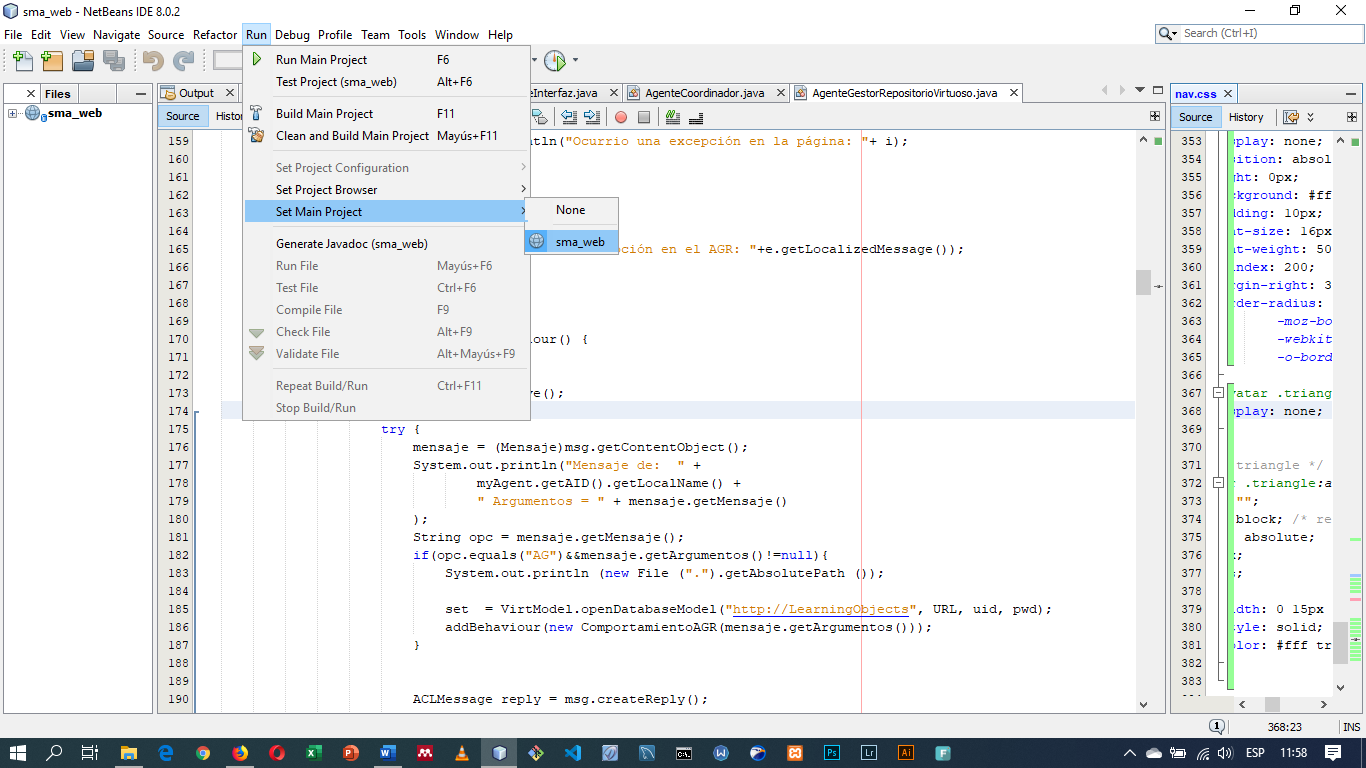


Figura 21. Seleccionar sma\_web como el proyecto principal.

1. Una vez seleccionado el proyecto en la misma ventana, seleccionamos la opción **Clean and Build Main Project**, y esperamos a que el proceso culmine.

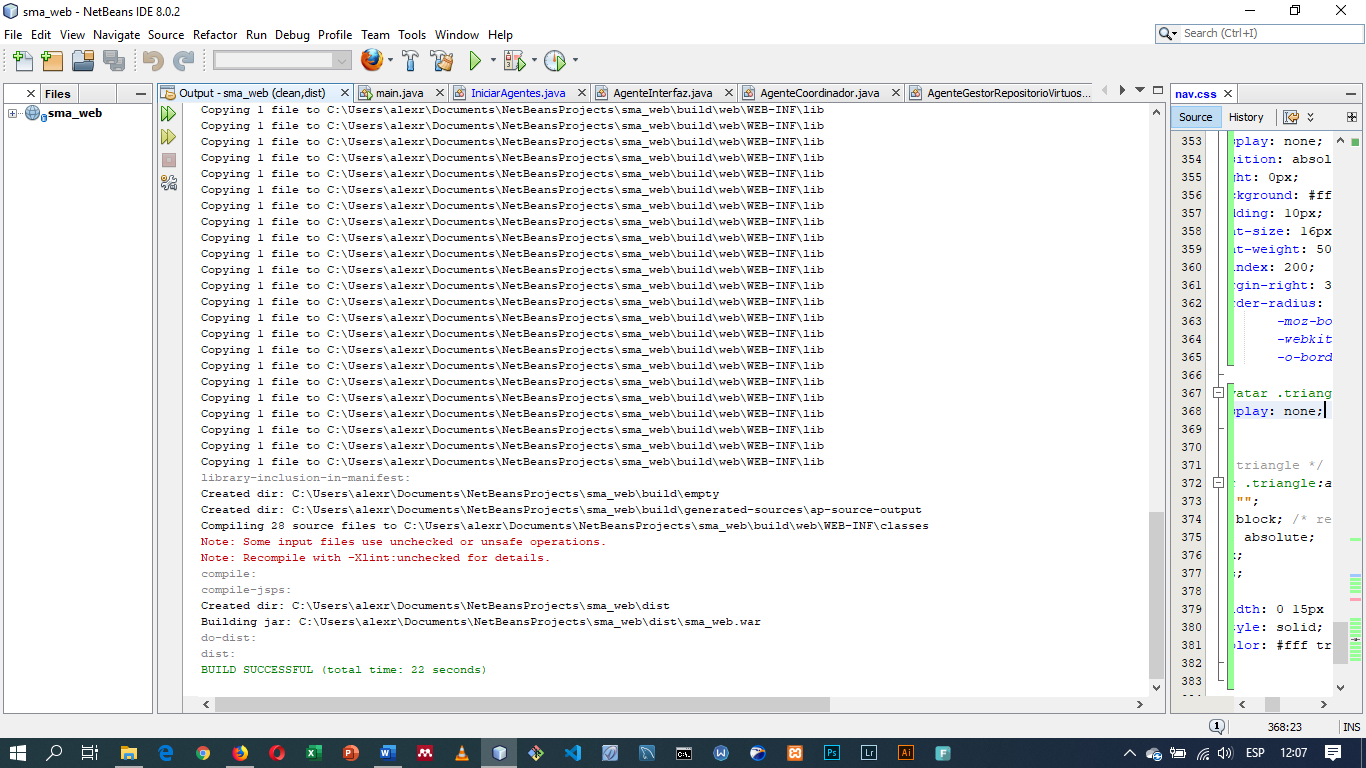


Figura 22. Mensaje de confirmación al generar el archivo WAR.

1. Una vez culminado el proceso de compilación, el archivo **WAR** se encuentra en la carpeta **dist** del proyecto Java.

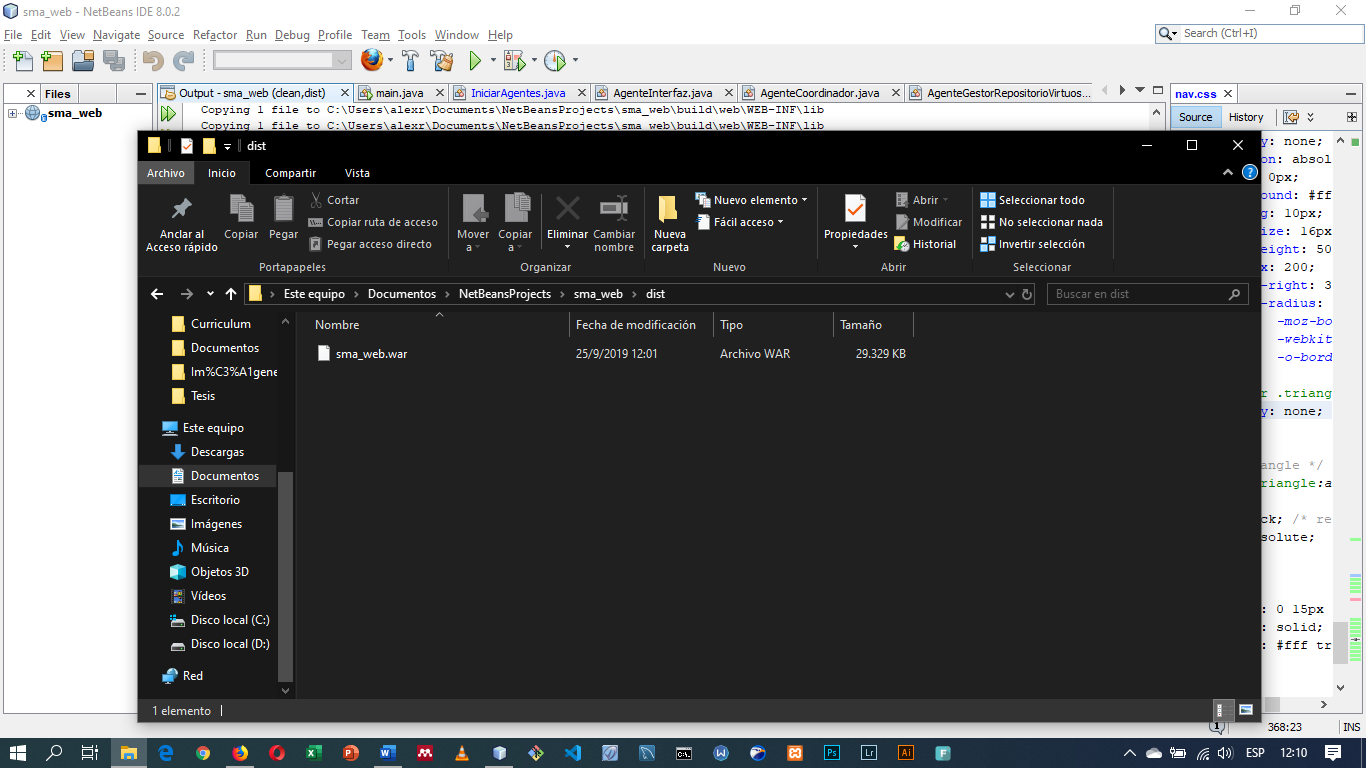


Figura 23. Archivo sma\_web.war.

1. El siguiente paso consiste en descomprimir el archivo **WAR** en el servidor Web, para lo cual existen varios métodos y depende de las características del servidor. Para este caso específico, *MS- Azure* contiene una herramienta de gestión de archivos llamada **Kudu.**
2. Navegamos hasta la carpeta **wwwroot/webapps/ROOT/** en el servidor Web y soltamos el archivo WAR para que se descomprima.

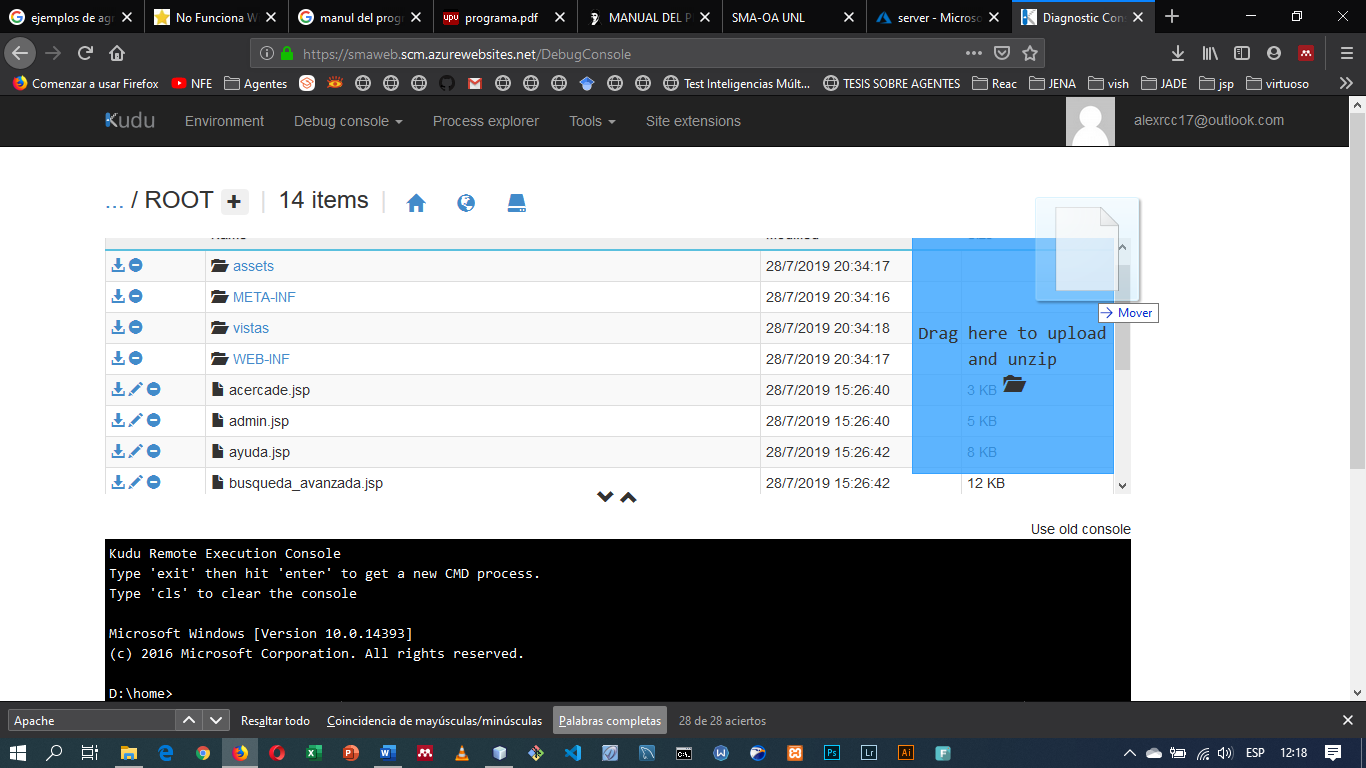


Figura 24. Descompresión del archivo WAR en webapps/ROOT.

1. Depende de su gestor de archivos, usted deberá realizar una operación similar. Y una vez se descomprima todos los archivos, podrá ver implementado el sistema en el servidor web.

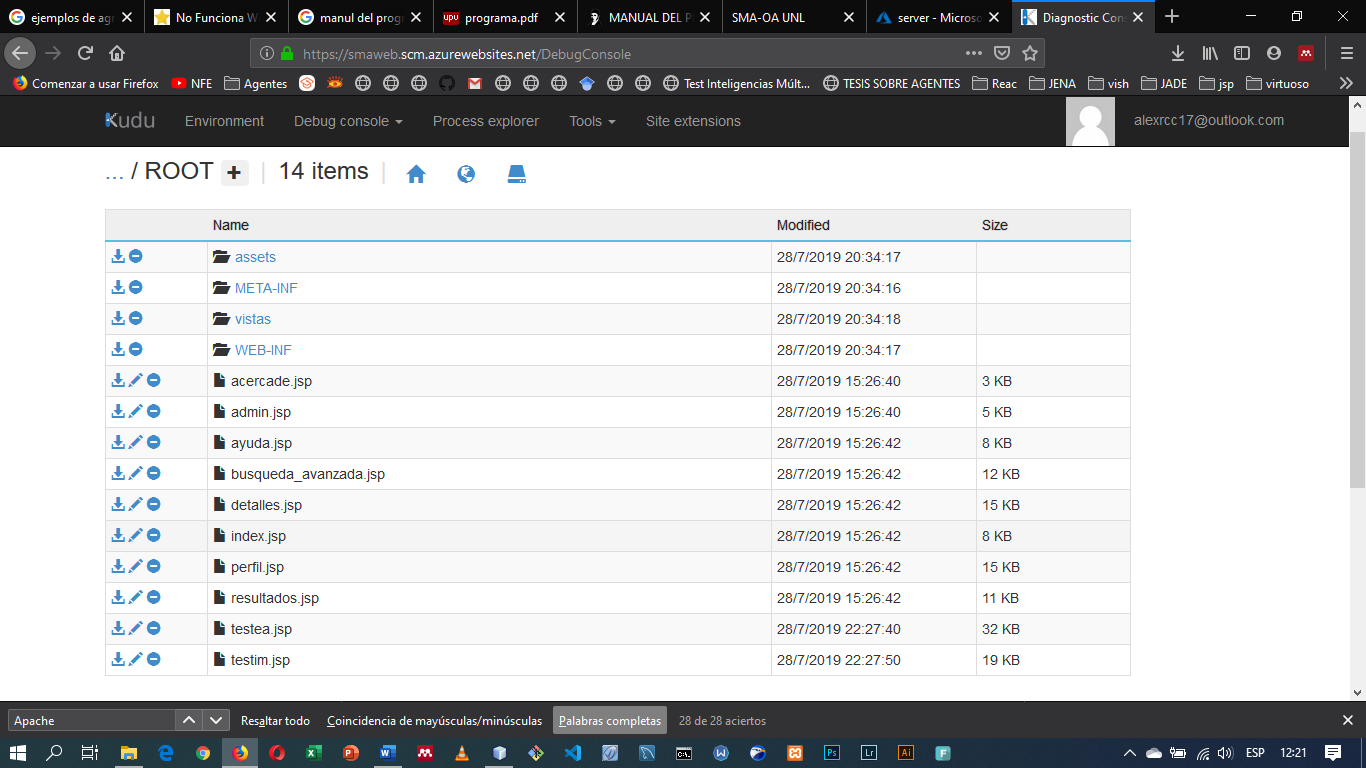


Figura 25. Despliegue del sistema en el servidor.

1. Ahora debe configurar su ruta de inicio (index) en el archivo WEB-INF/web.xml, e ingresar a su dirección Web para visualizar el sistema.

# INGRESAR

Una vez instalados el proyecto y todas sus dependencias, se puede ingresar a la dirección [www.smaweb.azurewebsites.net](http://www.smaweb.azurewebsites.net) y probar el sistema.

1. <https://github.com/alexrcc/Sistema-Multiagente-de-Recomendacion> [↑](#footnote-ref-1)