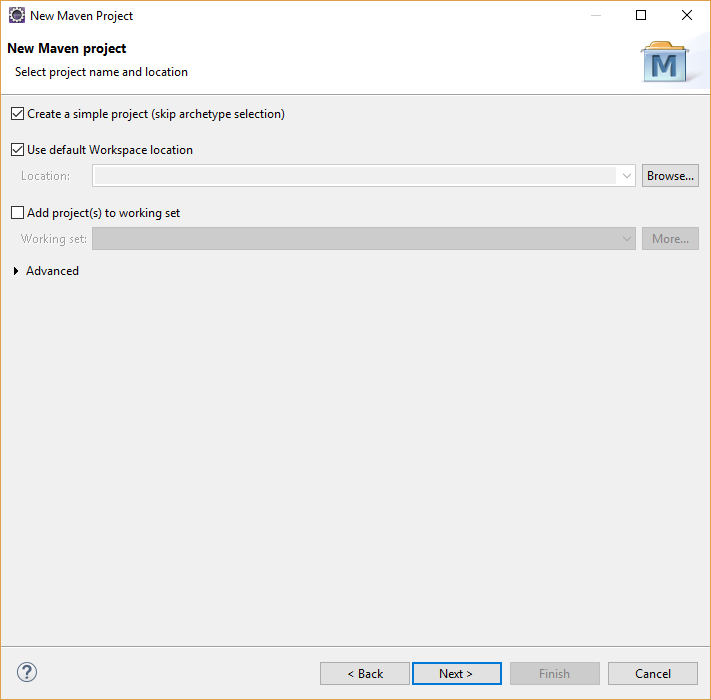
# Crear un Proyecto en ECLIPSE – NEON usando MAVEN – JSF –

Paso 1:

Instalar Eclipse Java EE

Paso 2:

New Project 🡪Other🡪 Maven 🡪Maven Project

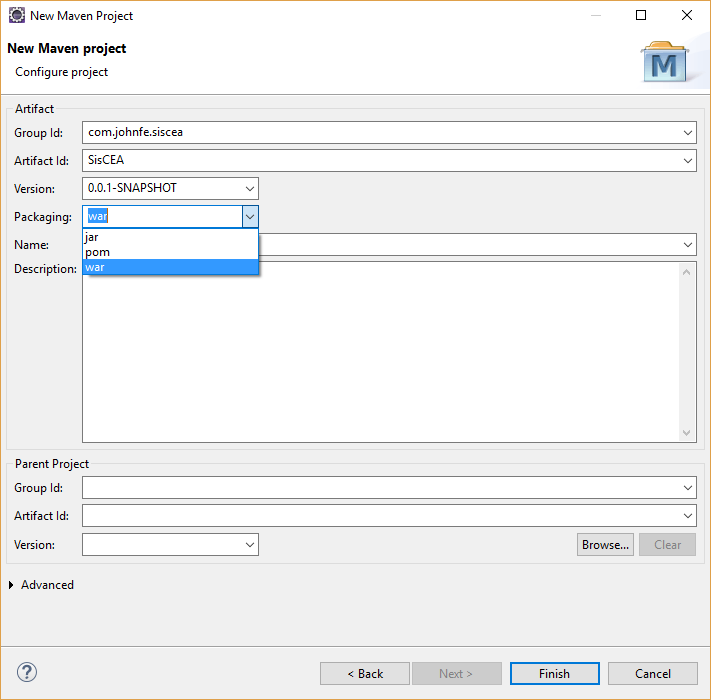


Paso 3:

Group Id 🡪 Se escribe el package principal del Proyecto

Artifact ID 🡪 Se escribe el nombre del Proyecto

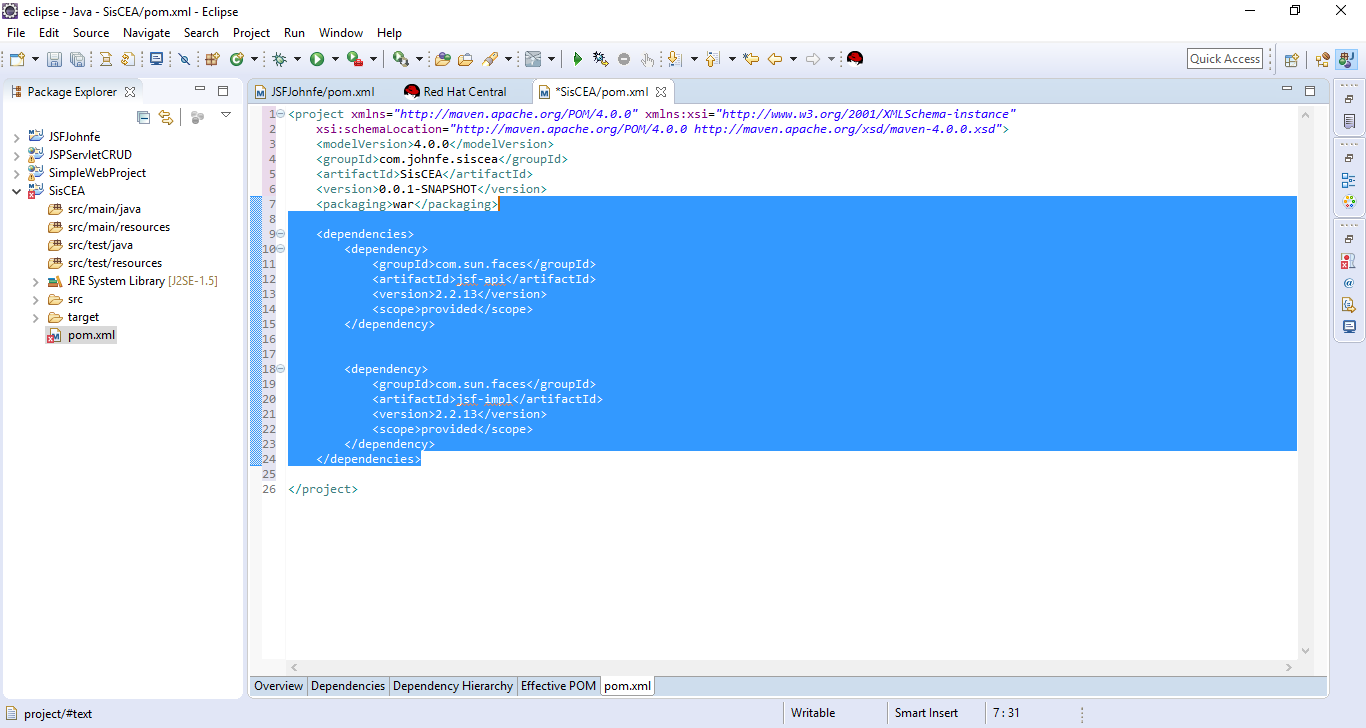
Packaging 🡪 Se selecciona “war”, ya que es un proyecto WEB, luego se da clic en finish



Paso 3:

Abrimos el “pom.xml” y agregamos las dependencias para agregar JSF al proyecto.

Librerías JSP-api y JSF-impl



<dependencies>

<dependency>

<groupId>com.sun.faces</groupId>

<artifactId>jsf-api</artifactId>

<version>2.2.13</version>

<scope>provided</scope>

</dependency>

<dependency>

<groupId>com.sun.faces</groupId>

<artifactId>jsf-impl</artifactId>

<version>2.2.13</version>

<scope>provided</scope>

</dependency>

</dependencies>

<scope>provided</scope> 🡪 Si se agrega esta linea significa que la dependencia va a ser proveída por el servidor de aplicaciones, de lo contrario va a ser descargada.

Luego agregamos las siguientes líneas debajo de las dependencias para indicar con que JDK se va a compilar el proyecto, en este caso con la versión 1.8

<build>

<finalName>SisCEA</finalName>

<plugins>

<plugin>

<groupId>org.apache.maven.plugins</groupId>

<artifactId>maven-compiler-plugin</artifactId>

<version>3.1</version>

<configuration>

<source>1.8</source>

<target>1.8</target>

</configuration>

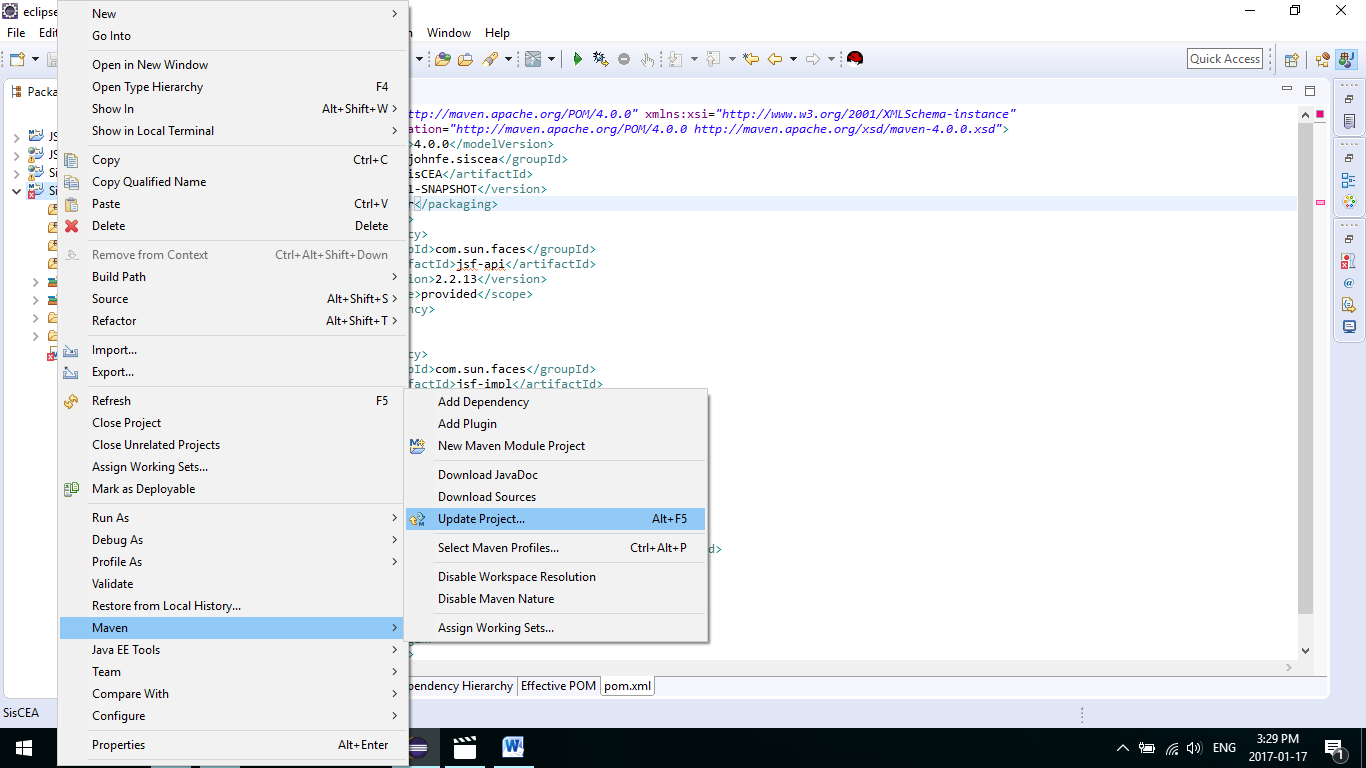
</plugin>

</plugins>

</build>

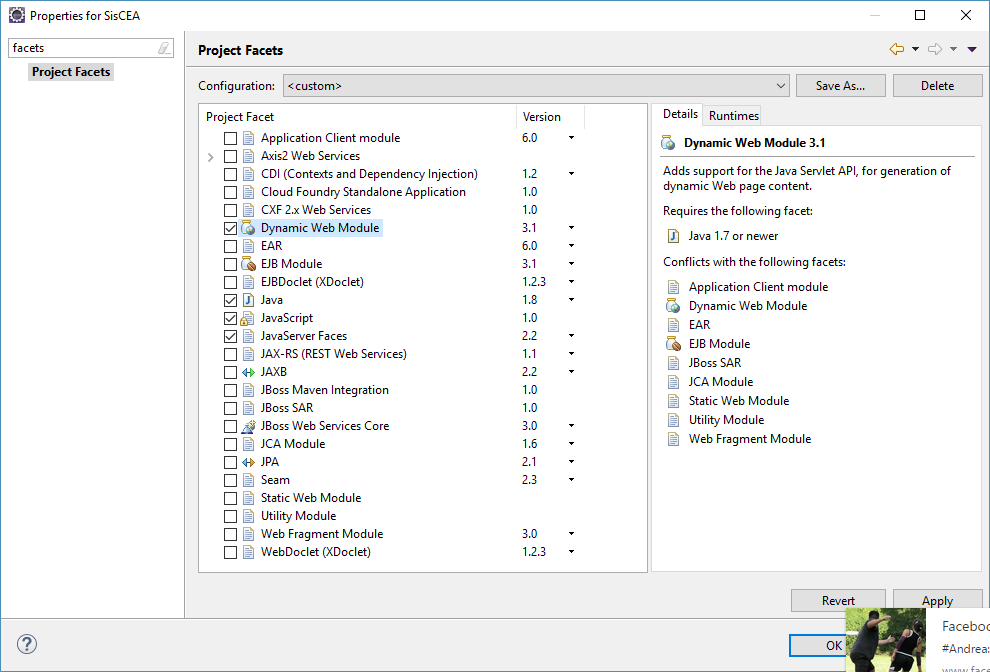
Paso 4:

Clic derecho en el proyecto🡪 Maven🡪Update Project🡪Ok



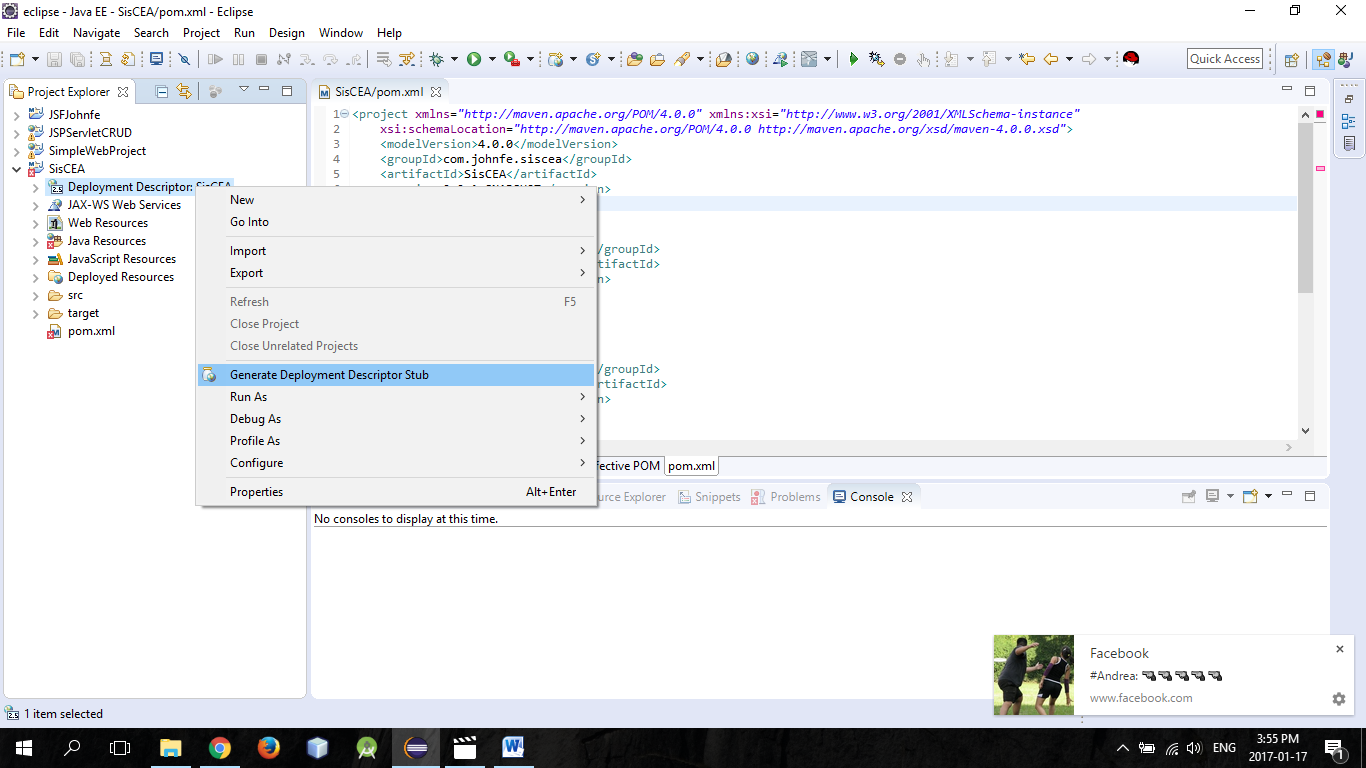
Paso 5:

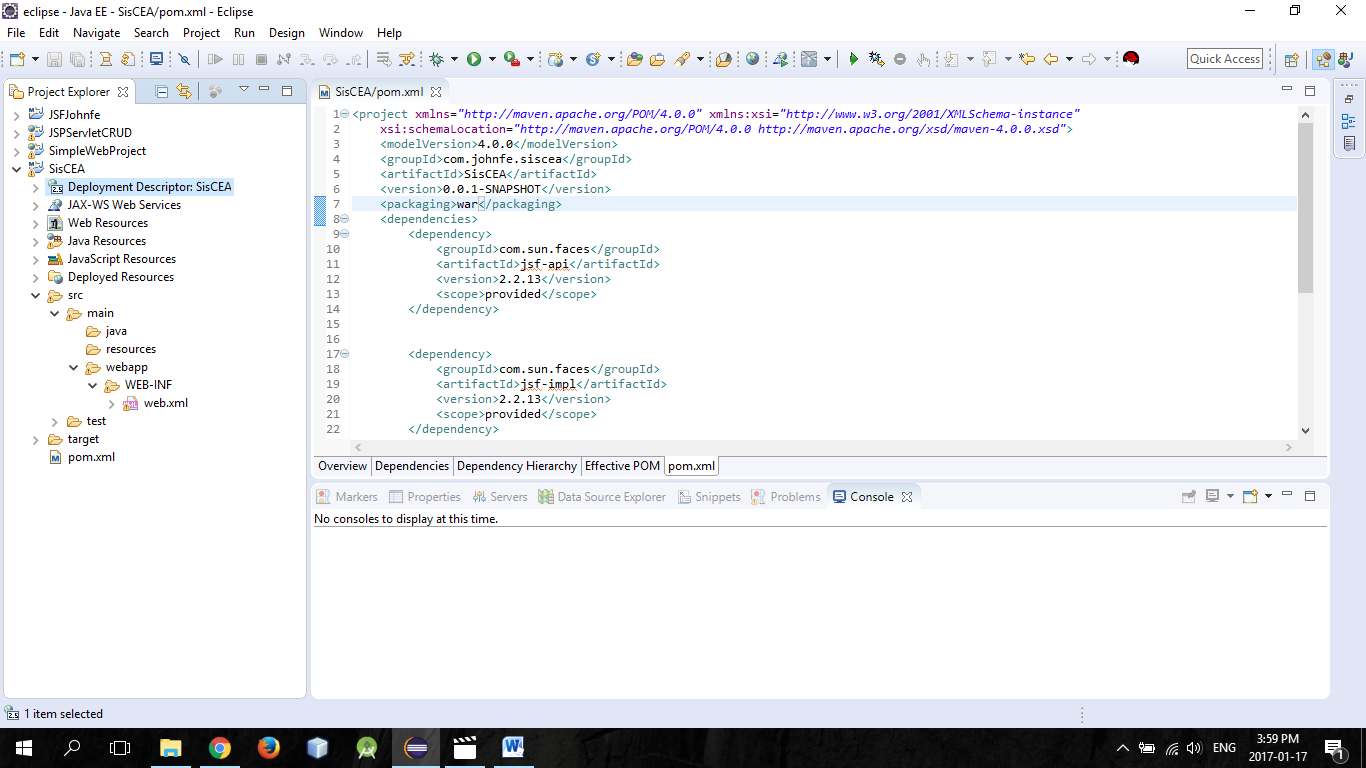
De nuevo clic en propiedades del proyecto luego



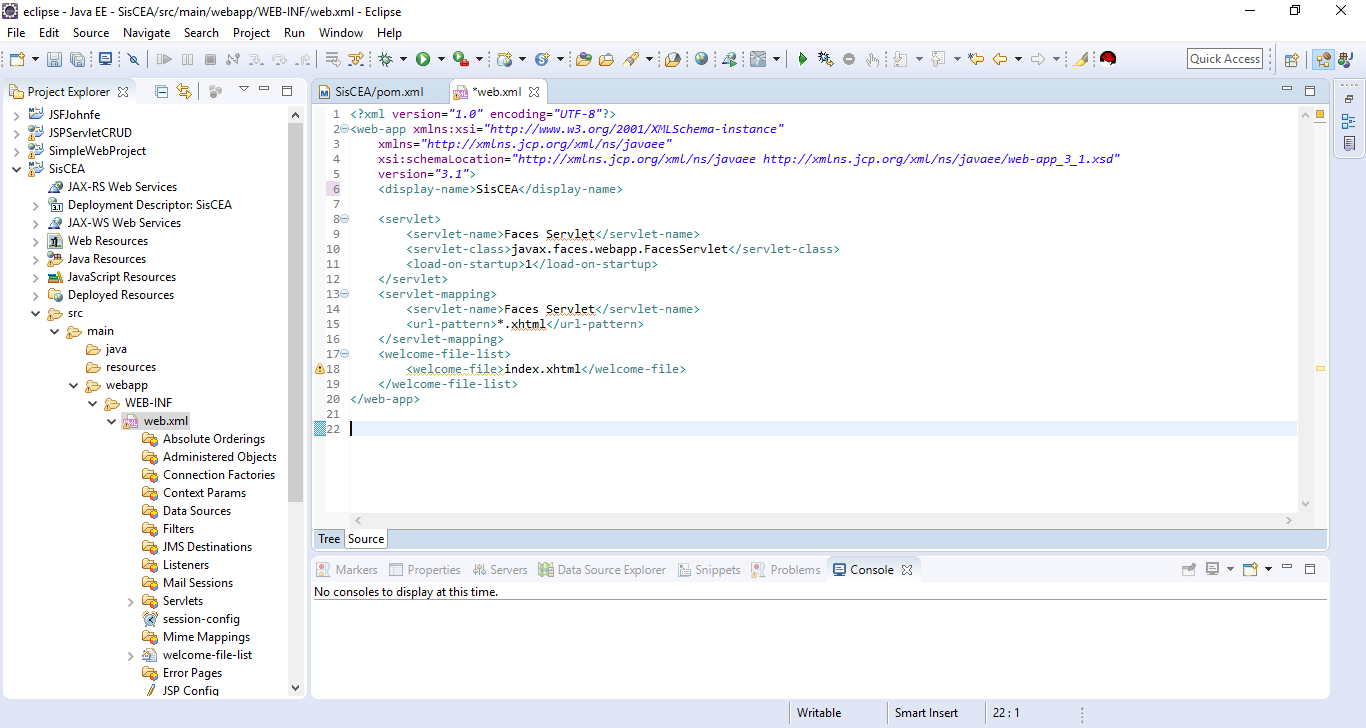
Filtramos por “Facets” y dejamos tal cual como se ve en la imagen, luego vamos a Runtimes y seleccionamos nuestro servidor de aplicaciones.

Luego vamos a Deployment Descriptor y le damos clic en “Generate Deployment Descriptor stub”, con esto creamos nuestro web.xml (Perspectiva de Java EE en eclipse) y en el web.xml lo dejamos de la siguiente manera





El web.xml debe quedar de la siguiente manera:



<?xml version="1.0" encoding="UTF-8"?>

<web-app xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"

xmlns="http://xmlns.jcp..org/xml/ns/javaee"

xsi:schemaLocation="http://xmlns.jcp..org/xml/ns/javaee hhttp://xmlns.jcp..org/xml/ns/javaee/web-app\_3\_1.xsd"

version="3.1">

<display-name>MavenJSF</display-name>

<servlet>

<servlet-name>Faces Servlet</servlet-name>

<servlet-class>javax.faces.webapp.FacesServlet</servlet-class>

<load-on-startup>1</load-on-startup>

</servlet>

<servlet-mapping>

<servlet-name>Faces Servlet</servlet-name>

<url-pattern>\*.xhtml</url-pattern>

</servlet-mapping>

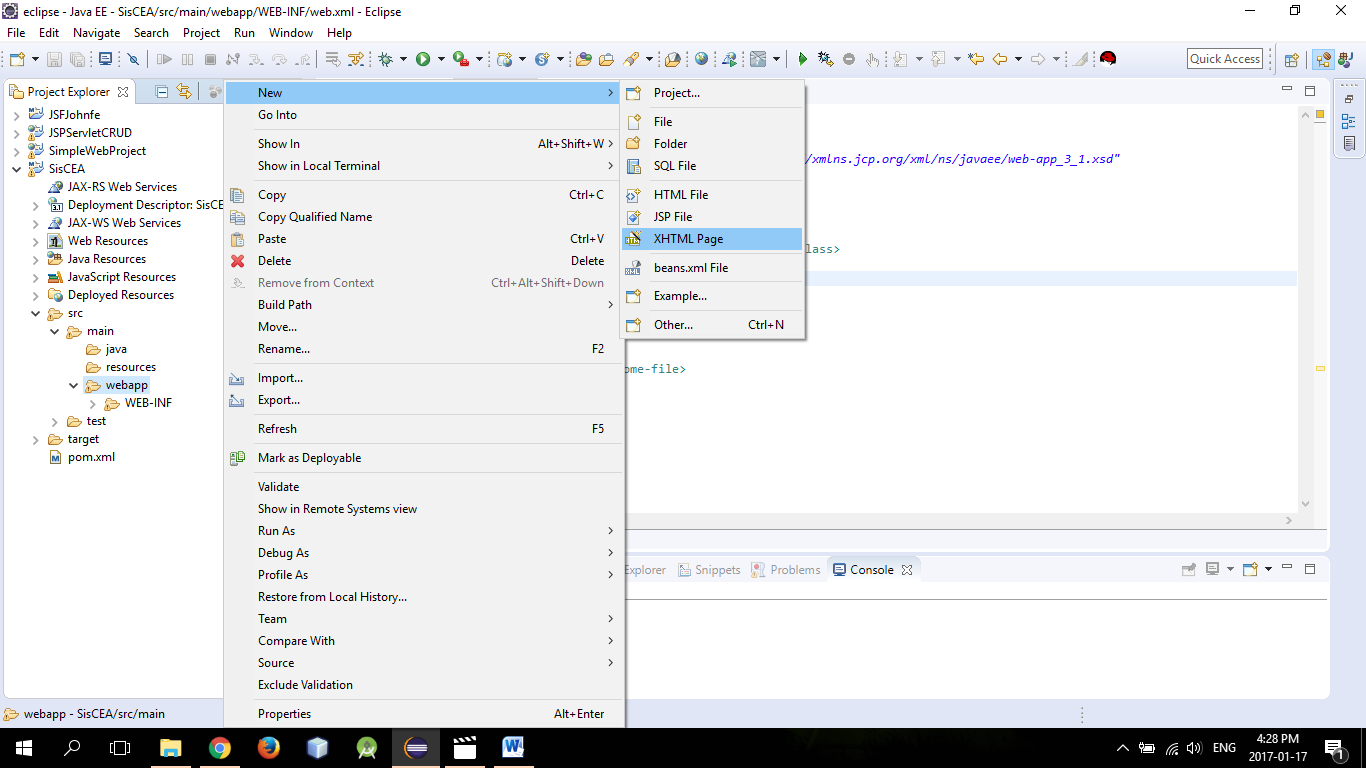
<welcome-file-list>

<welcome-file>index.xhtml</welcome-file>

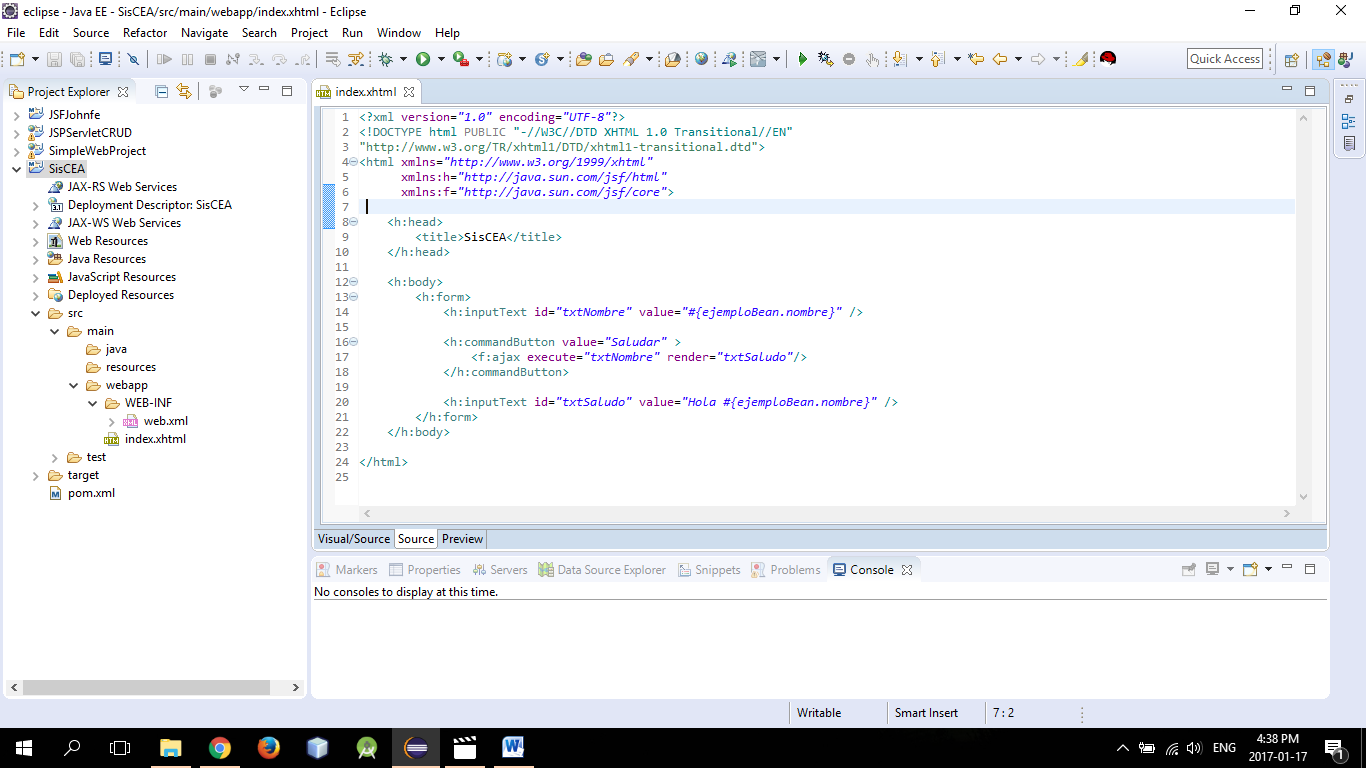
</welcome-file-list>

</web-app>

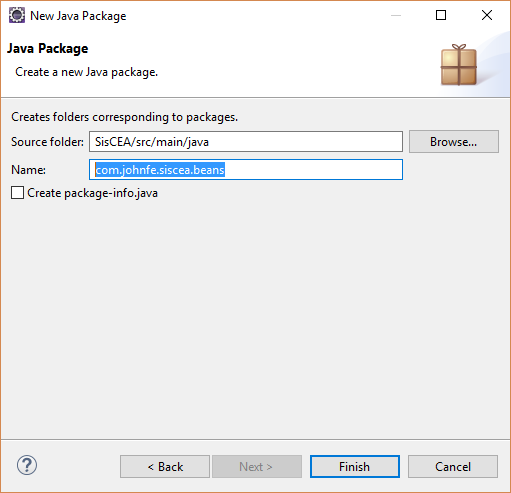
Una vez que tengamos nuestro web.xml de esa manera, seguimos con la creación de nuestro archivo index.xhtml dentro de la carpeta webapp



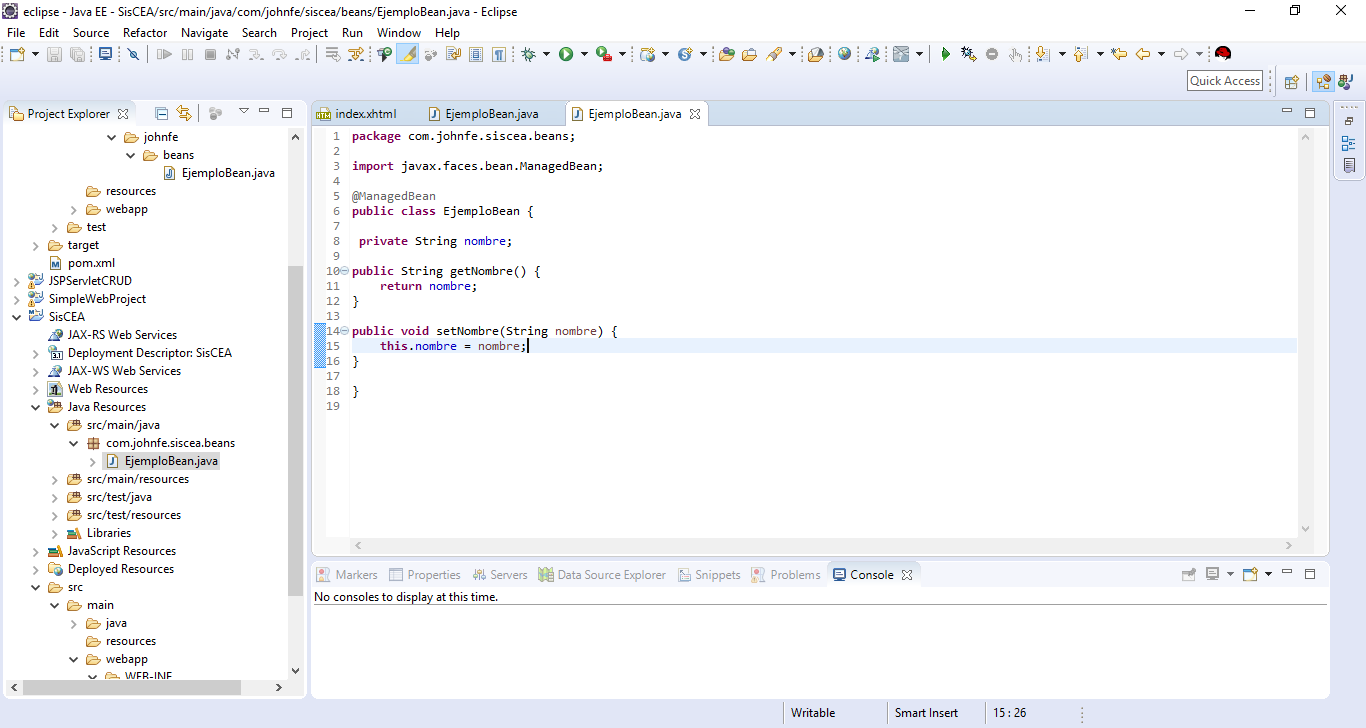
Dejamos nuestro index.xhtml asi



Luego creamos un paquete llamado “beans” dentro de la carpeta “java”



Luego dentro del packete creamos una clase dentro del package y la transformamos en un bean y debe quedar asi:



Un bean es la representación de la clase Java para la vista xhtml. Corremos nuesta aplicación

