**SERVICIO DE CONSULTORÍA PARA HABILITAR PLATAFORMA SOFTWARE RELACIONADA AL MODELO DE INTEGRACIÓN CONTINUA PARA REALIZAR ANÁLISIS ESTÁTICO DEL CÓDIGO FUENTE DE LOS SISTEMAS DEL OSCE BASADO EN TECNOLOGÍA JAVA**

**Orden de Servicio Nº 0000849-2017-OSCE**

**MANUAL DE USO DE LA PLATAFORMA SOFTWARE HABILITADA**

**Elaborado por: Ing. Yhan Fransua Mandros Poblet**

**10 de Julio del 2017**

**HISTORIAL DE REVISIONES**

|  |  |  |  |
| --- | --- | --- | --- |
| **Fecha** | **Versión** | **Autor** | **Descripción** |
| 28/04/2017 | 1.0 | Yhan Fransua Mandros Poblet | Versión inicial |

**Tabla de Contenido**

[**1.** **LINEAMIENTOS DE USO DE PLATAFORMA SOFTWARE HABLITADA** 4](#_Toc500215157)

[**1.1.** **SERVICIOS DE GESTION DE CODIGO FUENTE** 4](#_Toc500215158)

[**1.1.1.** **Entorno de desarrollo** 4](#_Toc500215159)

[**1.1.2.** **Entorno de servidores** 4](#_Toc500215160)

[**1.1.3.** **Alcance técnico de la herramienta** 4](#_Toc500215161)

[**1.1.4.** **Estructuras de proyectos recomendadas como buenas prácticas.** 5](#_Toc500215162)

[**1.2.** **GESTION DE REPOSITORIOS** 5](#_Toc500215163)

[**1.3.** **ANALISIS ESTATICO DE CÓDIGO FUENTE** 5](#_Toc500215164)

[**2.** **ENTORNO DE DESARROLLADOR CON PLATAFORMA SOFTWARE HABILITADA** 5](#_Toc500215165)

[**2.1.** **INTEGRACIÓN CON SVN** 5](#_Toc500215166)

[**2.1.1.** **Instalar Tortoise SVN** 5](#_Toc500215167)

[**2.1.2.** **Instalar plugin Eclipse** 6](#_Toc500215168)

[**2.1.3.** **Administrar repositorios en servidor** 9](#_Toc500215169)

[**2.2.** **INTEGRACIÓN CON NEXUS** 9](#_Toc500215170)

[**2.2.1.** **Instalación de maven** 9](#_Toc500215171)

[**2.2.2.** **Configurar maven Eclipse** 12](#_Toc500215172)

[**2.3.** **INTEGRACIÓN CON SONARQUBE** 14](#_Toc500215173)

[**2.3.1.** **Instalar SonarQube Scanner** 14](#_Toc500215174)

[**2.3.2.** **Configurar Eclipse con SonarQube** 14](#_Toc500215175)

[**2.3.3.** **Crear proyectos SonarQube** 15](#_Toc500215176)

[**2.3.4.** **Personalizar configuraciones en SonarQube** 15](#_Toc500215177)

[**2.3.5.** **Analizar proyectos con SonarQube Scanner** 16](#_Toc500215178)

[**2.3.6.** **Analizar proyectos con Eclipse** 16](#_Toc500215179)

[**2.4.** **INTEGRACIÓN CON SELENIUM** 17](#_Toc500215180)

[**3.** **ASPECTOS TÉCNICOS EN MIGRACIÓN DE REPOSITORIOS** 18](#_Toc500215181)

[**3.1.** **MIGRACION REPOSITORIO SVN 1.7.4 HACIA 1.9.7** 18](#_Toc500215182)

[**3.1.1.** **Migración por repositorio** 18](#_Toc500215183)

[**3.1.2.** **Migración por copia física** 18](#_Toc500215184)

[**3.1.3.** **Copia de trabajo en entorno de desarrollador** 18](#_Toc500215185)

[**3.2.** **MIGRACION REPOSITORIO NEXUS 2.1.2 HACIA *3.6.2*** 18](#_Toc500215186)

[**3.2.1.** **Migración asistida por consola de administración** 18](#_Toc500215187)

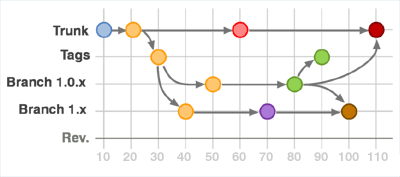
[**3.2.2.** **Personalización post-migración** 18](#_Toc500215188)

[**4.** **RELACIÓN DE LA PLATAFORMA SOFTWARE HABILITADA CON LA NTP ISO/IEC 12207** 19](#_Toc500215189)

**MANUAL DE USO DE LA PLATAFORMA SOFTWARE HABILITADA**

1. **LINEAMIENTOS DE USO DE PLATAFORMA SOFTWARE HABLITADA**
   1. **SERVICIOS DE GESTION DE CODIGO FUENTE**
      1. **Entorno de desarrollo**

* Usar la estructura TTB
* **Trunk**: Rama de desarrollo principal.
* **Tags:** Rama de gestión de versiones. Reservado para versiones cerradas, por tanto no se desarrollará sobre esta rama.
* **Branches:** Rama con evoluciones paralelas al Trunk.
* Para sincronizar el código del entorno local con el del repositorio usar lo sigueinte:
* El comando **Checkout** descargará al entorno local una copia fiel del código del repositorio. Útil para comenzar a desarrollar sobre proyectos nuevos.
* El comando **Update** descargará al entorno local únicamente las modificaciones que hayan tenido lugar desde la última sincronización. Sólo se podrá hacer esta operación si se dispone ya de una versión local del código del repositorio.
* El comando **Commit** actualizará el contenido del repositorio con los cambios del entorno local. Subversion sólo permitirá esta operación si no existen conflictos con el código ya existente en el repositorio. Es decir, no permitirá hacer Commit si otro miembro del equipo ha modificado el mismo elemento de forma paralela desde la última sincronización de código.
* La dinámica habitual de trabajo deberá ser la siguiente
* Antes de comenzar con la resolución de una tarea, se deberá asegurar la sincronización con el repositorio, bien con un Update o bien con un Checkout dependiendo de si se dispone previamente del código en el entorno local o no.
* Una vez resuelta la tarea, se deberá hacer otro Update para traer al entorno local los cambios que hayan podido ser realizados en paralelo al desarrollo actual. Subversion sabrá integrar los cambios del repositorio con los del entorno local en la mayoría de los casos, pero existirán situaciones que requieran de intervención humana para la integración. Estos casos, se deberán resolver de forma manual procurando mantener las modificaciones propias y las realizadas por los otros desarrolladores en paralelo.
* Finalmente se deberá hacer el Commit para hacer público al resto del equipo el código desarrollado. El alcance del Commit deberá limitarse al código relevante a la resolución de la tarea, y no mezclar desarrollos de distintas tareas en un mismo Commit.
* *Flujo de desarrollo*



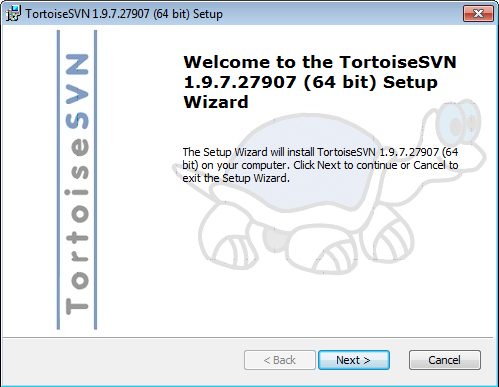
* + 1. **Alcance técnico de la herramienta**
* (lenguajes de programación que soporta, entornos de desarrollo que soporta)
  + 1. **Estructuras de proyectos recomendadas como buenas prácticas.**
  1. **GESTION DE REPOSITORIOS**
* *Qué tipo de información se debe almacenar en el repositorio*
* *Qué tipo de proyectos tecnológicos utilizan las dependencias de este tipo de repositorios*
* *Estructura de proyectos recomendada*
  1. **ANALISIS ESTATICO DE CÓDIGO FUENTE**
* *Umbrales recomendados en compuertas de calidad*
* *Tipos de tecnologías que se pueden utilizar en SonarQube*

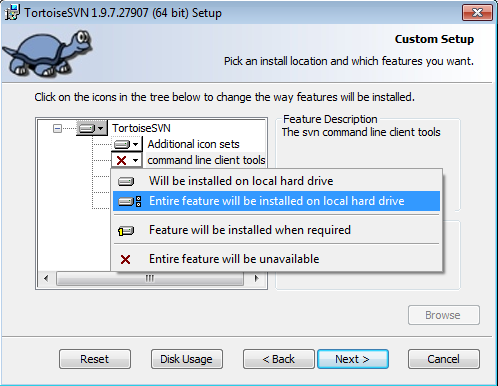
1. **ENTORNO DE DESARROLLADOR CON PLATAFORMA SOFTWARE HABILITADA**
   1. **INTEGRACIÓN CON SVN**

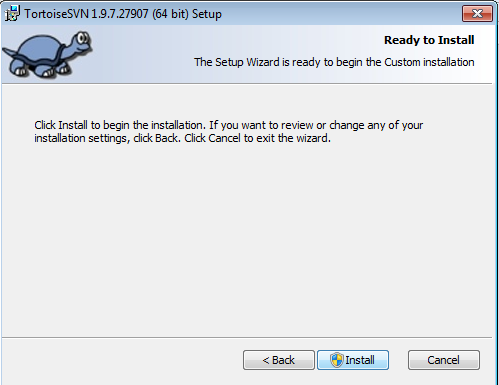
El servicio SVN 1.9.7 se encuentra publicado en la URL [**https://192.168.32.40/svn**](https://192.168.32.40/svn)

* + 1. **Instalar Tortoise SVN**

Descargar el instalador Tortoise SVN 1.9.7 desde <https://tortoisesvn.net/downloads.html> y realizar la instalación en la estación de desarrollador:







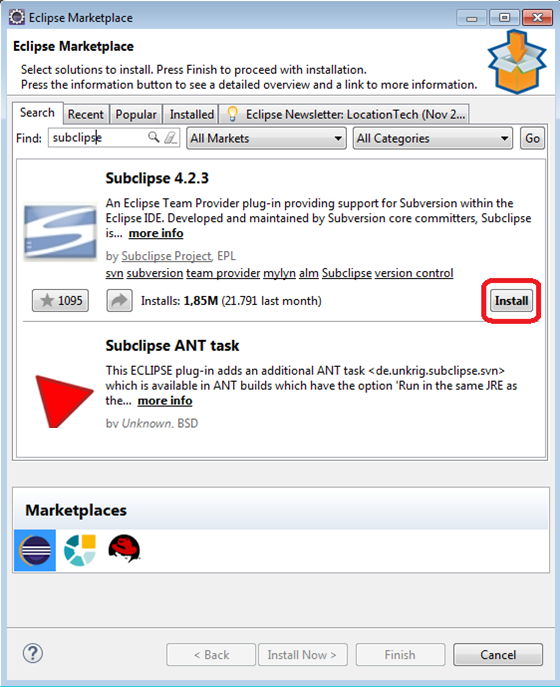
* + 1. **Instalar plugin Eclipse**

Como plugin se usra la versión Sublcipse 4.2.0 el cual requiere como mínimo

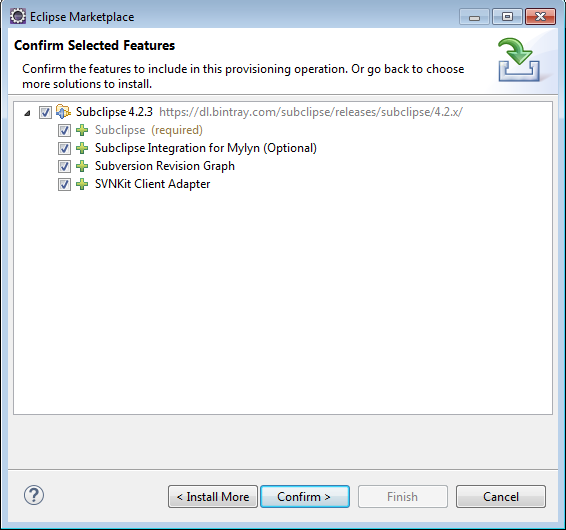
* Lista de IDE’s basados en Eclipse

|  |  |  |  |
| --- | --- | --- | --- |
| **Versión** | **Fecha de lanzamiento** | **Versión de plataforma** | **Proyecto** |
| Juno | 27 de junio de 2012 | 4.2 | [Juno projects](http://wiki.eclipse.org/Juno) |
| Kepler | 26 de junio de 2013 | 4.3 | [Kepler projects](http://wiki.eclipse.org/Kepler) |
| Luna | 25 de junio de 2014 | 4.4 | [Luna projects](http://wiki.eclipse.org/Luna) |
| Mars | 24 de junio de 2015 | 4.5 | [Mars projects](http://wiki.eclipse.org/Mars) |
| Neon | 22 de junio de 2016 | 4.6 | [Neon projects](http://wiki.eclipse.org/Neon) |
| Oxigen | 28 de junio de 2017 | 4.7 | [Oxigen projects](http://wiki.eclipse.org/Oxygen) |

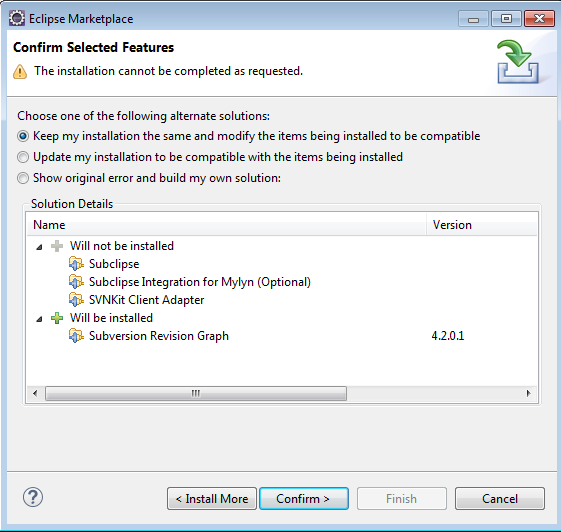
* En Eclipse ir a la opción “Help > Eclipse Marketplace …” y buscar “Sublcipse”



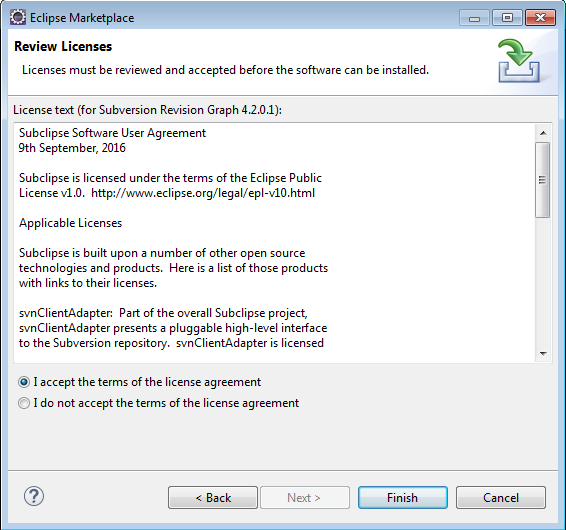
* Dar la opción “Install” y dar clic en la opción “Confirm >”

****

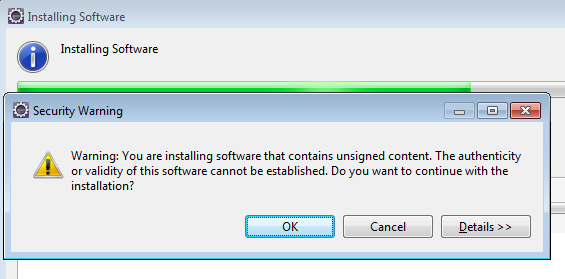
* En caso de ser Eclipse anterior a la compatibilidad del plugin el proceso de instalación preguntará cuál de las alternativas de solución usará, se recomienda usar el de la imagen y dar clic en “Confirm >”:



* Aceptar los términos y condiciones y dar clic en “Finish”



* Aceptar el continuar con la instalación y al terminar reiniciar el IDE:

****

* + 1. **Administrar repositorios en servidor**
* Acceder por SSH al servidor que proporciona el servicio SVN (192.168.32.40)
* Ingresar a la línea de comandos del contenedor docker que ejecuta el SVN:

$ docker exec --it repfsw-svn1 /bin/sh

* Para CREAR REPOSITORIO SVN de nombre new-repo:

$ cd /home/svn/data

$ svnadmin create **new-repo**

$ chown -R apache.apache /home/svn/data/**new-repo**

* 1. **INTEGRACIÓN CON NEXUS**

El servicio NEXUS 3.6.1 se encuentra publicado en la URL [**https://192.168.32.40/**](https://192.168.32.40/svn)**nexus3**

* + 1. **Instalación de maven**
* Descomprimir el recurso maven apache-maven-3.5.2.7z en el directorio “D:\sw\app” de manera que se pueda invocar al ejecutable “D:\sw\app\apache-maven-3.5.2\bin\mvn.cmd”
* Agregar las variables de sistema:

M2\_HOME = D:\sw\app\apache-maven-3.5.2

* Modificar el PATH del sistema operativo para agregar la variable creada:

PATH = …;%M2\_HOME%\bin;

* Verificar el acceso al binario de Maven ejecutando:

:\ mvn –version

* Editar el archivo %M2\_HOME%\conf\ settings.xml para que contenga lo siguiente:

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

<settings xmlns="http://maven.apache.org/SETTINGS/1.0.0"

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

xsi:schemaLocation="http://maven.apache.org/SETTINGS/1.0.0 http://maven.apache.org/xsd/settings-1.0.0.xsd">

<localRepository>d:/svr/maven352/repositorio</localRepository>

<pluginGroups>

</pluginGroups>

<proxies>

<proxy>

<id>proxyOsce</id>

<active>true</active>

<protocol>http</protocol>

<host>172.16.2.3</host>

<port>3128</port>

<nonProxyHosts>localhost|172.\*|192.\*|&lt;local&gt;|localhost.localdomain|127.0.0.1|::1|172.16.51.26|[::1]</nonProxyHosts>

</proxy>

</proxies>

<servers>

<server>

<id>libs3-repository</id>

<username>usuario</username>

<password>123456</password>

</server>

<server>

<id>osce-releases</id>

<username>usuario</username>

<password>123456</password>

</server>

<server>

<id>osce-usuario</id>

<username>usuario</username>

<password>123456</password>

</server>

</servers>

<mirrors>

<mirror>

<id>nexus</id>

<mirrorOf>\*</mirrorOf>

<url>http://192.168.32.41/nexus3/content/groups/public</url>

</mirror>

</mirrors>

<profiles>

<profile>

<id>osce</id>

<repositories>

<repository>

<id>osce-releases</id>

<url>http://192.168.32.41/nexus3/content/repositories/osce-releases</url>

<releases>

<enabled>true</enabled>

</releases>

<snapshots>

<enabled>false</enabled>

</snapshots>

</repository>

<repository>

<id>osce-snapshots</id>

<url>http://192.168.32.41/nexus3/content/repositories/osce-snapshots</url>

<releases>

<enabled>false</enabled>

</releases>

<snapshots>

<enabled>true</enabled>

</snapshots>

</repository>

</repositories>

<pluginRepositories>

<pluginRepository>

<id>osce-releases</id>

<url>http://192.168.32.41/nexus3/content/repositories/osce-releases</url>

<releases>

<enabled>true</enabled>

</releases>

<snapshots>

<enabled>false</enabled>

</snapshots>

</pluginRepository>

<pluginRepository>

<id>osce-snapshots</id>

<url>http://192.168.32.41/nexus3/content/repositories/osce-snapshots</url>

<releases>

<enabled>false</enabled>

</releases>

<snapshots>

<enabled>true</enabled>

</snapshots>

</pluginRepository>

</pluginRepositories>

</profile>

<profile>

<id>libs3</id>

<repositories>

<repository>

<id>libs3-repository</id>

<url>http://192.168.32.41/nexus3/content/repositories/libs3-repository</url>

<releases>

<enabled>true</enabled>

</releases>

<snapshots>

<enabled>true</enabled>

</snapshots>

</repository>

</repositories>

<pluginRepositories>

<pluginRepository>

<id>libs3-repository</id>

<url>http://192.168.32.41/nexus3/content/repositories/libs3-repository</url>

<releases>

<enabled>true</enabled>

</releases>

<snapshots>

<enabled>true</enabled>

</snapshots>

</pluginRepository>

</pluginRepositories>

</profile>

<profile>

<activation>

<activeByDefault>true</activeByDefault>

</activation>

<repositories />

</profile>

</profiles>

<activeProfiles>

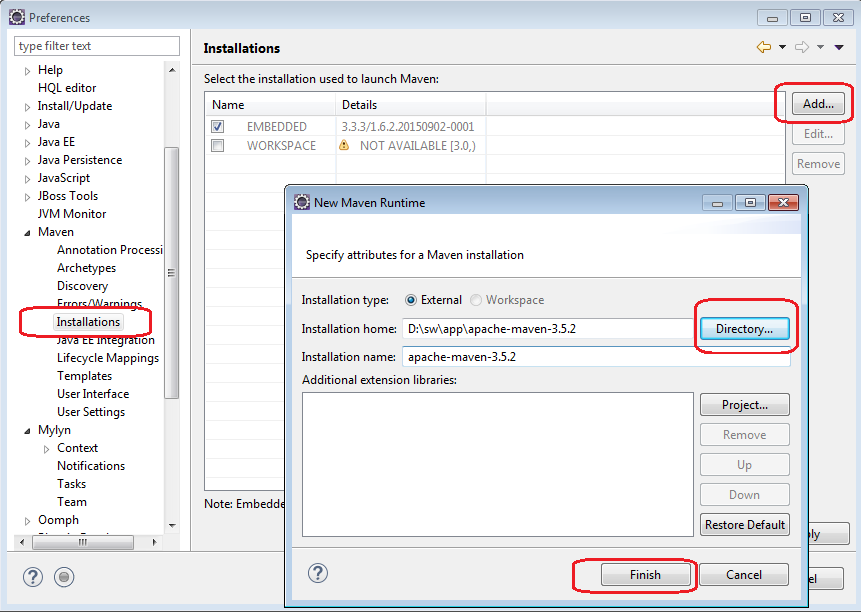
<activeProfile>osce</activeProfile>

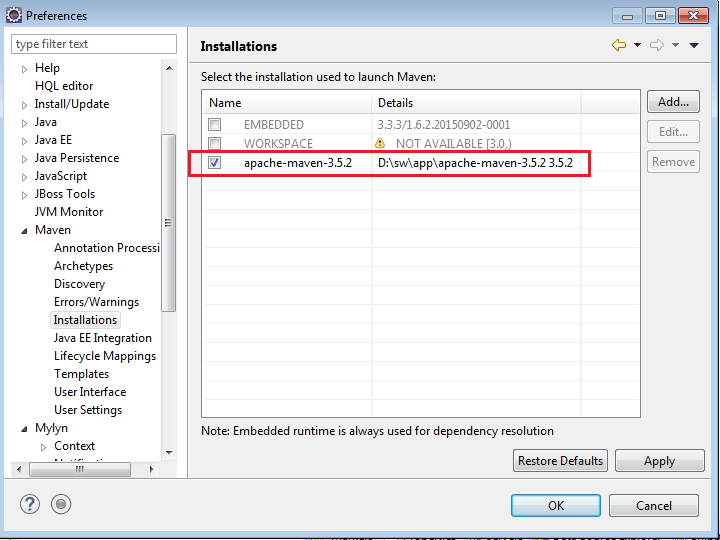
<activeProfile>libs3</activeProfile>

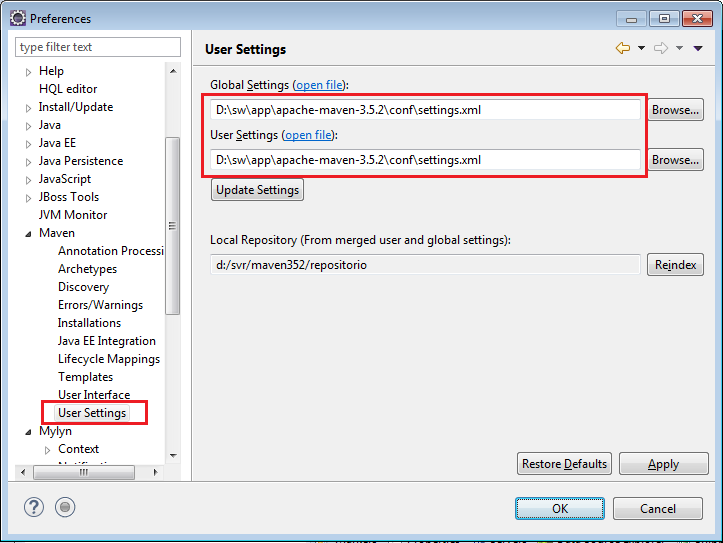
</activeProfiles>

</settings>

* + 1. **Configurar maven Eclipse**
* Agregar instalación externa
* Configurar el acceso a repositorios







* A
  1. **INTEGRACIÓN CON SONARQUBE**

El servicio SonarQube 6.7 se encuentra publicado en la URL [**https://192.168.32.41/**](https://192.168.32.41/)**sonarqube67**

* + 1. **Instalar SonarQube Scanner**

SonarQube Scanner permite analizar proyectos con SonarQube directamente en el directorio donde se encuentre el código fuente.

* SonarQube Scanner requiere de Java 1.8 o superior.
* Según el sistema operativo descargar del sonar-scanner-cli de la siguiente URL <https://docs.sonarqube.org/display/SCAN/Analyzing+with+SonarQube+Scanner>
* Descomprimir el contenido del archivo descargado y dejar en el nombre la versión a 3 niveles, ejemplo: D:\sw\app\sonar-scanner-3.0.3
* Editar las variables de entorno del sistema operativo para agregar la variable:

SQSCANNER\_HOME = D:\sw\app\sonar-scanner-3.0.3

* Modificar el PATH del sistema operativo para agregar la variable creada:

PATH = …;%SQSCANNER\_HOME%\bin;

* Verificar el acceso al binario del SonarQube Scanner ejecutando:

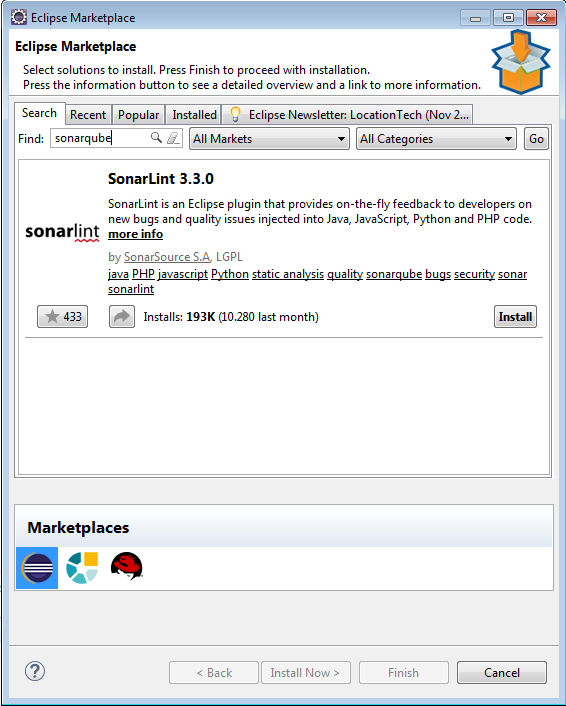
:\ sonar-scanner --version

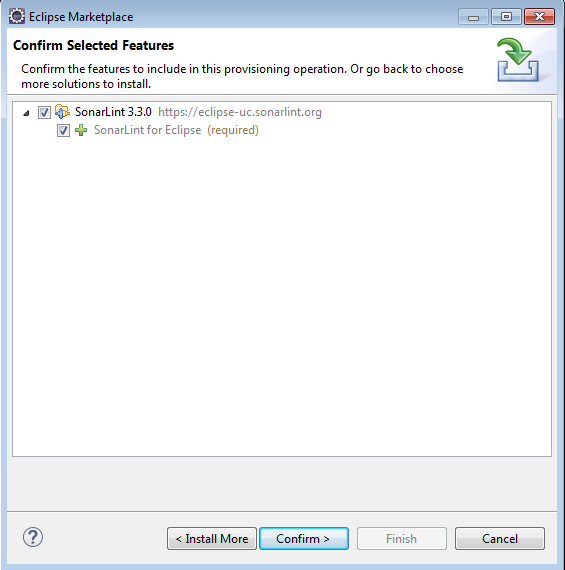
* Actualizar la configuración global del servidor SonarQube que se utilizará, para ello editar el archivo %SQSCANNER\_HOME%\conf\sonar-scanner.properties al valor que le corresponderá

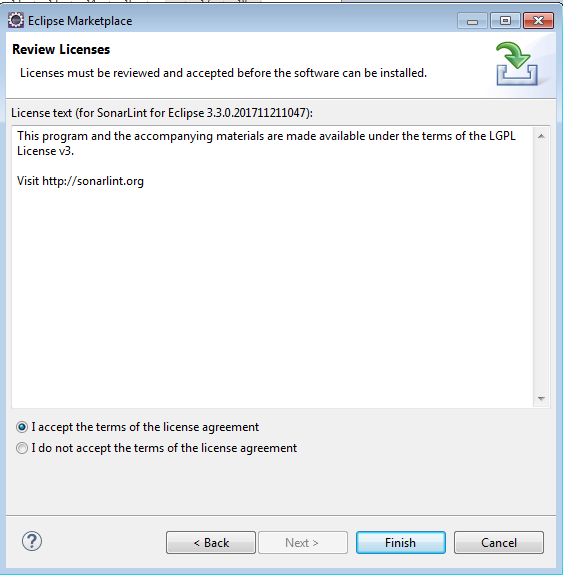
#----- Default SonarQube server

sonar.host.url=https://192.168.32.41/sonarqube67

* + 1. **Configurar Eclipse con SonarQube**
* Agregar instalación externa







* Configurar el acceso a repositorios
  + 1. **Crear proyectos SonarQube**
* Agregar instalación externa
* Configurar el acceso a repositorios
  + 1. **Personalizar configuraciones en SonarQube**
* Agregar instalación externa
* Configurar el acceso a repositorios
  + 1. **Analizar proyectos con SonarQube Scanner**
* Crear archivo de configuración de nombre **sonar-project.properties** en el directorio raíz donde se encuentre el proyecto con el siguiente contenido:

*# must be unique in a given SonarQube instance*

*sonar.projectKey=****my:project***

*# this is the name and version displayed in the SonarQube UI. Was mandatory prior to SonarQube 6.1.*

*sonar.projectName=****My project***

*sonar.projectVersion=****1.0***

*# Path is relative to the sonar-project.properties file. Replace "\" by "/" on Windows.*

*# This property is optional if sonar.modules is set.*

*sonar.sources=****.***

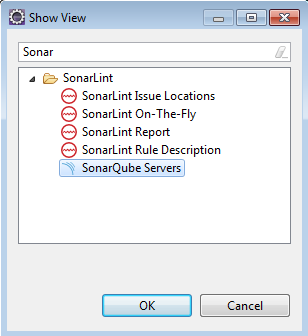
*# Encoding of the source code. Default is default system encoding*

*#sonar.sourceEncoding=****UTF-8***

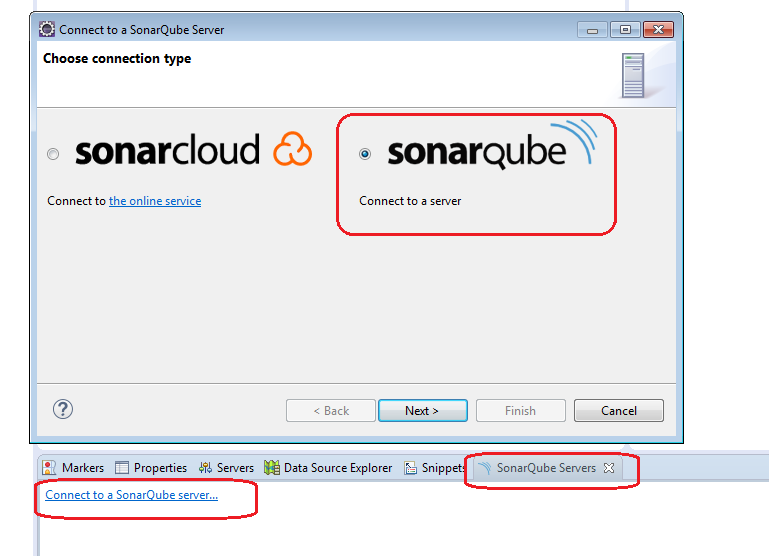
* En el directorio que contiene el proyecto y el archivo sonar-project.properties ejecutar:

:\ sonar-scanner

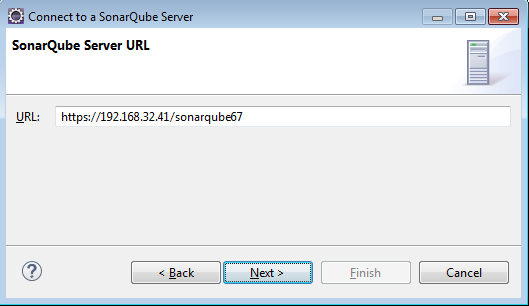
* La ejecución anterior realizara la carga del análisis en el servidor SonarQube configurado durante el proceso de instalación
  + 1. **Analizar proyectos con Eclipse**
* Para crear conexión a servidor SonarQube buscar en “Windows > Show view > Others” la palabra Sonar y seleccionar SonarQube Servers:



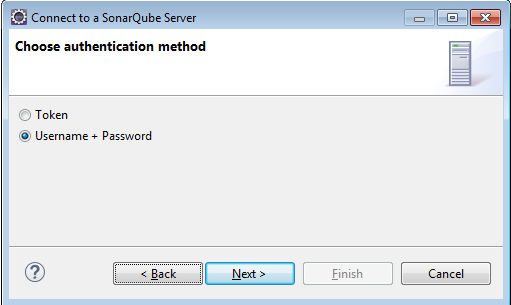
* En la vista apertura dar clic en la opción “Connect to a SonarQube Server”



* Ingresar la URL del servidor que se usará en el proyecto



* Seleccione el método de autenticación al servidor SonarQube



* A
  1. **INTEGRACIÓN CON SELENIUM**

1. **ASPECTOS TÉCNICOS EN MIGRACIÓN DE REPOSITORIOS**
   1. **MIGRACION REPOSITORIO SVN 1.7.4 HACIA 1.9.7**
      1. **Migración por repositorio**

* En el servidor que contiene los repositorios a migrar crear el directorio donde se almacenara los dump’s de los repositorios:

$ mkdir -p /home/svrsvn/dumps

$ chown apache:apache -R /home/svrsvn/dumps

$ cd /home/srvsvn/dumps

* Generar dump de repositorio con el siguiente comando:

$ svnadmin dump /home/svrsvn/repo/repo-migrar/ > \

/home/srvsvn/dumps/repo-migrar\_dump

* Copiar archivo dump en nuevo servidor:

$ scp /home/srvsvn/dumps/repo-migrar\_dump \

operadorsvc@192.168.32.40:/fssvc/svn/1.9.7/svn-data/temp/repo-migrar\_dump

* En el nuevo servidor SVN dentro del contenedor crear el repositorio destino:

$ docker exec --it repfsw-svn1 /bin/sh

$ cd /home/svn/data

$ svnadmin create repo-migrar

* Dentro del contenedor efectuar la importación del dump copiado:

$ svnadmin load /home/svn/data/ < \

/home/svn/data/temp/repo-migrar\_dump

* + 1. **Migración por copia física**
* Para realizar copia masiva de los repositorios existentes realizar lo siguiente:

$ nohup scp -pr /home/svrsvn/repo/ \

operadorsvc@192.168.32.40:/fssvc/svn/1.9.7/svn-data/

* En el nuevo servidor SVN dentro del contenedor crear el repositorio destino:

$ docker exec --it repfsw-svn1 /bin/sh

$ chown -R apache.apache /home/svn/data

* + 1. **Copia de trabajo en entorno de desarrollador**
* En las estaciones clientes por cada repositorio se debe realizar 02 tareas:
  + svn cleanup
  + svn upgrade

**Nota:** De presentarse inconvenientes en la copia de trabajo es preferible crear una nueva copia ejecutando **svn checkout**

* 1. **MIGRACION REPOSITORIO NEXUS 2.1.2 HACIA *3.6.2***
     1. **Migración asistida por consola de administración**

**No migrar EMAIL, LDAP-Configuration, Realms**

* + 1. **Personalización post-migración**

1. **RELACIÓN DE LA PLATAFORMA SOFTWARE HABILITADA CON LA NTP ISO/IEC 12207**

|  |  |
| --- | --- |
| **NTP ISO/IEC 12207** | **ENTREGABLES DE CONSUTORIA** |
| **5.4 PROCESO DE OPERACIÓN**  **5.4.2 Pruebas de operación**   * Para cada release del producto software, el operador deberá llevar a cabo pruebas de operación y tras satisfacerse los criterios especificados, liberar el software para uso en operación. * El operador deberá asegurar que el código software y las bases de datos se inicializan, ejecutan y terminan tal como se describe en el plan | **INFORME DE INSTALACIÓN DE LA PLATAFORMA SOFTWARE HABILITADA**  **3. Instrucciones de operaciones de los productos software’s habilitados**   * Instrucciones de operaciones de los productos software’s habilitados * AAA bbbb aaaaaaaaaaaaaa aaaaaaaaa aaaaaaa |
| **5.4 PROCESO DE OPERACIÓN**  **5.4.3 Operación del sistema**   * El sistema deberá ser operado en el entorno previsto de acuerdo con la documentación de usuario | * Instrucciones de operaciones de los productos software’s habilitados |