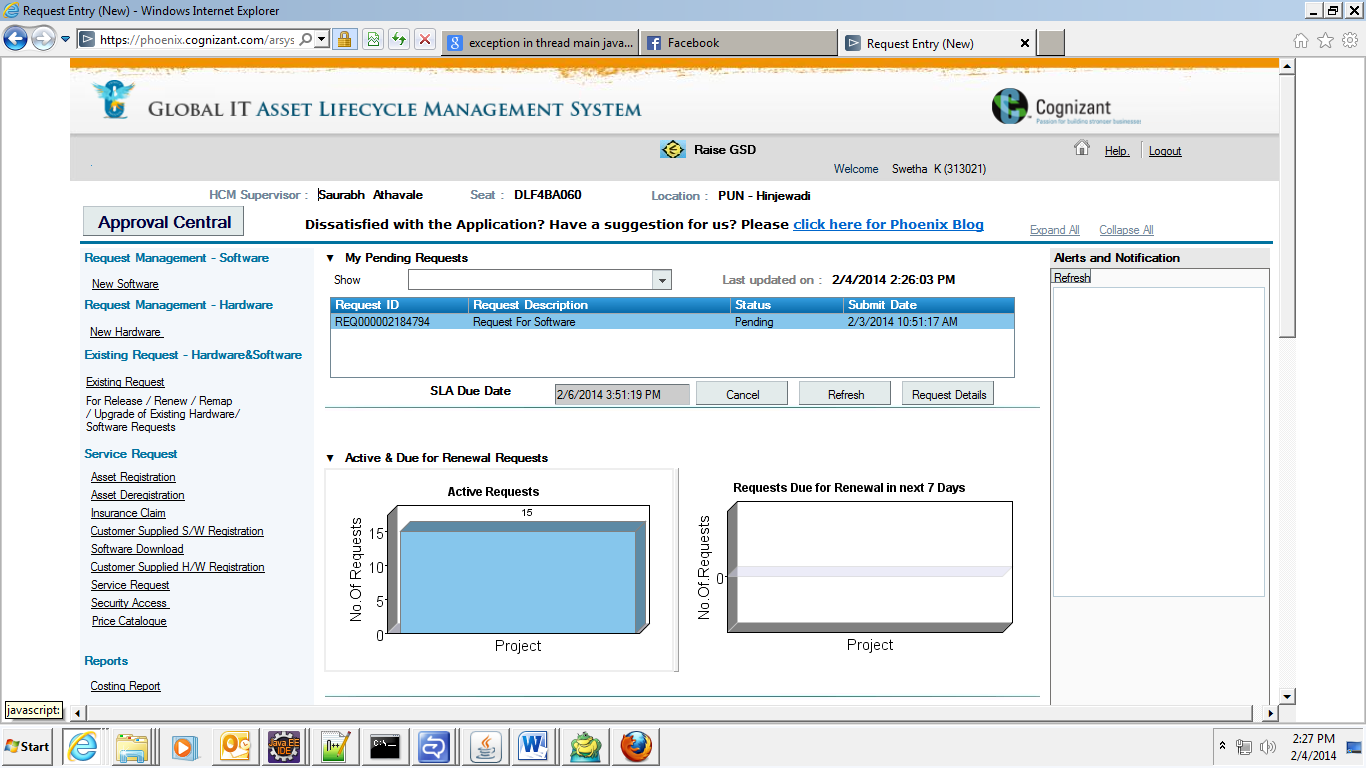
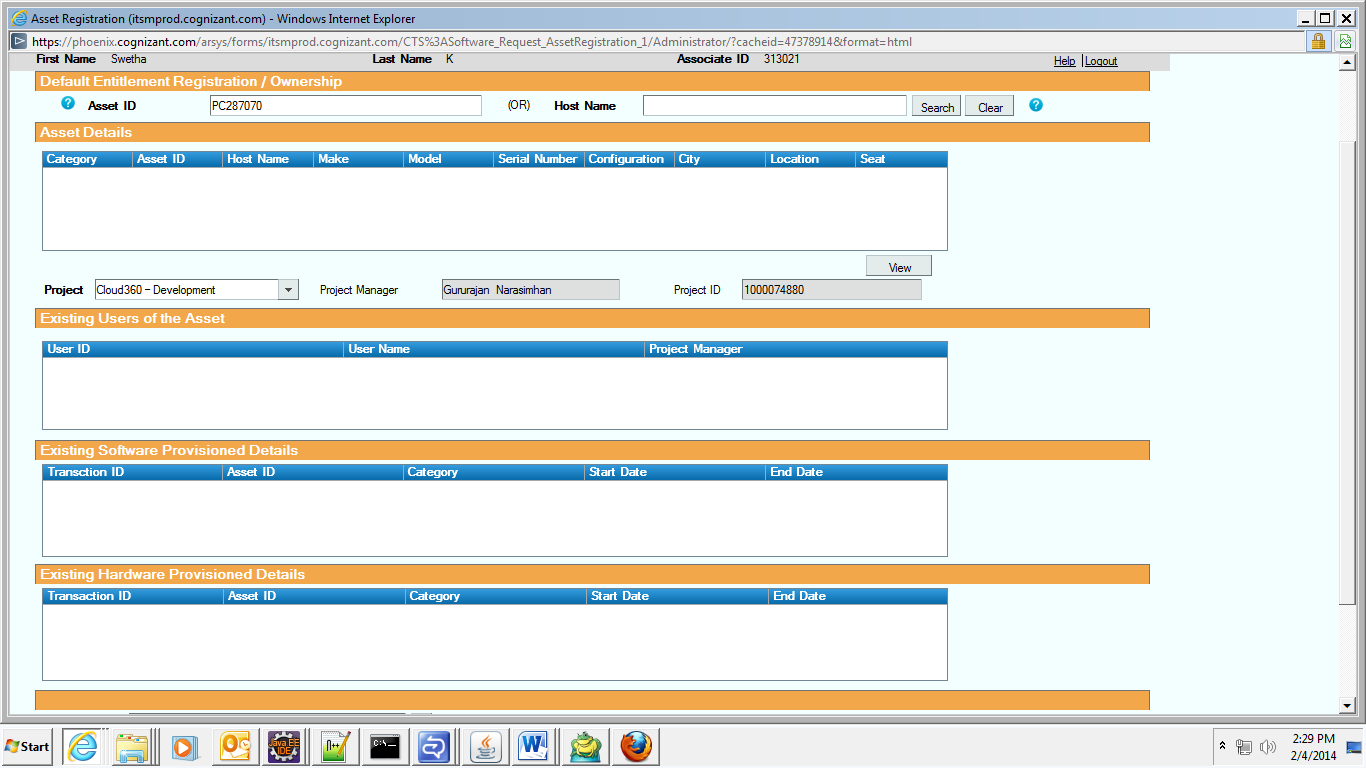
1. Asset Registration through phoenix



 Click on I agree and I am done. Asset will register. Once asset is registered one can raise the s/w request as follows:

Provide Project and asset Details and say proceed

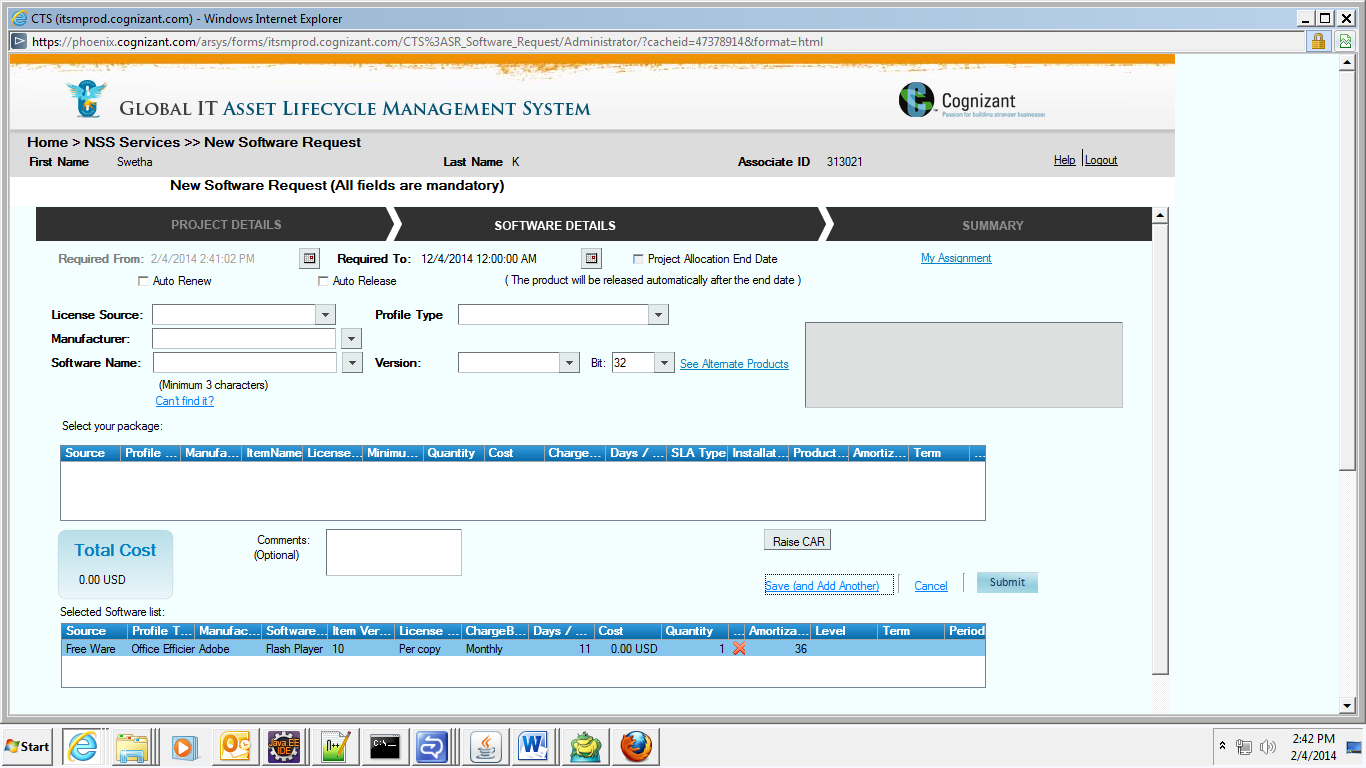
1. Raise the phoenix request for below s/w with following details.

Required To date: 1 year from the current date

Each time do Save (and Add Another)

|  |  |  |  |
| --- | --- | --- | --- |
| **Sl.No** | **License Source** | **Software Name** | **Version** |
| 1 | Freeware | Cisco VPN Client | 5.03 |
| 2 | Freeware | Toad for MySQL | 4.5 |
| 3 | Open Source | MySQL Database Server Community Edition | 5.0.81 |
| 4 | Open Source | WinSCP SFTP & SCP Client | 4.1.5 |
| 5 | Freeware | Flash Player | 10 |
| 6 | Open Source | Apache Tomcat | 6.0.17 |
| 7 | Free ware | JDK | 1.7.\* |
| 8 | Free ware | JRE | 1.6.13 |
| 9 | Open Source | SDE | 6 |
| 10 | Open Source | mysqlserver 5.5 | 5.5 |
| 11 | Open Source/freeware | maven | latest |
| 12 | Open Source/freeware | svn | latest |
| 13 | Open Source/freeware | putty | latest |
| 14 | Open Source and profile type : browser | Mozilla FireFox | latest |

Ex Snap-shot



Repeat same for all s/w.

1. Admin Rights

Phoenix -> Service Approval -> Security Access -> Raise Request: Provide all details and say submit.

Note: If you are transferred from other location first make sure your Domestic Transfer is completed before raising this request.

1. VPN client set Up

Make Sure your IP is moved to Cloud LAN : The IP should look like : 10.226.147.\*

Open VPN client : Connection Entries : Import :

Import the following file :

And then connect as San Ramon providing your user Id and Password.

1. Environment Variables to be set :

User Variables :

JAVA\_HOME : C:\Program Files\Java\jdk1.7.0\_45

M2 : %M2\_HOME%\bin

MVN\_HOME : D:\apache-maven-3.1.0

MYSQL\_HOME : C:\Program Files\MySQL\MySQL Server 5.5\bin

System Variables :

CATALINA\_HOME : D:\apache-tomcat-7.0.27

CATALINA\_OPTS : -XX:MaxPermSize=200m

CLOUDONE\_HOME : D:\cloud\_home unzip this and set the path

M2\_HOME=C:\apache-maven-3.0.3

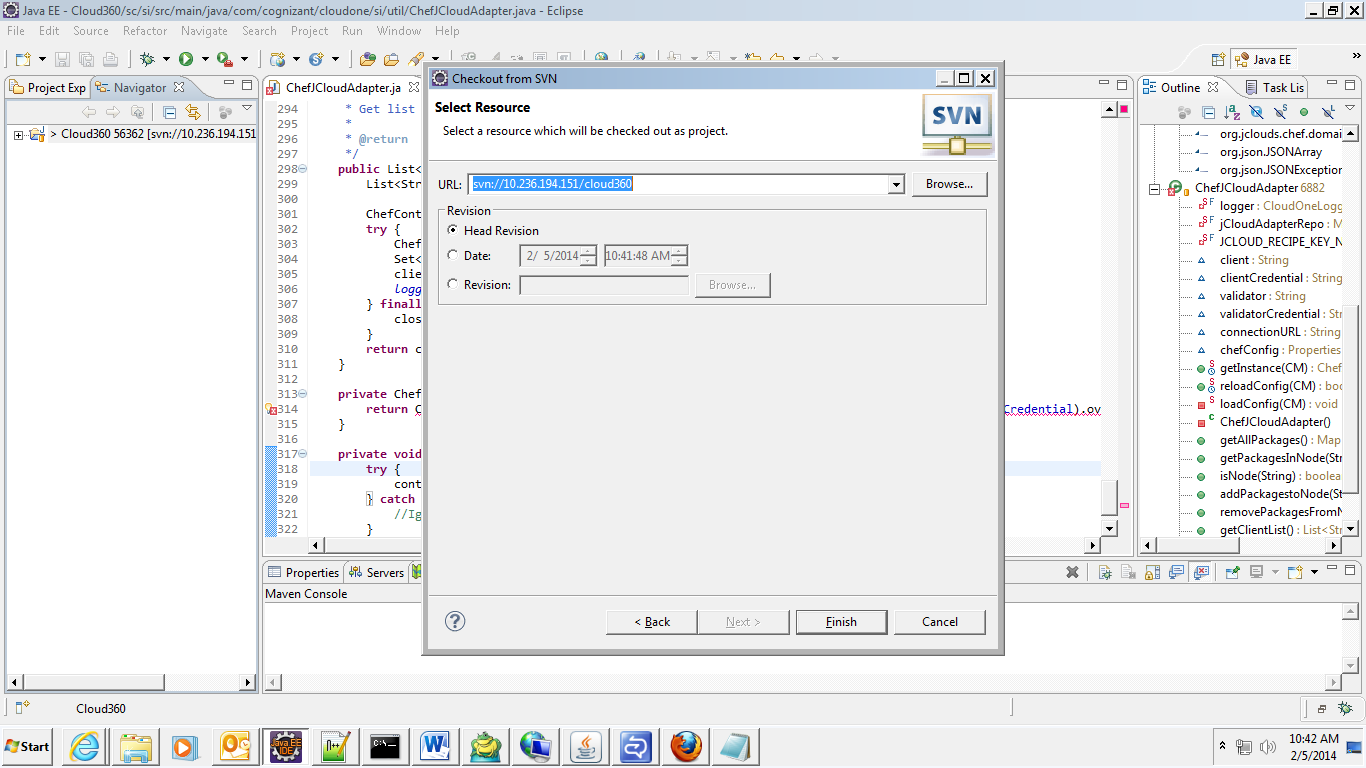
Path : %SystemRoot%\system32;%SystemRoot%;%SystemRoot%\System32\Wbem;%SYSTEMROOT%\System32\WindowsPowerShell\v1.0\;C:\Program Files\TortoiseSVN\bin;%JAVA\_HOME%\bin;%M2\_HOME%\bin;C:\Program Files\MySQL\MySQL Server 5.5\bin;D:\apache-maven-3.1.0\bin;C:\Program Files\Java\jdk1.7.0\_45\bin;D:\Cloud\_Code\old svn\trunk\docs\liquibase\bin;D:\apache-tomcat-7.0.27\bin

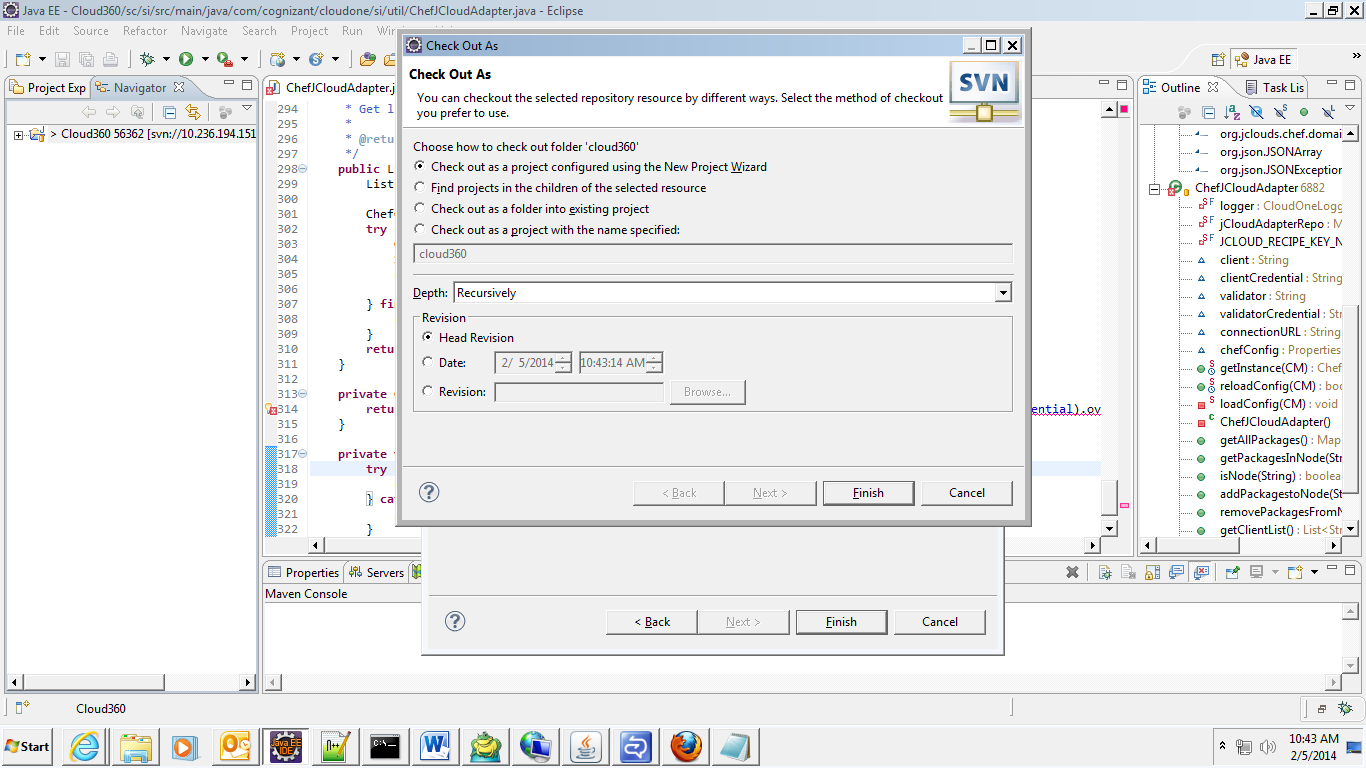
1. Code Set Up In Eclipse

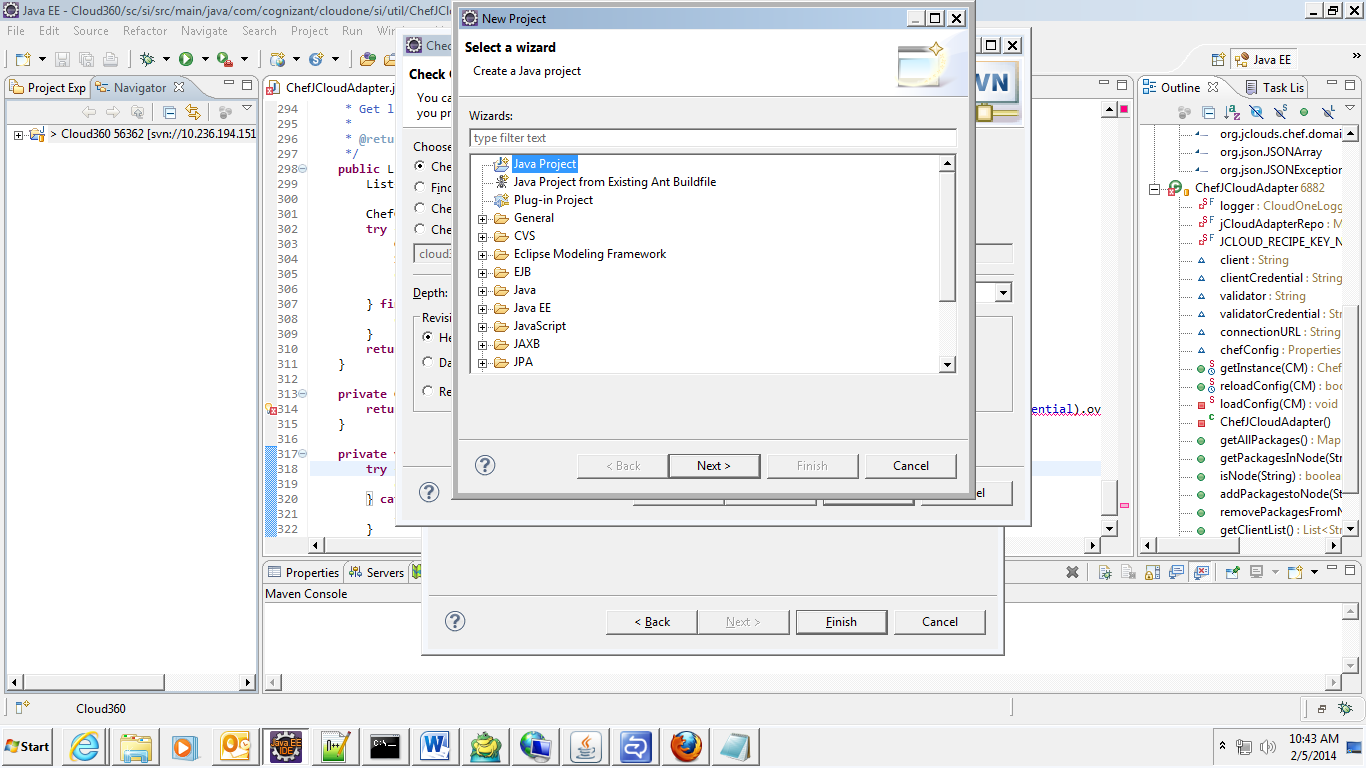
Create a WorkSpace In eclipse and follow below steps:

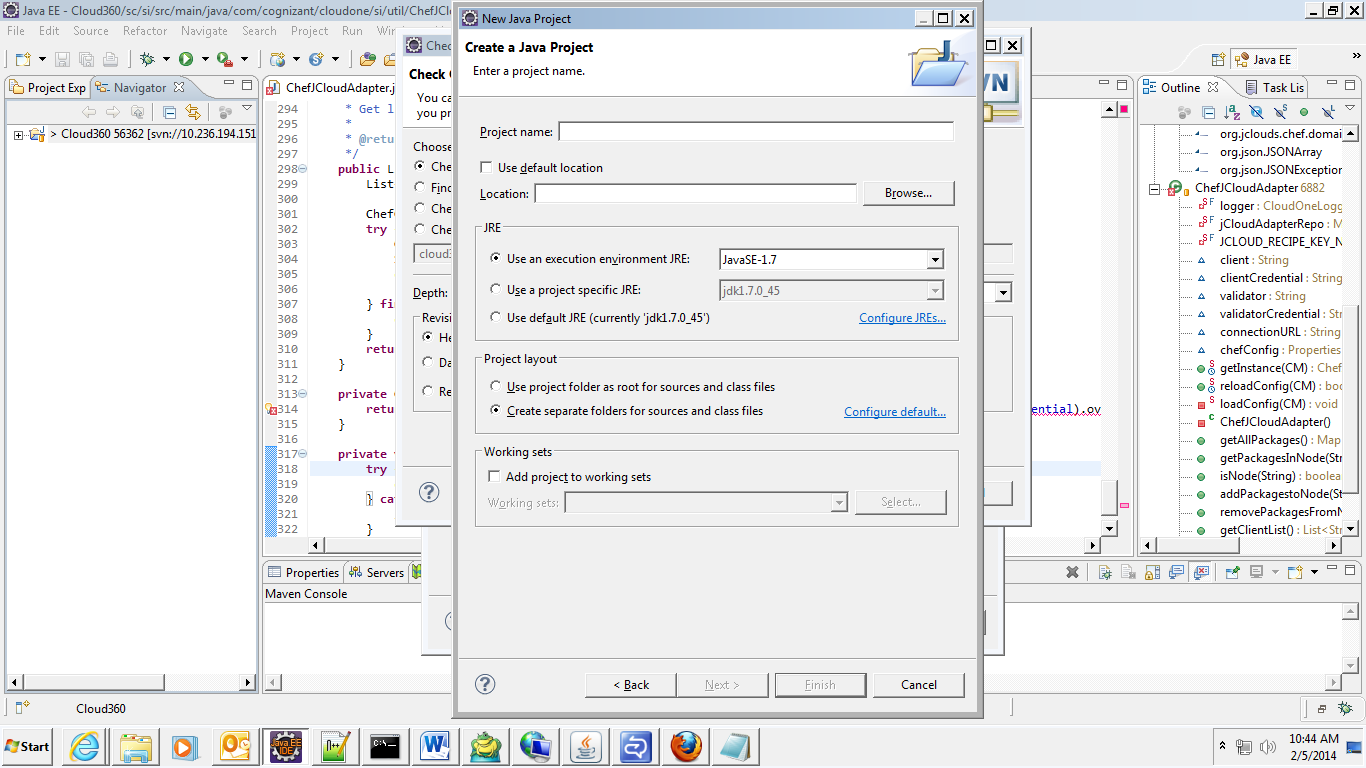
File->New->Other->SVN->Project From SVN

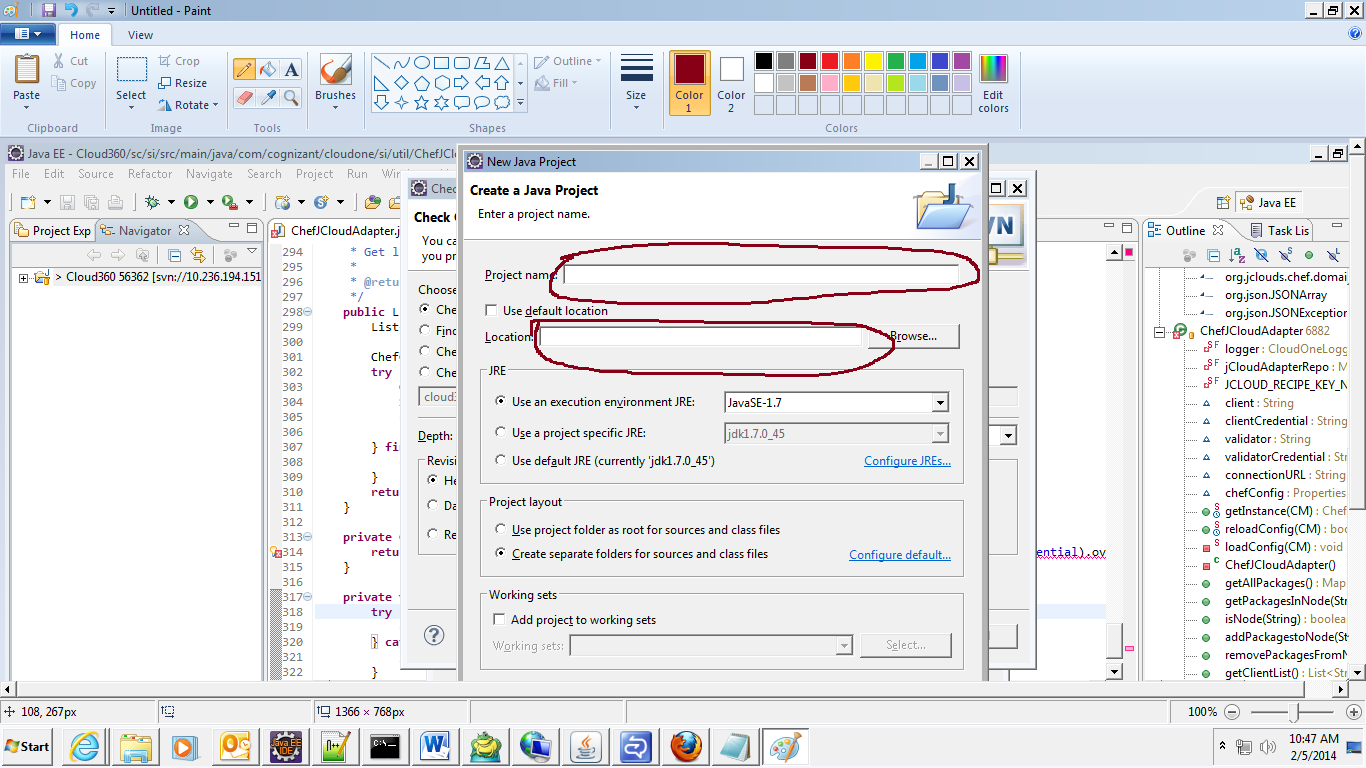
Provide the SVN URL of latest code : svn://10.236.194.151/cloud360/trunk and say finish







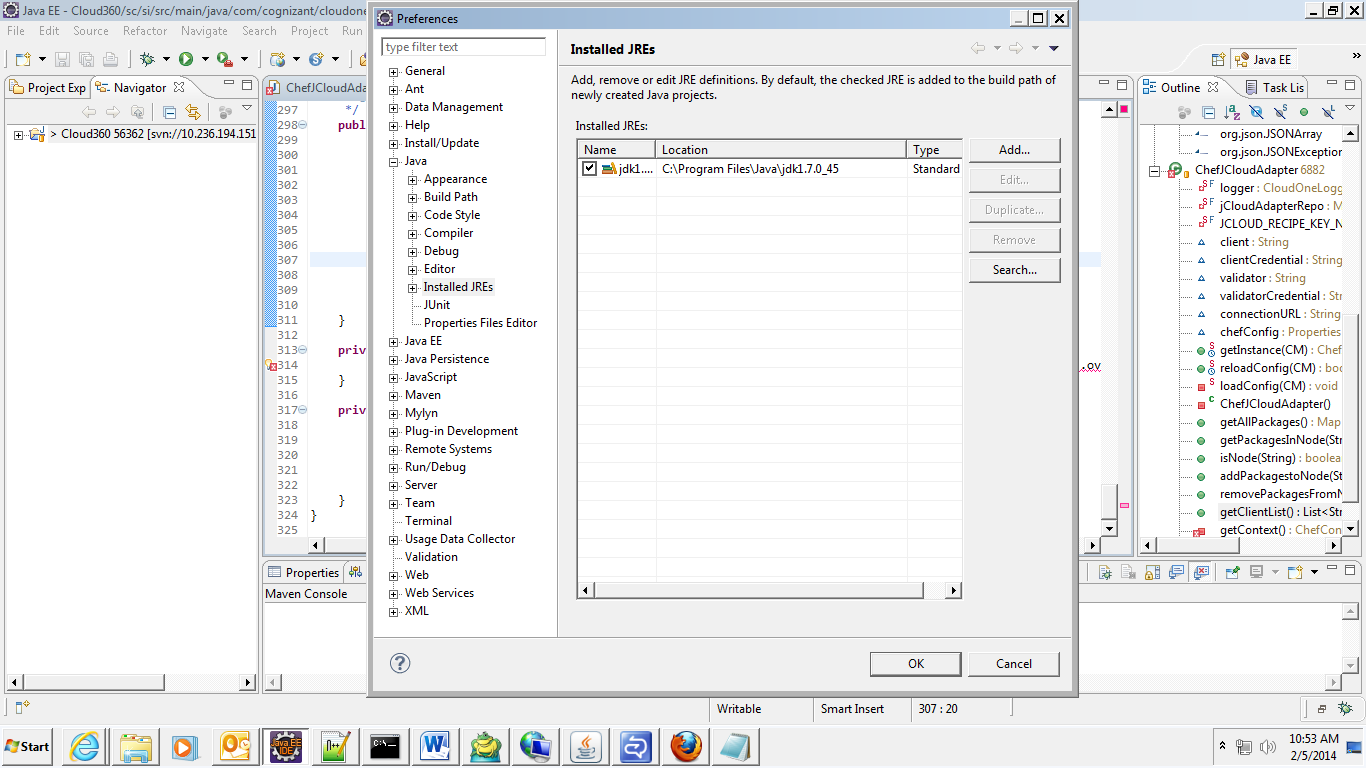




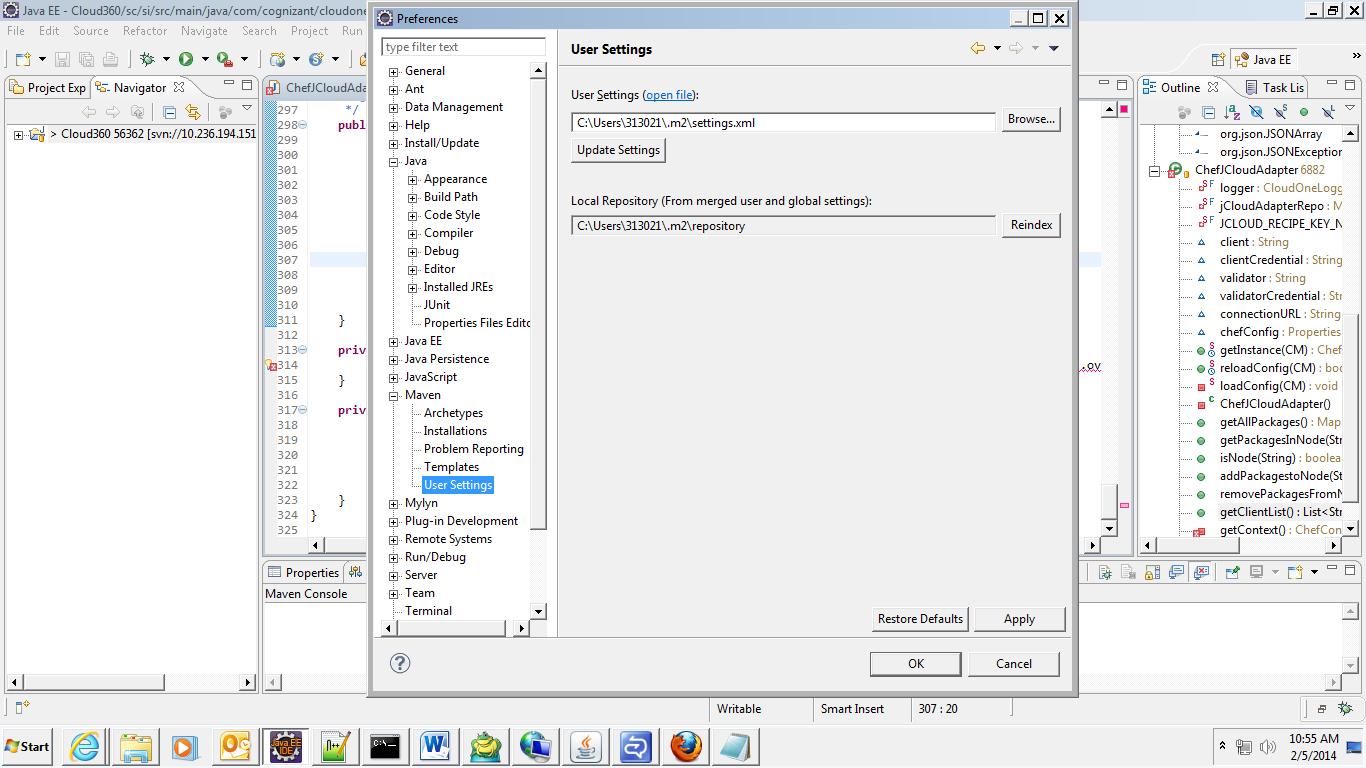
Project Name : Name you wish to see as a eclipse project : Cloud360

Location : Location to Check out the Code. Ex : D:\Cloud\_Code\New SVN

Check The Installed JRE is same as the JAVA\_HOME variable.



Check If your maven is pointing to the right location

The required settings.xml is : 

Copy The maven plugins to eclipse plugins : D:\software\eclipse indigo sr2\eclipse-jee-indigo-SR2-win32\eclipse\plugins :

1. DB set Up

change the mysql pwd in following 2 files :

D:\cloud\_home\config\dal\override-db-config.properties

(NOTE: **D:\cloud\_home\config\dal** is the directory structure created by user.) D:\Cloud\_Code\New SVN\trunk\docs\liquibase\mc-changelog\liquibase.properties

Open **MySQL command line** and run the following DB scripts:

drop database if exists cloudonedb;

create database cloudonedb;

grant ALL PRIVILEGES on cloudonedb.\* to 'cloudone' identified by 'password';

grant select, insert, update, delete, execute on cloudonedb.\* to 'cloudone'@'localhost' identified by 'password';

commit;

Copy the **bin** and **lib** directory

from **Cloud\_360\common\liquibase**  to **Cloud\_360\mc**

Open CMD PROMPT

Goto directory, D:\Cloud Workspace\Cloud\_one360\mc\mc-liquibase-changelog

Set the path

Set path=” D:\Cloud Workspace\Cloud\_one360\common\liquibase\bin”%PATH%

Run the bat file

lb\_install.bat DEV

1. To build the Project.

Open CMD PROMPT and goto, D:\Cloud Workspace\Cloud\_one360

Command to convert project to eclipse project : mvn eclipse:eclipse -Dwtpversion=2.0

[Note: **mvn clean install -Dmaven.test.skip=true**  is used when we need to clean and install. For initial setup, just run the first maven command]

**NOTE:** Once build is failed because of Agent = failed, Then build the tools separately using the following way,

Goto D:\Cloud Workspace\Cloud\_one360\tools\tools in CMD PROMPT

and build using mvn clean install -Dmaven.test.skip=true

And cloudone-tools-0.0.1-SNAPSHOT.jar in Sample Project library

1. Deploying and running .

Copy the following war: **cloudone-ui-0.0.1-SNAPSHOT.war**

From D:\Cloud Workspace\Cloud\_one360\mc\ui\target

to D:\apache-tomcat-7.0.50\webapps

Copy the following jar **org.springframework.instrument.tomcat-3.0.4.RELEASE**

**To** D:\apache-tomcat-7.0.50\lib

Replace **catalina.bat** and **tom-debug.bat** to D:\apache-tomcat-7.0.50\bin

Start the Tomcat using **startup.bat**

from : D:\apache-tomcat-7.0.50\bin

Check if all the domains are fine using these url :

<http://localhost:8080/cloudone-ic-services-0.0.1-SNAPSHOT/services>

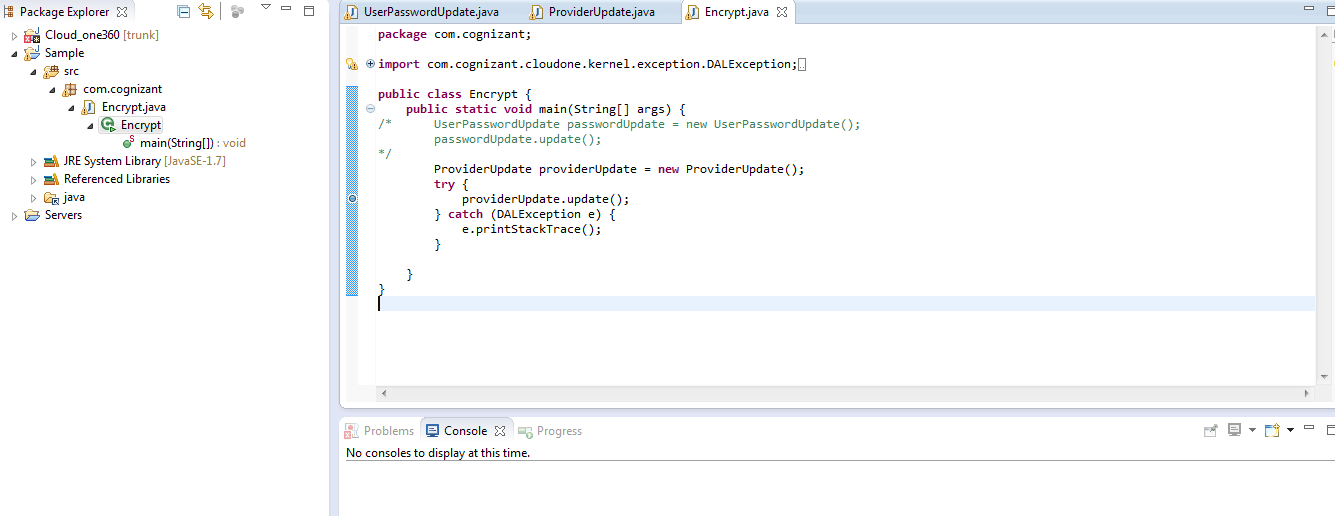
<http://localhost:8080/cloudone-agent-0.0.1-SNAPSHOT/services/>

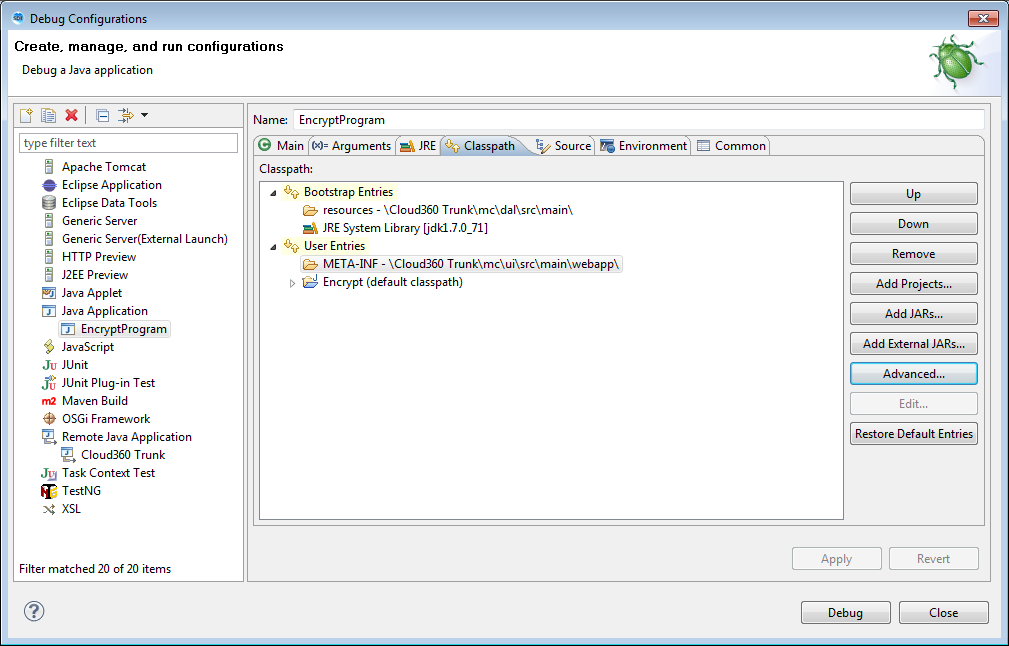
<http://localhost:8080/cloudone-ui-0.0.1-SNAPSHOT/login/login.cl>

10) Encryption:

i) create Java Project named 'Sample'

create Java class named 'Encrpt' under com.cognizant folder structure.





ii) Set Debug configuration as per the following details,

Main:

Project - Cloud\_one360

Main Class- com.cognizant.cloudone.tools.UserPasswordUpdate

Arguments:

VM Arguments-

-javaagent:C:\Users\411967\.m2\repository\org\springframework\spring-instrument\3.2.6.RELEASE\spring-instrument-3.2.6.RELEASE.jar

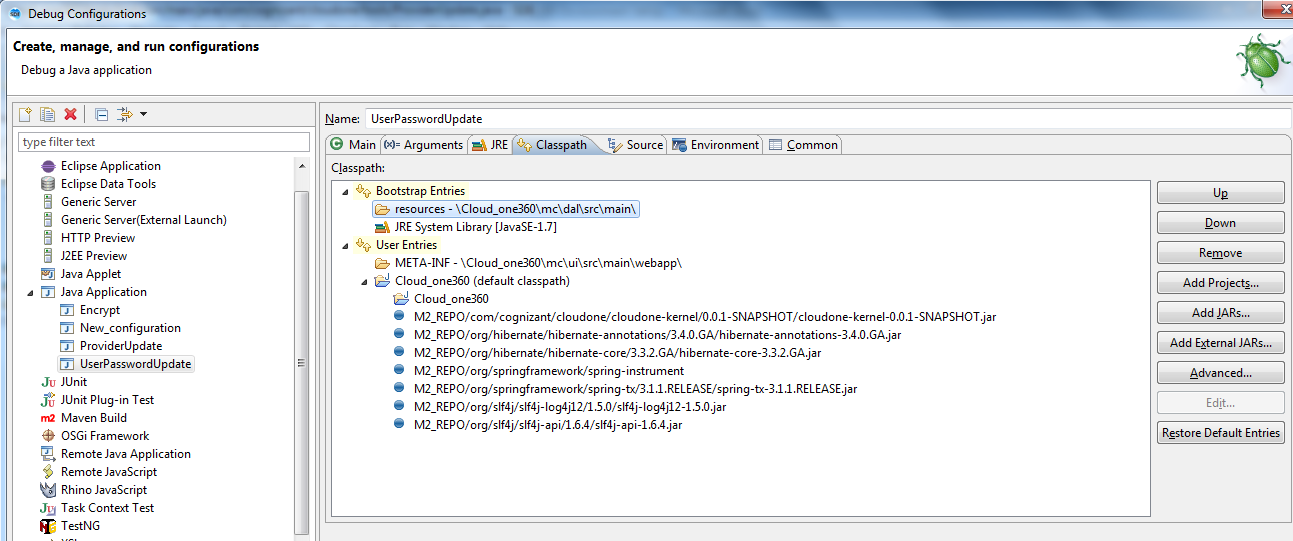
-Djava.awt.headless=true -Dorg.apache.el.parser.SKIP\_IDENTIFIER\_CHECK=true -Dfile.encoding=UTF-8 -server -Xms256m -Xmx512m -XX:NewSize=128m -XX:MaxNewSize=128m -XX:PermSize=128m -XX:MaxPermSize=128m -Xloggc:D:\Cloudone\_home\logs\gc.log

Class path:

Set the classpath as per the following Image,

Click **Add External jars** and add the jar from

D:\Cloud Workspace\Cloud\_one360\tools\tools\target\cloudone-tools-0.0.1-SNAPSHOT.jar

D:\apache-tomcat-7.0.50\webapps\cloudone-ui-0.0.1-SNAPSHOT\WEB-INF\lib\ 

11) Once the Encryption is finished, we can start the tomcat and hit the following URL to launch our application.

[**http://localhost:8080/cloudone-ui-0.0.1-SNAPSHOT/login/login.cl**](http://localhost:8080/cloudone-ui-0.0.1-SNAPSHOT/login/login.cl)

12) **DEBUG Configuration:**

1. Run **D:\apache-tomcat-7.0.50\bin\tom-debug.bat**
2. Right click on the Cloud360 in Eclipse, Debug As -> Debug Configuration

Double click Remote Java Application and create new profile with the following details,

Connect: Project – Cloud360 Host: localhost port:81

Source: Add the following Resources,

Java - \Cloud\_one360\mc\ui\src\main

Java - \Cloud\_one360\mc\ic\src\main

Java - \Cloud\_one360\mc\dal\src\main

And click Debug button and launch the Application.

13) **Creating quartz DB**

Its mandatory to select OS DropDown, while creating Instances. Initially OS dropdown will not have data.

For populating that,

Open MYSQL PROMT and run the following scripts,

D:\Cloud Workspace\Cloud\_one360\mc\mc-liquibase-changelog\install\tab\quartz\QuartzDB\_CreateDBScript.sql

D:\Cloud Workspace\Cloud\_one360\mc\mc-liquibase-changelog\install\tab\quartz\QuartzDB\_Schema\_2.x.sql

14) To connect with Cloud providers, we need to change the proxy settings as shown below,

In cloudonedb ---> systemspec table --->

change,

specKey: proxyRequired

specValue: true

specKey: proxyHost

specValue: 10.236.194.206