



Continuous Integration & Deploying using Jenkins and uDeploy

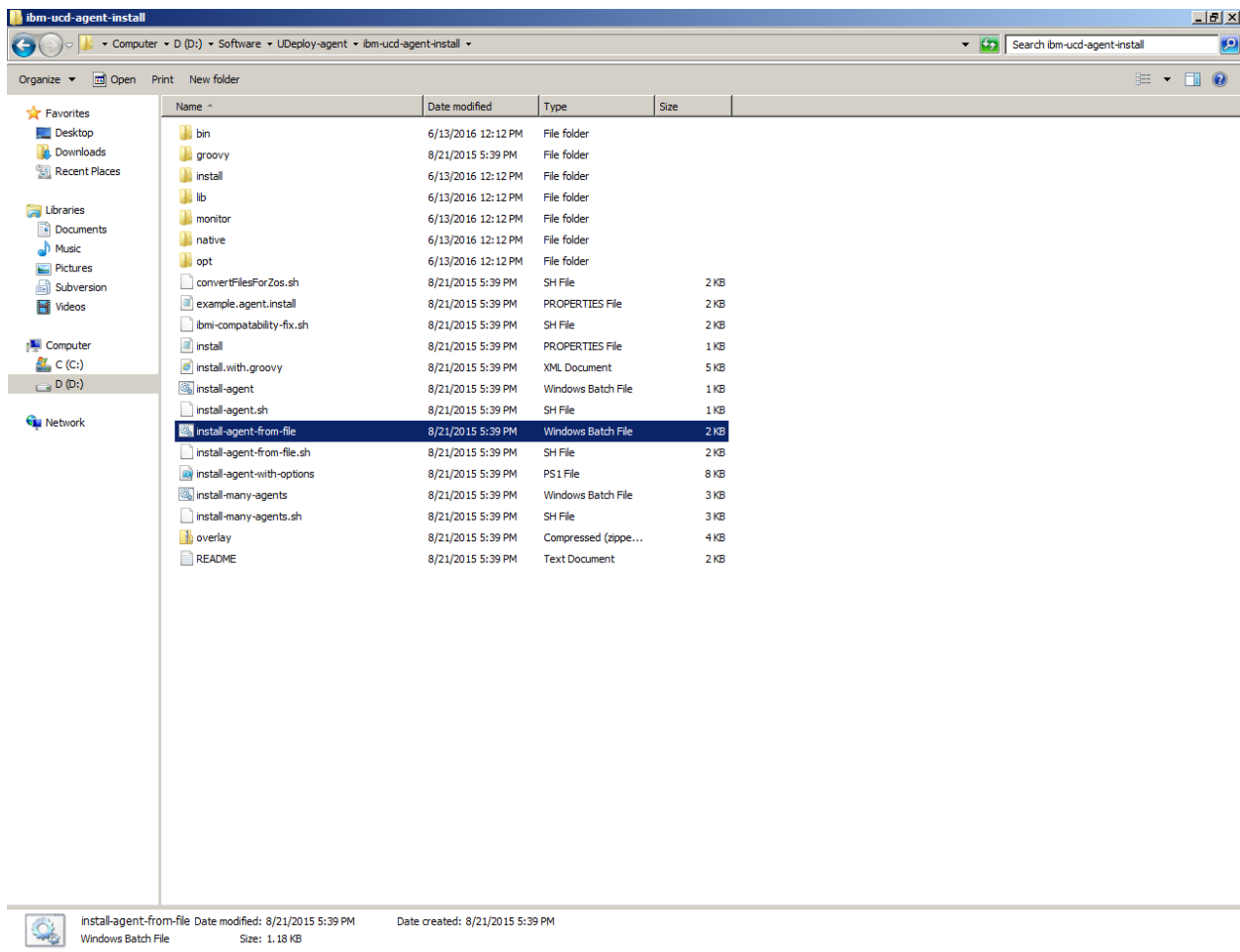
(Projects used are of Java,PHP&Dotnet)

Contents

Installing the UDeploy Agent	3
Installing the Jenkins agent (Master-Slave Configuration)	5
Deploying jpetstore in the Tomcat with the help of the JenkinsPlugin	9
1. Check-in the Jpetstore project in a repository in the SVN.....	9
2. Committing the code in the subversion.....	9
3. Running the project on the slave node:.....	10
4. Pre Steps (SonarQube Scanner Configuration in Jenkins)	11
5. Post Build Action (Deployment in the Tomcat Server)	12
6. Build the Jenkins job.	13
7. Tomcat Deployment	16
Deployment of the php project	19
1. Check-in php code in SVN	19
2. Analyzing php project in Jenkins with Sonar integration.....	20
3. Deployment.....	24
MS Deploy using UDeploy	25
1. Pre-requisites	25
2. Steps to be followed in IBM UDeploy to deploy the project	25
Deployment In IIS Server	39
WEB DEPLOYED	41

Installing the UDeploy Agent

Extract the files from the IBM UDeploy Agent folder → go to bin folder and run the install-agent-from-file.cmd command through command prompt.



Just after this a command prompt will ask multiple question set all the answers there once its done the agent will be online on the udeploy server(<http://10.242.181.79:9002>).

```
C:\Windows\system32\cmd.exe
[echo] Do you want to install the Agent as Windows service? y,N [Default: N]
]
y
[echo] Enter a unique service name for the Agent. No spaces please. [Default:
t: ibm-urbancode-deploy-agent]

[echo] Enter the user account name including domain path to run the service
as <for local use '.' before login>, The local system account will be used by
default. [Default: .\localsystem]

[echo] Do you want to start the 'ibm-urbancode-deploy-agent' service automa
tically? y,N [Default: N]
y
[exec] Checking if service with given name is already installed...
[exec] Service 'ibm-urbancode-deploy-agent' is not installed. Installing...

[exec] Service 'ibm-urbancode-deploy-agent' has been successfully installed

[exec] Agent service ibm-urbancode-deploy-agent started successfully
[propertyfile] Updating property file: D:\Software\IBM\UCDeploy-agent\conf\agent
\installed.properties
[delete] Deleting: D:\Software\IBM\UCDeploy-agent\bin\service\svc.properties

[echo] Installer Complete. <press return to exit installer>
```

UDeploy - agent installed

You can check it on the web server of the IBM UDeploy (<http://10.242.181.79:9002>).

The screenshot shows the IBM UrbanCode Deploy web interface. The top navigation bar includes 'Dashboard', 'Components', 'Applications', 'Configuration', 'Processes', 'Resources', 'Calendar', 'Work Items', 'Reports', and 'Settings'. The 'Resources' section is active, showing a list of agents. The 'seema' agent is highlighted with a red box and is in an 'Online' status. The table below shows the details of the agents.

Name	Description	Status	License	Version
Lakshmi		Online	Unlicensed	6.1.1.8.692130
LT040728.cts.com		Offline	Unlicensed	6.1.1.8.692130
PC248889.cts.com		Online	Unlicensed	6.1.1.8.692130
pravinth4		Offline	Unlicensed	6.1.1.8.692130
Sachin		Online	Unlicensed	6.1.1.8.692130
seema		Online	Unlicensed	6.1.1.8.692130
Stefi		Online	Unlicensed	6.1.1.8.692130
yavana		Online	Unlicensed	6.1.1.8.692130

Installing the Jenkins agent(Master-Slave Configuration)

Go to Manage Jenkins → manage node → click on New Node → Give the Node name and make it a dumb slave.

The screenshot shows the Jenkins web interface for adding a new node. The 'Node name' field is highlighted with a red circle and contains the text 'ragavi'. The 'Dumb Slave' radio button is selected, and the 'Copy Existing Node' option is also visible. The left sidebar shows the 'Build Queue' and 'Build Executor Status' sections. The 'Build Executor Status' section shows the following information:

Node Name	Number of Executors	Status
master	1	Idle
master	2	Idle
Swati	1	Idle
ragavi_node	1	Idle

While making the slave configuration give the root directory name of the folder where we want to store our workspace in the slave's machine.

The screenshot shows the Jenkins web interface. At the top, there is a search bar and the user name 'seema' with a 'log out' link. The main navigation bar includes 'Jenkins' and 'Nodes'. On the left sidebar, there are links for 'Back to Dashboard', 'Manage Jenkins', 'New Node', and 'Configure'. Below these are two expandable sections: 'Build Queue' (showing 'No builds in the queue.') and 'Build Executor Status' (showing a list of nodes: 'master' with 1 idle, 'Swati' with 1 idle, and 'ragavi_node' with 1 idle). The main content area is the configuration page for the 'ragavi' node. It contains several form fields: 'Name' (ragavi), 'Description' (empty), '# of executors' (1), 'Remote root directory' (D:\Jenkins), 'Labels' (Ragavi), 'Usage' (Utilize this node as much as possible), 'Launch method' (Launch slave agents via Java Web Start), and 'Availability' (Keep this slave on-line as much as possible). There is an 'Advanced...' button next to the 'Launch method' field. At the bottom, there are checkboxes for 'Environment variables' and 'Tool Locations', both of which are currently unchecked.

Now you need to connect your slave machine to the master using the following steps.

1. Open a browser on the slave machine and go to the Jenkins master server URL.
2. Go to **Manage Jenkins, Manage Nodes**, Click on the newly created slave machine.
3. Download the slave.jar in the slave machine.

4. Browse the slave.jar just downloaded → go to command prompt → copy and paste the command given in the window for running the slave from command line.

Jenkins search seema | log out

Jenkins > Nodes > ragavi_node ENABLE AUTO REFRESH

[Back to List](#)
[Status](#)
[Delete Slave](#)
[Configure](#)
[Build History](#)
[Load Statistics](#)
[Log](#)

Build Executor Status

Slave ragavi_node Mark this node temporarily offline

⚠ Ping response time is too long or timed out.

Connect slave to Jenkins one of these ways:

- [Launch](#) Launch agent from browser on slave
- Run from slave command line:

```
java -jar slave.jar -jnlpUrl http://localhost:9003/computer/ragavi_node/slave-agent.jnlp -secret 21774fad3a20387e93d5e370441de998454e52d495e6159c2d2075b068e94e48
```

Created by [seema](#)

Labels

[Ragavi_Slave](#)

Projects tied to ragavi_node

None

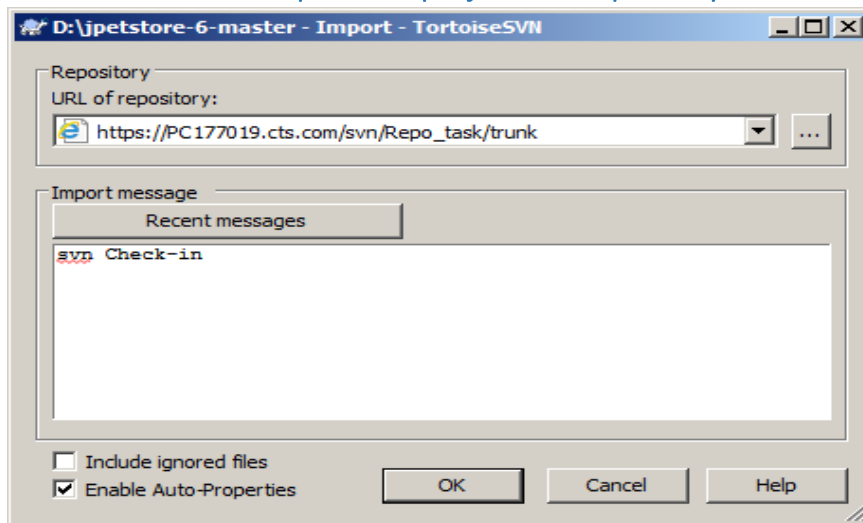
5. Jenkins Slave connected to the master machine.

```
Administrator: C:\Windows\System32\cmd.exe - java -jar slave.jar -jnlpUrl http://10.219.17.239:8085...
Microsoft Windows [Version 6.1.7601]
Copyright (c) 2009 Microsoft Corporation. All rights reserved.

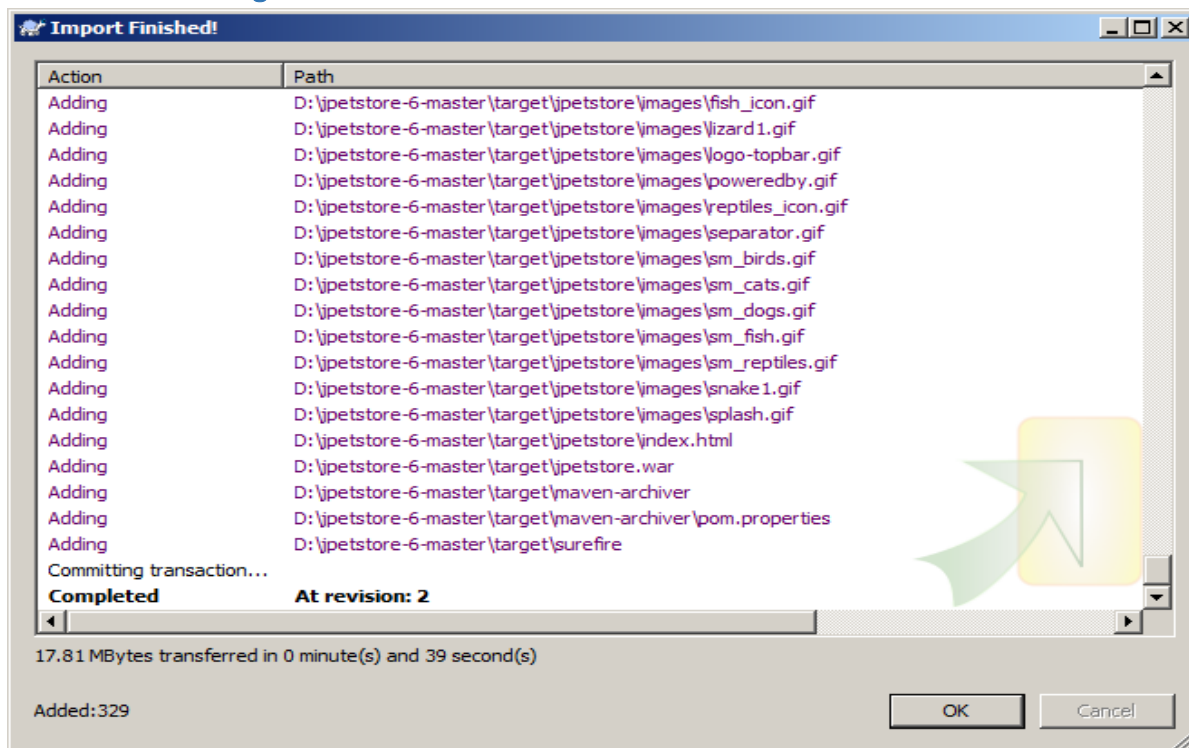
C:\Users\514356\Downloads>java -jar slave.jar -jnlpUrl http://10.219.17.239:8085
/computer/Seema_new/slave-agent.jnlp -secret a1df4ab3c10d1e47e49f72f38d431394bf9
ee76b542e3f15afe4e4404025377f
Jun 13, 2016 5:59:30 PM hudson.remoting.jnlp.Main createEngine
INFO: Setting up slave: Seema_new
Jun 13, 2016 5:59:30 PM hudson.remoting.jnlp.Main$CuiListener <init>
INFO: Jenkins agent is running in headless mode.
Jun 13, 2016 5:59:30 PM hudson.remoting.jnlp.Main$CuiListener status
INFO: Locating server among [http://localhost:8085/, http://10.219.17.239:8085/]
Jun 13, 2016 5:59:31 PM hudson.remoting.jnlp.Main$CuiListener status
INFO: Handshaking
Jun 13, 2016 5:59:31 PM hudson.remoting.jnlp.Main$CuiListener status
INFO: Connecting to 10.219.17.239:49216
Jun 13, 2016 5:59:31 PM hudson.remoting.jnlp.Main$CuiListener status
INFO: Trying protocol: JNLP2-connect
Jun 13, 2016 5:59:31 PM hudson.remoting.jnlp.Main$CuiListener status
INFO: Connected
```

Deploying jpetstore in the Tomcat with the help of the Jenkins Plugin

1. Check-in the Jpetstore project in a repository in the SVN.



2. Committing the code in the subversion



3. Running the project on the slave node:

- Restricting the project running on the specified slave node will make the user to create the workspace in the slave's machine.
- Polling the subversion Repository every 15 minutes will allow Jenkins to poll the code change in every 15 minutes of time and build the code changes.

Source Code Management

- None
- CVS
- CVS Projectset
- File System
- Subversion

Modules

Repository URL	<input type="text" value="https://PC177019.cts.com/svn/jpetstore_repo_seema/trunk"/>	
Local module directory (optional)	<input type="text" value="."/>	
Repository depth	<input type="text" value="infinity"/>	
Ignore externals	<input type="checkbox"/>	

Add more locations...

Check-out Strategy	<input type="text" value="Use 'svn update' as much as possible"/>	
<small>Use 'svn update' whenever possible, making the build faster. But this causes the artifacts from the previous build to remain when a new build starts.</small>		

Repository browser	<input type="text" value="(Auto)"/>	
--------------------	-------------------------------------	--

Advanced...

Build Triggers

- Build whenever a SNAPSHOT dependency is built
- Trigger builds remotely (e.g., from scripts)
- Build after other projects are built
- Build periodically
- Maven Dependency Update Trigger
- Poll SCM

Schedule	<input type="text" value="H/15 * * * *"/>	
----------	---	--

4. Pre Steps (SonarQube Scanner Configuration in Jenkins)

Add Execute SonarQube Scanner in Pre Step and as here maven project is used the pom.xml will require the Goals. Set the goal as clean install sonar: sonar so that it will be analyzed in the sonar automatically.

Pre Steps

Execute SonarQube Scanner

SonarQube Installation: ?

Task to run ?

JDK ?
JDK to be used for this sonar analysis

Path to project properties ?

Analysis properties ?

Additional arguments ?

JVM Options ?

Delete

Add pre-build step ▼

Build

Root POM ?

Goals and options ?

Advanced...

Post Steps

Run only if build succeeds Run only if build succeeds or is unstable Run regardless of build result
Should the post-build steps run only for successful builds, etc.

Add post-build step ▼

Build Settings

Save **Apply**

5. Post Build Action (Deployment in the Tomcat Server)

- Scroll down to bottom of page to “**Post Build Actions**”. Select “**Deploy war/ear to container**” from “**Add Post Build Actions**”.
- For WAR file you can use wild cards, e.g. ****/*.war**. The context path is the path of the URL under which your application will be published in Tomcat.
- Select the appropriate Tomcat version and for Manager User name and password just copy the name and password from Tomcat-users.xml file.
- The tomcat URL is the base URL under which your application can be reached.

Post Steps

Run only if build succeeds Run only if build succeeds or is unstable Run regardless of build result
Should the post-build steps run only for successful builds, etc.

Add post-build step ▾

Build Settings

E-mail Notification

Post-build Actions

Deploy war/ear to a container

WAR/EAR files ?

Context path ?

Containers

Tomcat 7.x

Manager user name

Manager password

Tomcat URL

Delete



Add Container ▾

Deploy on failure












Delete

Here, the Deploy war/ear to a container plugin is configured in Jenkins, this plugin will automatically deploy war file to the Tomcat server.

6. Build the Jenkins job.

 **Jenkins** search  admin

Jenkins > seema_new_maven > #14

-  Back to Project
-  Status
-  Changes
-  **Console Output**
 -  View as plain text
-  Edit Build Information
-  Delete Build
-  Tag this build
-  Redeploy Artifacts
-  See Fingerprints
-  Previous Build

Console Output

Started by user [admin](#)
Building on master in workspace D:\Swati Jenkins\workspace\seema_new_maven
Switching from https://pc177019.cts.com/svn/jpetstore_repo/trunk to https://PC177019.cts.com/svn/jpetstore_repo/trunk
Switching to https://PC177019.cts.com/svn/jpetstore_repo/trunk at revision '2016-06-20T17:43:27.828 +0530'

```
A .sonar\sonar_lock
U target\classes\org\mybatis\jpetstore\domain\Account.class
U target\classes\org\mybatis\jpetstore\domain\Cart.class
U target\classes\org\mybatis\jpetstore\domain\CartItem.class
U target\classes\org\mybatis\jpetstore\domain\Category.class
U target\classes\org\mybatis\jpetstore\domain\Item.class
U target\classes\org\mybatis\jpetstore\domain\LineItem.class
U target\classes\org\mybatis\jpetstore\domain\Order.class
U target\classes\org\mybatis\jpetstore\domain\Product.class
U target\classes\org\mybatis\jpetstore\domain\Sequence.class
U target\classes\org\mybatis\jpetstore\persistence\AccountMapper.class
U target\classes\org\mybatis\jpetstore\persistence\CategoryMapper.class
U target\classes\org\mybatis\jpetstore\persistence\ItemMapper.class
U target\classes\org\mybatis\jpetstore\persistence\LineItemMapper.class
U target\classes\org\mybatis\jpetstore\persistence\OrderMapper.class
U target\classes\org\mybatis\jpetstore\persistence\ProductMapper.class
U target\classes\org\mybatis\jpetstore\persistence\SequenceMapper.class
U target\classes\org\mybatis\jpetstore\service\AccountService.class
U target\classes\org\mybatis\jpetstore\service\CatalogService.class
U target\classes\org\mybatis\jpetstore\service\OrderService.class
U target\classes\org\mybatis\jpetstore\web\actions\AbstractActionBean.class
U target\classes\org\mybatis\jpetstore\web\actions\AccountActionBean.class
U target\classes\org\mybatis\jpetstore\web\actions\CartActionBean.class
U target\classes\org\mybatis\jpetstore\web\actions\CatalogActionBean.class
U target\classes\org\mybatis\jpetstore\web\actions\OrderActionBean.class
U target\jpetstore\WEB-INF\classes\org\mybatis\jpetstore\domain\Account.class
U target\jpetstore\WEB-INF\classes\org\mybatis\jpetstore\domain\Cart.class
U target\jpetstore\WEB-INF\classes\org\mybatis\jpetstore\domain\CartItem.class
U target\jpetstore\WEB-INF\classes\org\mybatis\jpetstore\domain\Category.class
U target\jpetstore\WEB-INF\classes\org\mybatis\jpetstore\domain\Item.class
U target\jpetstore\WEB-INF\classes\org\mybatis\jpetstore\domain\LineItem.class
U target\jpetstore\WEB-INF\classes\org\mybatis\jpetstore\domain\Order.class
U target\jpetstore\WEB-INF\classes\org\mybatis\jpetstore\domain\Product.class
U target\jpetstore\WEB-INF\classes\org\mybatis\jpetstore\domain\Sequence.class
U target\jpetstore\WEB-INF\classes\org\mybatis\jpetstore\persistence\AccountMapper.class
U target\jpetstore\WEB-INF\classes\org\mybatis\jpetstore\persistence\CategoryMapper.class
U target\jpetstore\WEB-INF\classes\org\mybatis\jpetstore\persistence\ItemMapper.class
U target\jpetstore\WEB-INF\classes\org\mybatis\jpetstore\persistence\LineItemMapper.class
U target\jpetstore\WEB-INF\classes\org\mybatis\jpetstore\persistence\OrderMapper.class
U target\jpetstore\WEB-INF\classes\org\mybatis\jpetstore\persistence\ProductMapper.class
U target\jpetstore\WEB-INF\classes\org\mybatis\jpetstore\persistence\SequenceMapper.class
U target\jpetstore\WEB-INF\classes\org\mybatis\jpetstore\service\AccountService.class
U target\jpetstore\WEB-INF\classes\org\mybatis\jpetstore\service\CatalogService.class
U target\jpetstore\WEB-INF\classes\org\mybatis\jpetstore\service\OrderService.class
U target\jpetstore\WEB-INF\classes\org\mybatis\jpetstore\web\actions\AbstractActionBean.class
U target\jpetstore\WEB-INF\classes\org\mybatis\jpetstore\web\actions\AccountActionBean.class
U target\jpetstore\WEB-INF\classes\org\mybatis\jpetstore\web\actions\CartActionBean.class
U target\jpetstore\WEB-INF\classes\org\mybatis\jpetstore\web\actions\CatalogActionBean.class
U target\jpetstore\WEB-INF\classes\org\mybatis\jpetstore\web\actions\OrderActionBean.class
U target\jpetstore\WEB-INF\lib\aoapalliance-1.0.jar
U target\jpetstore\WEB-INF\lib\commons-logging-1.1.3.jar
```

```
[INFO] Sensor JavaSquidSensor (done) | time=3599ms
[INFO] Sensor Lines Sensor
[INFO] Sensor Lines Sensor (done) | time=15ms
[INFO] Sensor QProfileSensor
[INFO] Sensor QProfileSensor (done) | time=0ms
[INFO] Sensor SurefireSensor
[INFO] parsing D:\Swati Jenkins\workspace\seema_new_maven\target\surefire-reports
[INFO] Sensor SurefireSensor (done) | time=0ms
[INFO] Sensor XmlFileSensor
[INFO] 3 source files to be analyzed
[INFO] Sensor XmlFileSensor (done) | time=445ms
[INFO] Sensor XmlSensor
[INFO] 3/3 source files have been analyzed
[INFO] Sensor XmlSensor (done) | time=203ms
[INFO] Sensor SCM Sensor
[INFO] SCM Publisher is disabled
[INFO] Sensor SCM Sensor (done) | time=0ms
[INFO] Sensor Code Colorizer Sensor
[INFO] Sensor Code Colorizer Sensor (done) | time=2ms
[INFO] Sensor CPD Sensor
[INFO] JavaCpdIndexer is used for java
[INFO] DefaultCpdIndexer is used for xml
[INFO] Sensor CPD Sensor (done) | time=74ms
[INFO] Analysis report generated in 196ms, dir size=180 KB
[INFO] Analysis reports compressed in 499ms, zip size=93 KB
[INFO] Analysis report uploaded in 82ms
[INFO] ANALYSIS SUCCESSFUL, you can browse http://localhost:9000/dashboard/index/org.mybatis.jpjpetstore
[INFO] Note that you will be able to access the updated dashboard once the server has processed the submitted analysis report
[INFO] More about the report processing at http://localhost:9000/api/ce/task?id=AVVe1JNgUv16tZJTcu7u
[INFO] Executing post-job org.sonar.plugins.buildbreaker.ForbiddenConfigurationBreaker
[INFO] Executing post-job org.sonar.plugins.buildbreaker.QualityGateBreaker
[INFO] Waiting for report processing to complete...
[INFO] Quality gate status: OK
[INFO] -----
[INFO] BUILD SUCCESS
[INFO] -----
[INFO] Total time: 28.322 s
[INFO] Finished at: 2016-06-17T17:00:51+05:30
[INFO] Final Memory: 43M/604M
[INFO] -----
[JENKINS] Archiving D:\Swati Jenkins\workspace\seema_new_maven\pom.xml to org.mybatis.jpjpetstore/6.0.2-SNAPSHOT/jpetstore-6.0.2-SNAPSHOT.pom
[JENKINS] Archiving D:\Swati Jenkins\workspace\seema_new_maven\target\jpetstore.war to org.mybatis.jpjpetstore/6.0.2-SNAPSHOT/jpetstore-6.0.2-SNAPSHOT.war
channel stopped
Deploying D:\Swati Jenkins\workspace\seema_new_maven\target\jpetstore.war to container Tomcat 7.x Remote
[D:\Swati Jenkins\workspace\seema_new_maven\target\jpetstore.war] is not deployed. Doing a fresh deployment.
Deploying [D:\Swati Jenkins\workspace\seema_new_maven\target\jpetstore.war]
Finished: SUCCESS
```

The sonar URL given in the Jenkins build (Console Output) will redirect you directly to the sonar analysis of the project as shown below:

The screenshot displays the SonarQube dashboard for a project named 'JPetStore Demo 6'. The interface includes a top navigation bar with options like 'Dashboards', 'Issues', 'Measures', 'Rules', 'Quality Profiles', and 'Quality Gates'. The main content area is divided into several sections:

- Quality Gate:** A green 'Passed' badge is shown.
- Technical Debt:** A green 'A' grade is displayed. Metrics include 2d Debt (started 2 days ago) and 35 Issues. A 'Leak Period' section indicates that the period since the previous version started 2 days ago, with 0 New Debt and 0 New Issues.
- Duplications:** A progress indicator shows 1.5% Duplications, corresponding to 2 Duplicated Blocks. A change of +0.0% Duplications is noted.
- Structure:** A bar chart shows code coverage: Java at 88.3% and XML at 11.7%. The total Lines of Code are 1.6k, with a change of +0 Lines of Code.


On the right side, there is a sidebar with navigation links for Bug Tracker, Continuous integration, Developer connection, Home, and Sources. Below these are sections for Quality Gate (Default: SonarQube way), Quality Profiles (live: Sonar way, XML: Sonar way), and Events (All). The footer indicates the version is 6.0.2-SNAPSHOT, dated June 17, 2018.

7. Tomcat Deployment

For viewing the deployment browse the **Tomcat URL** (the URL given in the Jenkins in Deploy to container plugin configuration). → Click on **Manager App**.


Home Documentation Configuration Wiki Mailing Lists Find Help

Apache Tomcat/7.0.12



http://www.apache.org/

If you're seeing this, you've successfully installed Tomcat. Congratulations!



Recommended Reading:

- [Security Considerations HOW-TO](#)
- [Manager Application HOW-TO](#)
- [Clustering/Session Replication HOW-TO](#)

Server Status

Manager App

Host Manager

Developer Quick Start

Tomcat Setup	Realms & AAA	Servlet Examples	Servlet Specifications
First Web Application	JDBC DataSources	JSP Examples	Tomcat Versions

Managing Tomcat

For security, access to the [manager webapp](#) is restricted. Users are defined in:

```
$CATALINA_HOME/conf/tomcat-users.xml
```

In Tomcat 7.0 access to the manager application is split between different users. [Read more...](#)

[Release Notes](#)

[Changelog](#)

[Migration Guide](#)

[Security Updates](#)

Documentation

[Tomcat 7.0 Documentation](#)

[Tomcat 7.0 Configuration](#)

[Tomcat Wiki](#)

Find additional important configuration information in:

```
$CATALINA_HOME/RUNNING.txt
```

Developers may be interested in:

- [Tomcat 7.0 Bug Database](#)
- [Tomcat 7.0 JavaDocs](#)
- [Tomcat 7.0 SVN Repository](#)
- [Tomcat 7.0 Examples](#)

Getting Help

[FAQ](#)

[Mailing Lists](#)

The following mailing lists are available:

- announce@tomcat.apache.org
Important announcements, releases, security vulnerability notifications. (Low volume).
- users@tomcat.apache.org
User support and discussion
- taqlibs-user@tomcat.apache.org
User support and discussion for [Apache Taqlibs](#)
- dev@tomcat.apache.org
Development mailing list, including commit messages

Other Downloads	Other Documentation	Get Involved	Miscellaneous	Apache Software Foundation
Tomcat Connectors	Tomcat Connectors	Overview	Contact	Who We Are
Tomcat Native	mod_ik Documentation	SVN Repositories	Legal	Heritage
Taqlibs	Tomcat Native	Mailing Lists	Sponsorship	Apache Home
Deployer	Deployer	Wiki	Thanks	Resources

The deployment will be visible over here.



Tomcat Web Application Manager

Message: OK

Manager			
List Applications	HTML Manager Help	Manager Help	Server Status

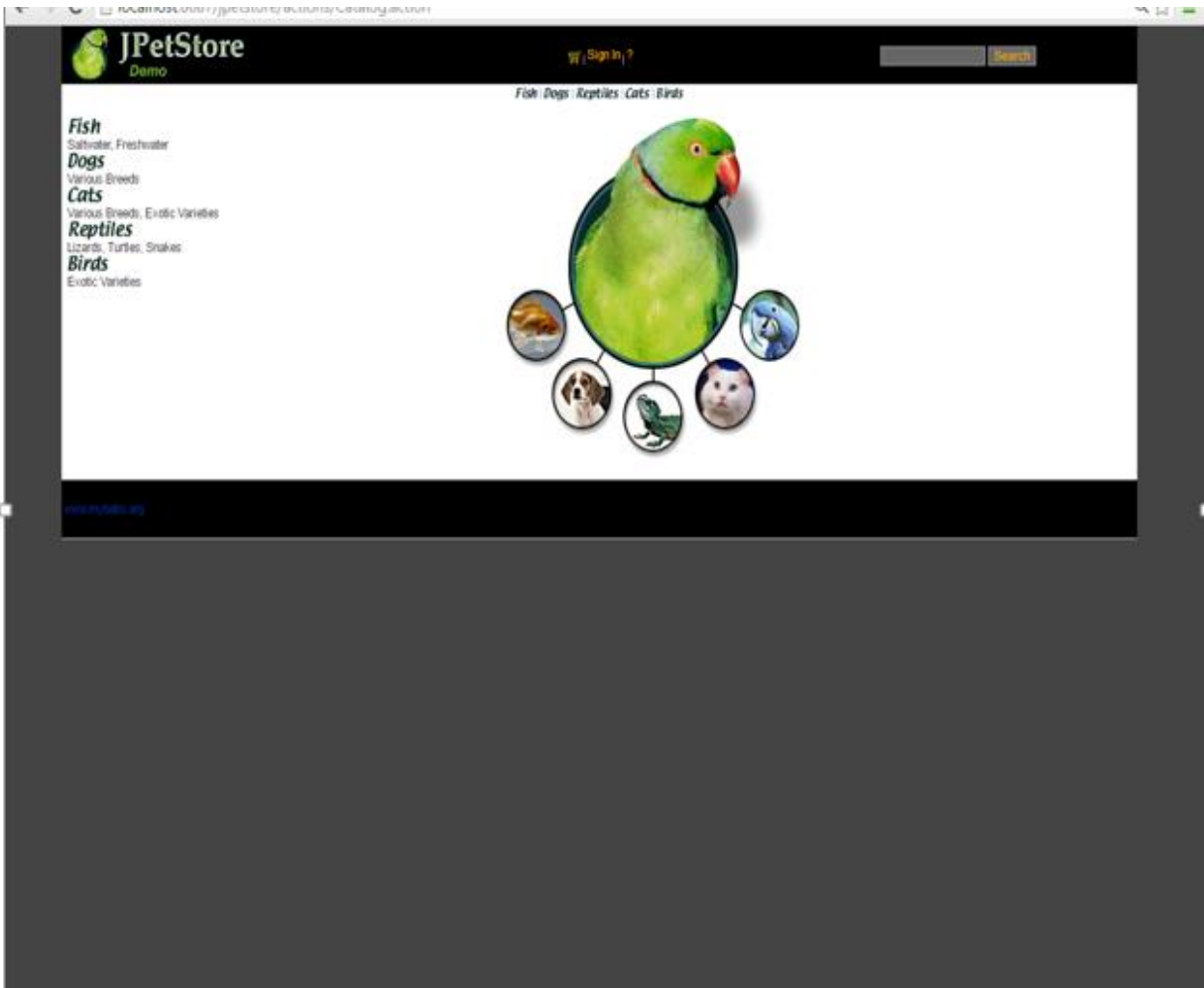
Applications					
Path	Version	Display Name	Running	Sessions	Commands
/	None specified	Welcome to Tomcat	true	2	Start <input type="button" value="Stop"/> <input type="button" value="Reload"/> <input type="button" value="Undeploy"/> <input type="button" value="Expire sessions"/> with idle ≥ <input type="text" value="30"/> minutes
/artifactory	None specified	artifactory-webapp	true	0	Start <input type="button" value="Stop"/> <input type="button" value="Reload"/> <input type="button" value="Undeploy"/> <input type="button" value="Expire sessions"/> with idle ≥ <input type="text" value="120"/> minutes
/docs	None specified	Tomcat Documentation	true	0	Start <input type="button" value="Stop"/> <input type="button" value="Reload"/> <input type="button" value="Undeploy"/> <input type="button" value="Expire sessions"/> with idle ≥ <input type="text" value="30"/> minutes
/examples	None specified	Servlet and JSP Examples	true	0	Start <input type="button" value="Stop"/> <input type="button" value="Reload"/> <input type="button" value="Undeploy"/> <input type="button" value="Expire sessions"/> with idle ≥ <input type="text" value="30"/> minutes
/host-manager	None specified	Tomcat Host Manager Application	true	0	Start <input type="button" value="Stop"/> <input type="button" value="Reload"/> <input type="button" value="Undeploy"/> <input type="button" value="Expire sessions"/> with idle ≥ <input type="text" value="30"/> minutes
/jpetstore	None specified	JPetStore	true	0	Start <input type="button" value="Stop"/> <input type="button" value="Reload"/> <input type="button" value="Undeploy"/> <input type="button" value="Expire sessions"/> with idle ≥ <input type="text" value="30"/> minutes
/manager	None specified	Tomcat Manager Application	true	1	Start <input type="button" value="Stop"/> <input type="button" value="Reload"/> <input type="button" value="Undeploy"/> <input type="button" value="Expire sessions"/> with idle ≥ <input type="text" value="30"/> minutes
/new_jpetstore	None specified	JPetStore	true	0	Start <input type="button" value="Stop"/> <input type="button" value="Reload"/> <input type="button" value="Undeploy"/> <input type="button" value="Expire sessions"/> with idle ≥ <input type="text" value="30"/> minutes

Deploy

Click on the jpetstore → it will redirect to the web deployed jpetstore.

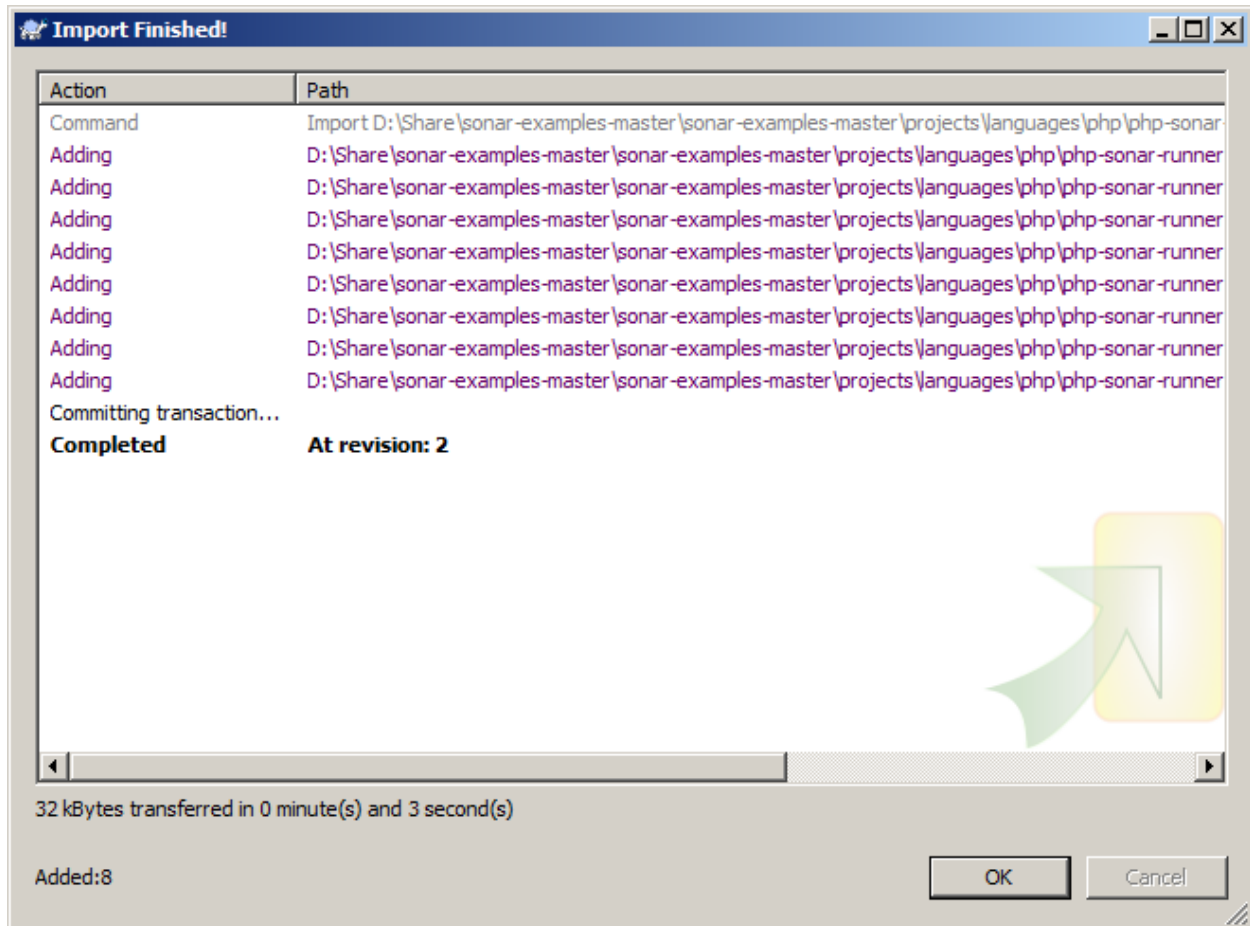


Click on the **Enter the store** → the below shown page will be displayed for jpetstore



Deployment of the php project

1. Check-in php code in SVN



2. Analyzing php project in Jenkins with Sonar integration

Set the following configuration for building Jenkins job with sonar analysis.

- Take the project from the subversion repository (give the appropriate repository URL), you can even poll the SCM by setting the schedule for it.

Project name

Description

[Plain text] [Preview](#)

Discard Old Builds ?

This build is parameterized ?

Disable Build (No new builds will be executed until the project is re-enabled.) ?

Execute concurrent builds if necessary ?

Restrict where this project can be run ?

Advanced Project Options

Source Code Management

None

CVS

CVS Projectset

File System

Subversion

Modules

Repository URL ?

Local module directory (optional) ?

Repository depth ?

Ignore externals ?

Check-out Strategy ▼

Use 'svn update' whenever possible, making the build faster. But this causes the artifacts from the previous build to remain when a new build starts.

- Poll the SVN repository by scheduling the time. (Polling the code from the SVN every 15 minute through Build trigger poll SCM).

The screenshot displays the Jenkins Job Configuration page. The **Build Triggers** section has the **Poll SCM** checkbox checked, and the **Schedule** field contains the cron expression `H/15 * * * *`. The **Build Environment** section shows various integration options. The **Build** section is highlighted with a red box and includes the following configuration:

- Execute SonarQube Scanner**
 - SonarQube Installation: `sonarqube-5.4`
 - Task to run: (empty)
 - JDK: `(Inherit From Job)`
 - Path to project properties: (empty)
 - Analysis properties:

```
sonar.projectKey=org.sonarqube.php-test
sonar.projectName=PHP :: Test Project
sonar.projectVersion=1.0
sonar.sources=
sonar.language=php
sonar.sourceEncoding=UTF-8
```
 - Additional arguments: (empty)
 - JVM Options: (empty)
- Execute Windows batch command**
 - Command: `xcopy /s *php D:\xampp\htdocs`

A red **Delete** button is visible at the bottom right of the Build section.

- Configure the sonar Scanner and the property file, then after building the php project in Jenkins the below shown is the output which we will get.

- Add the windows batch command for copying the php file to the htdocs folder of the XAMPP for direct deployment of the web application (Here, xcopy /s *.php C:\xampp\htdocs).

JVM Options Delete

Execute Windows batch command

Command Delete

[See the list of available environment variables](#)

Add build step

Post-build Actions

- After all these configuration Save and apply the configuration and build the Jenkins job.

```

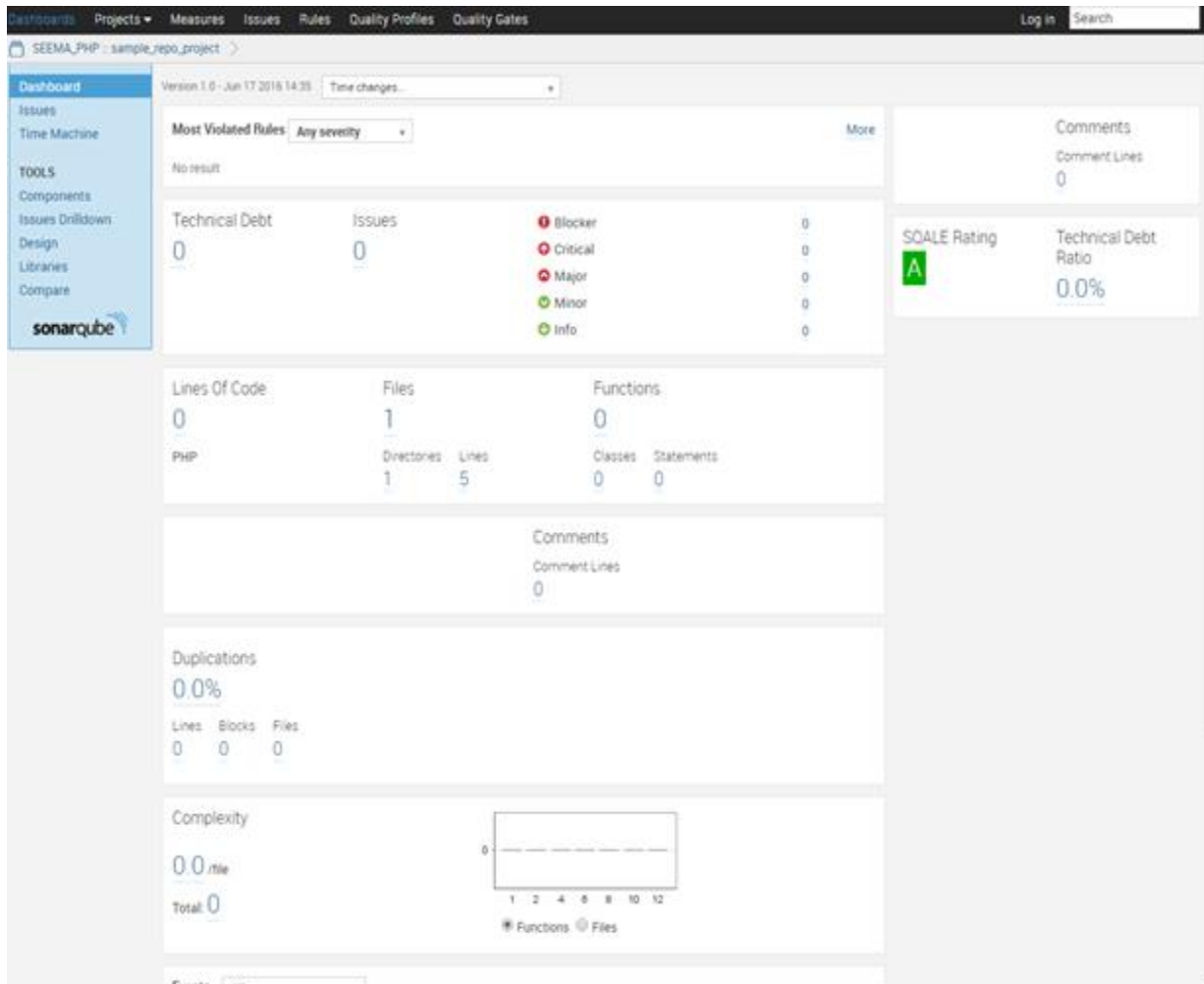
14:30:09.239 INFO - Index files
14:30:09.254 INFO - Excluded sources:
14:30:09.254 INFO -   src/HelloWorld.java
14:30:09.285 INFO - 1 files indexed
14:30:09.285 INFO - 0 files ignored because of inclusion/exclusion patterns
14:30:09.301 INFO - Quality profile for php: Sonar way
14:30:10.103 INFO - JaCoCoSensor: JaCoCo report not found : D:\Swati Jenkins\workspace\Test_Php\target\jacoco.exec
14:30:10.103 INFO - JaCoCoITSensor: JaCoCo IT report not found: D:\Swati Jenkins\workspace\Test_Php\target\jacoco-it.exec
14:30:10.114 INFO - Sensor NoSonar and Commented out LOC Sensor
14:30:10.142 INFO - Sensor NoSonar and Commented out LOC Sensor (done) | time=28ms
14:30:10.143 INFO - Sensor Lines Sensor
14:30:10.144 INFO - Sensor Lines Sensor (done) | time=1ms
14:30:10.145 INFO - Sensor QProfileSensor
14:30:10.150 INFO - Sensor QProfileSensor (done) | time=5ms
14:30:10.150 INFO - Sensor PHPSensor
14:30:10.631 INFO - 1 source files to be analyzed
14:30:10.744 INFO - Sensor PHPSensor (done) | time=594ms
14:30:10.745 INFO - Sensor PHPUnit Sensor
14:30:10.745 INFO - No PHPUnit test report provided (see 'sonar.php.tests.reportPath' property)
14:30:10.745 INFO - No PHPUnit unit test coverage report provided (see 'sonar.php.coverage.reportPath' property)
14:30:10.745 INFO - No PHPUnit integration test coverage report provided (see 'sonar.php.coverage.itReportPath' property)
14:30:10.745 INFO - No PHPUnit overall coverage report provided (see 'sonar.php.coverage.overallReportPath' property)
14:30:10.745 INFO - Sensor PHPUnit Sensor (done) | time=0ms
14:30:10.745 INFO - Sensor SCM Sensor
14:30:10.745 INFO - SCM Publisher is disabled
14:30:10.745 INFO - Sensor SCM Sensor (done) | time=0ms
14:30:10.745 INFO - Sensor Code Colorizer Sensor
14:30:10.746 INFO - 1/1 source files have been analyzed
14:30:10.746 INFO - Sensor Code Colorizer Sensor (done) | time=1ms
14:30:10.746 INFO - Sensor CPD Sensor
14:30:10.746 INFO - DefaultCpdIndexer is used for php
14:30:10.768 INFO - Sensor CPD Sensor (done) | time=22ms
14:30:10.855 INFO - Analysis report generated in 75ms, dir size=18 KB
14:30:10.890 INFO - Analysis reports compressed in 35ms, zip size=7 KB
14:30:10.964 INFO - Analysis report uploaded in 74ms
14:30:10.964 INFO - ANALYSIS SUCCESSFUL, you can browse http://10.219.17.177:9000/dashboard/index/org.sonarqube:php-test
14:30:10.964 INFO - Note that you will be able to access the updated dashboard once the server has processed the submitted analysis report
14:30:10.964 INFO - More about the report processing at http://10.219.17.177:9000/api/ce/task?id=AVVdIskKUv16tZJtcvJ5
14:30:10.971 INFO - Executing post-job org.sonar.plugins.buildbreaker.ForbiddenConfigurationBreaker
14:30:10.971 INFO - Executing post-job org.sonar.plugins.buildbreaker.QualityGateBreaker
14:30:11.326 INFO - Waiting for report processing to complete...
14:30:21.395 INFO - Quality gate status: OK
INFO: -----
INFO: EXECUTION SUCCESS
INFO: -----
Total time: 15.852s
Final Memory: 13M/268M
INFO: -----
[Test_Php] $ cmd /c call C:\Windows\TEMP\hudson5918792318402994045.bat
D:\Swati Jenkins\workspace\Test_Php>xcopy /s *php D:\xampp\htdocs
D:add.php
1 File(s) copied

D:\Swati Jenkins\workspace\Test_Php>exit 0
Finished: SUCCESS

```

This means the project is successfully build in the Jenkins agent .As the Jenkins job is successfully build and copied the .php file from Jenkins workspace to the htdocs folder in XAMPP so we can now easily see the .php file on web.

The link given at the bottom of the above screenshot will redirect us to the sonar dashboard where we can see the sonar analysis for this project.



3. Deployment

After this for viewing the web page, open the XAMPP website and just write the php file name on the website after the host.



Apache Friends Applications FAQs HOW-TO Guides PHPInfo phpMyAdmin

XAMPP Apache + MariaDB + PHP + Perl

Welcome to XAMPP for Windows 5.5.35

You have successfully installed XAMPP on this system! Now you can start using Apache, MariaDB, PHP and other components. You can find more info in the FAQs section or check the HOW-TO Guides for getting started with PHP applications.

Start the XAMPP Control Panel to check the server status.

Community


XAMPP has been around for more than 10 years – there is a huge community behind it. You can get involved by joining our Forums, adding yourself to the Mailing List, and liking us on Facebook, following our exploits on Twitter, or adding us to your Google+ circles.

Contribute to XAMPP translation at translate.apachefriends.org.

Can you help translate XAMPP for other community members? We need your help to translate XAMPP into different languages. We have set up a site, translate.apachefriends.org, where users can contribute translations.

Install applications on XAMPP using Bitnami

Apache Friends and Bitnami are cooperating to make dozens of open source applications available on XAMPP, for free. Bitnami-packaged applications include Wordpress, Drupal, Joomla! and dozens of others and can be deployed with one-click installers. Visit the [Bitnami XAMPP page](#) for details on the currently available apps.



This will redirect you to the made web page in this case it is simple php file.

localhost:1234/seema.php

hello world!!!!!!! My name is Seema.....

MS Deploy using UDeploy

1. Pre-requisites

- Before starting with Udeploy one should have IBM UDeploy agent installed on the system and,
- The one should also install the Jenkins plug-in for IBM Urban Code Deploy on the Jenkins server.
- Download the latest version of the plug-in from the following link:<https://developer.ibm.com/urancode/plugin/jenkins/>.

2. Steps to be followed in IBM UDeploy to deploy the project

The following scenario shows how to use the Jenkins plug-in to publish Jenkins build output to versioned artifacts in IBM UrbanCode Deploy and to deploy the application.

- Create a resource group, and then add the IBM UrbanCode Deploy agent to the resource group, as shown in the following screen capture(click top level group)

The screenshot displays the IBM UrbanCode Deploy web interface. The top navigation bar includes 'Dashboard', 'Components', 'Applications', 'Configuration', 'Processes', 'Resources', 'Calendar', 'Work Items', 'Reports', and 'Settings'. The breadcrumb trail shows 'Home > Resources'. Below the navigation, there are tabs for 'Resource Tree', 'Resource Templates', 'Agents', 'Agent Pools', and 'Cloud Connections'. A toolbar contains a 'Create Top-Level Group' button, a 'Select All...' dropdown, an 'Actions...' dropdown, and a 'Show' dropdown. The main area is a table with columns for 'Name', 'Inventory', 'Status', and 'Description'. A modal dialog titled 'Create Resource' is open in the foreground, featuring a search icon, a close button, and input fields for 'Name', 'Include Agents Automatically' (checkbox), 'Description', and 'Teams' (with a plus sign). 'Save' and 'Cancel' buttons are at the bottom of the dialog. The background table shows a tree view of resource groups, with 'seema_hello' selected. The 'seema' resource is listed with a status of 'Offline'.

Name	Inventory	Status	Description
Resource Name	Tags		
Comcast-Dev			
HelloWorld_Sachin			
helloworldTutorial			
jpetstore_maven			
Ragavi_dotnet			Dotnet_project
sampleTutorial			seema
seema_hello		Offline	
seema (View Agent)			
stefi_agent			
Stefi_Hello			
Test			
UAT			
vehicle_store_seema			seema

ii. Create components in IBM UrbanCode Deploy

- Click the **Components** tab and then click **Create Component**. In this window, you specify where the artifacts for the component are.
- In the **Name** field, give your component name.
- In the **Source Configuration Type** list, select **Subversion** and pass the required repository URL.
- Click **Save**.

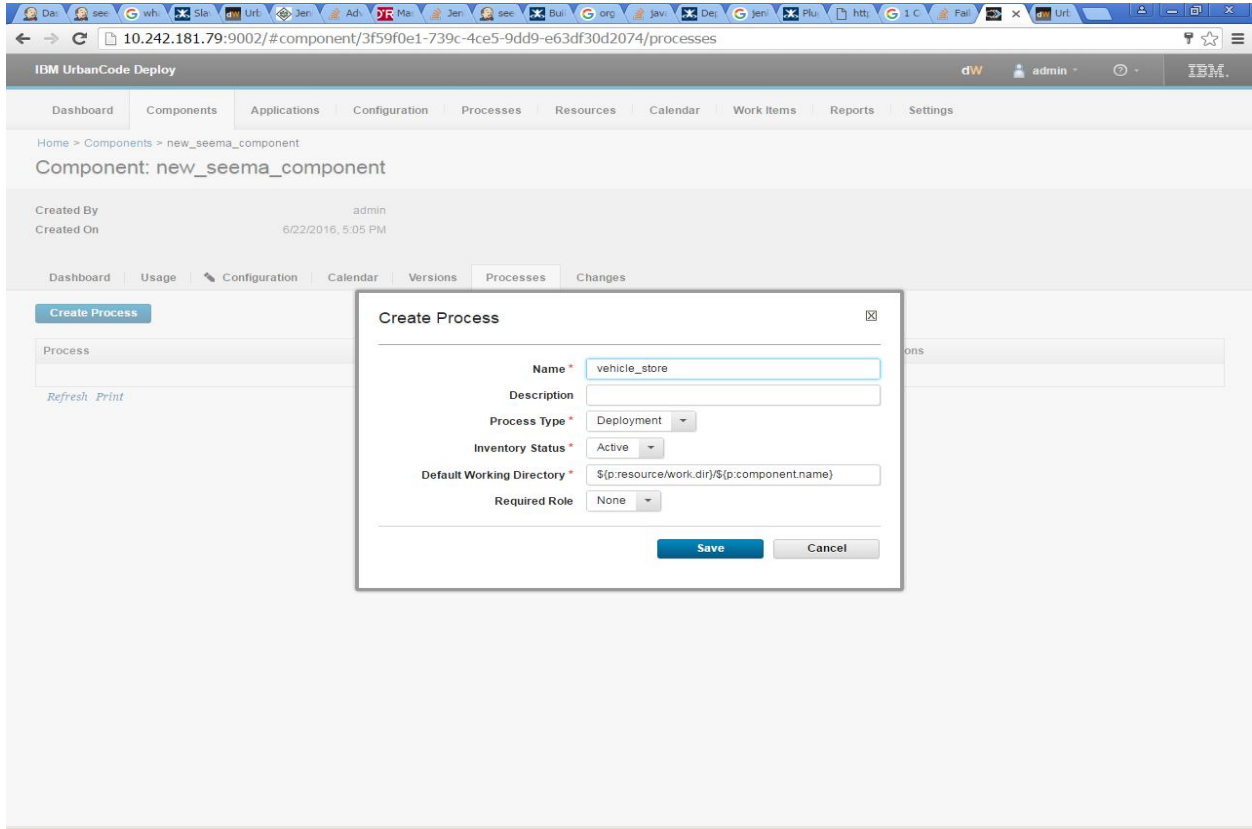
The screenshot shows the IBM UrbanCode Deploy interface with the 'Create Component' dialog box open. The background shows the 'Components' tab with a list of existing components like 'Hello_Seema', 'hello_world_component', and 'seema'. The 'Create Component' dialog is titled 'Create Component' and contains the following fields and options:

- Name ***: new_seema_component
- Description**: (empty)
- Teams**: +
- Template**: None
- Component Type**: Standard
- Source Configuration Type**: Subversion
- Repository URL ***: https://PC177019.cts.com/svn/dotnet_repo/trunk
- Watch for tags**:
- Includes**: **/*
- Excludes**: (empty)
- User**: seema
- Password**: ****
- SVN Path ***: svn
- Preserve Execute Permissions**:
- Extensions of files to Convert**: .bt_log,properties
- Import Versions Automatically**:
- Copy to Code Station**:
- Default Version Type ***: Full

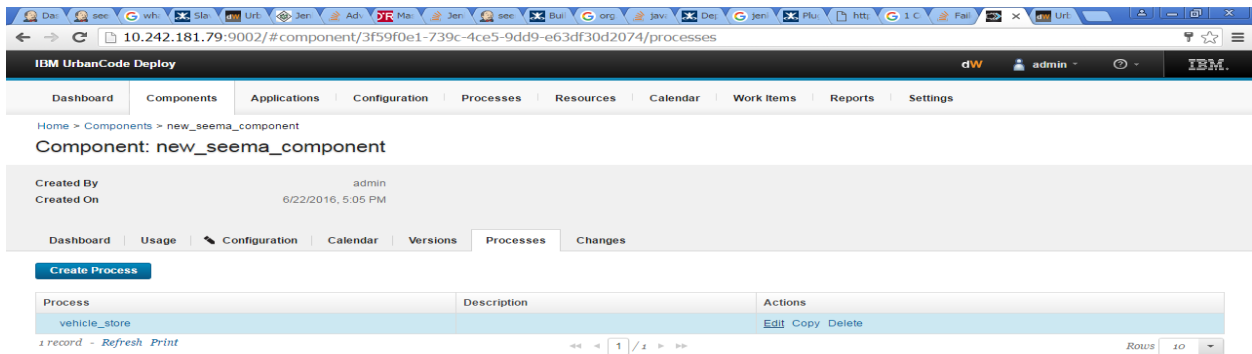
At the bottom of the dialog, there are three radio buttons for version import options:

- Use the system's default version import agent/tag.
- Import new component versions using a single agent.
- Import new component versions using any agent with the specified tag.

iii. In the component go to the process and click on the create process



iv. Go to process created and edit the process.



v. Click on edit and make the proper flow chart of the whole process.

The screenshot displays the IBM UrbanCode Deploy web interface. At the top, the navigation bar includes 'Dashboard', 'Components', 'Applications', 'Configuration', 'Processes', 'Resources', 'Calendar', 'Work Items', 'Reports', and 'Settings'. The breadcrumb trail indicates the current location: 'Home > Components > new_seema_component > Processes > Process: vehicle_store'. The main heading is 'Process: vehicle_store', with a sub-heading 'Version 3 of 3'. Below this, there are tabs for 'Design', 'Configuration', and 'Changes', with 'Design' being the active tab. On the left side, there is a 'Step Palette' with a search bar and a list of steps: 'Finish', 'Clipboard', 'Utility Steps', 'Application Server', 'IS', 'IIS MSDeploy', 'Delete', 'msdeploy', 'Recycle Application', 'Start Application', 'Stop Application', and 'Synchronize'. The main workspace shows a flowchart with the following steps: 1. 'Start' (grey box). 2. 'Download Artifacts' (purple box, 'Download Artifacts (v. 23)'). 3. 'Update XML File with XPath' (purple box, 'Update XML File with XPath (v. 40)'). 4. 'msdeploy' (purple box, 'msdeploy (v. 1)'). 5. 'Finish' (grey box). The flowchart is connected by arrows, and each step has a green checkmark icon, indicating that the process is complete or successful.

vi. Now go to the Application and create a new application

The screenshot displays the IBM UrbanCode Deploy web interface. The top navigation bar includes 'Dashboard', 'Components', 'Applications', 'Configuration', 'Processes', 'Resources', 'Calendar', 'Work Items', 'Reports', and 'Settings'. The user is logged in as 'admin'. The main content area shows a list of applications with columns for Name, Description, Created, and By. A 'Create Application' dialog box is open in the foreground, containing the following fields and options:

- Name ***: new_vehicle_store
- Description**: (empty text box)
- Teams**: + (add team button)
- Notification Scheme**: None (dropdown menu)
- Enforce Complete Snapshots**:

At the bottom of the dialog are 'Save' and 'Cancel' buttons. The background application list includes entries like 'Application_A_APP', 'Application_B_APP', 'ASHOKE_TEST', 'Employee_Sachin_DB', 'FirstApplication', 'Hello_Seema', 'hello_world', 'helloWorld', 'Jpetstore', and 'MyApp'.

vii. In the application click on create Environment to create a new environment.

The screenshot shows the IBM UrbanCode Deploy interface. At the top, the navigation bar includes 'Dashboard', 'Components', 'Applications', 'Configuration', 'Processes', 'Resources', 'Calendar', 'Work Items', 'Reports', and 'Settings'. The breadcrumb trail is 'Home > Applications > new_vehicle_store > Environments > Environment: vehicle_store'. The main heading is 'Environment: vehicle_store for new_vehicle_store'. Below this, there are tabs for 'Resources', 'History', 'Calendar', 'Configuration', and 'Changes'. The 'Basic Settings' section is active, showing the following configuration:

- Name:** vehicle_store
- Description:** (empty field)
- Teams:** +
- Require Approvals:**
- Exempt Processes:** None
- Lock Snapshots:**
- Color:** A color palette grid with a green circle highlighting the 'Mountain Meadow (Green)' color.
- Inherit Cleanup Settings:**

At the bottom of the settings panel are 'Save' and 'Cancel' buttons.

- viii. Go to the environment created and in that go to the Component then add your component to the created environment.

The screenshot displays the IBM UrbanCode Deploy web interface. At the top, the header shows 'IBM UrbanCode Deploy' and the user 'admin'. A navigation menu includes 'Dashboard', 'Components', 'Applications', 'Configuration', 'Processes', 'Resources', 'Calendar', 'Work Items', 'Reports', and 'Settings'. The breadcrumb trail is 'Home > Applications > new_vehicle_store'. The main heading is 'Application: new_vehicle_store'. Below this, it shows 'Created By: admin' and 'Created On: 6/22/2016, 5:21 PM'. A secondary navigation menu includes 'Environments', 'History', 'Configuration', 'Components', 'Blueprints', 'Snapshots', 'Processes', 'Calendar', and 'Changes'. The 'Add Component' button in the 'Components' section is circled in red, with an arrow pointing to a modal dialog box titled 'Add a Component'. The dialog box contains a dropdown menu labeled 'Select a Component *' and 'Save' and 'Cancel' buttons. In the background, a table lists components, with 'new_seema_component' visible. The table has columns for 'Component', 'Description', and 'Actions'. Below the table, it says '1 record - Refresh Print' and 'Rows 10'.

- ix. In the component Create a new process → in that process create the flow chart to install the component(Drag the install component and connect with start and finish).

The screenshot displays the IBM UrbanCode Deploy web interface. At the top, the navigation bar includes 'Dashboard', 'Components', 'Applications', 'Configuration', 'Processes', 'Resources', 'Calendar', 'Work Items', 'Reports', and 'Settings'. The breadcrumb trail shows 'Home > Applications > new_vehicle_store > Processes > Process: vehicle_store'. The main heading is 'Process: vehicle_store'. Below this, the 'Version' section shows '2 of 2'. The 'Design' tab is active, with sub-tabs for 'Configuration', 'Inventory Changes', and 'Changes'. On the left, the 'Step Palette' is visible, listing various actions such as 'Finish', 'Install Component...', 'Install Multiple Components...', 'Uninstall Component...', 'Rollback Component...', 'Run Process for Each Version...', 'Run Process for Each Affected Resource...', 'Apply Configuration', 'Utility Steps', 'Configuration Discovery', and 'new_seema_component'. The 'Install Component...' step is circled in red, with a red arrow pointing to its position in the process flow. The flowchart on the right shows a 'Start' node connected to an 'Install new_seema_component' node, which is then connected to a 'Finish' node. The 'Install new_seema_component' node is highlighted in purple and has a green checkmark icon next to it.

- x. Then add the base resource to the application.

In the Applications **map the resource to the agent** and then map the agent to the component for this

Go to the Applications → go to the environment created by you → click on the Add Base Resources and select your resource.

The screenshot shows the IBM UrbanCode Deploy web interface. The top navigation bar includes 'Dashboard', 'Components', 'Applications', 'Configuration', 'Processes', 'Resources', 'Calendar', 'Work Items', 'Reports', and 'Settings'. The breadcrumb trail is 'Home > Applications > new_vehicle_store > Environments > Environment: vehicle_store'. The main heading is 'Environment: vehicle_store for new_vehicle_store'. Below this, there are tabs for 'Resources', 'History', 'Calendar', 'Configuration', and 'Changes'. A message box says 'No Desired Inventory'. Below that, there are buttons for 'Add Base Resources', 'Select All...', 'Actions...', and 'Show'. On the right, there are links for 'Expand All' and 'Collapse All'. A table with columns 'Name', 'Inventory', 'Status', and 'Description' is displayed. The table contains three rows: a header row with input fields for 'Resource Name', 'Tags', and 'Description'; a row for a folder 'new_vehicle_store'; and a row for an agent 'seema (View Agent)' with a status of 'Online'. Below the table, there are links for 'Refresh' and 'Print'.

<input type="checkbox"/>	Name	Inventory	Status	Description
	<input type="text" value="Resource Name"/> <input type="text" value="Tags"/>	<input type="text"/>		<input type="text"/>
	▼ new_vehicle_store			
:: <input type="checkbox"/>	▼ seema (View Agent)		Online	
:: <input type="checkbox"/>	new_seema_component			

- xi. Then go to the resources → go to your agent → go to configuration, Add the required property in the property you have to give the name of the property. (like, here `IIS-MSD/commandPath`) and the value in value you have to specify the physical location of the Microsoft Web deploy V3.

IBM UrbanCode Deploy dW admin

Dashboard | Components | Applications | Configuration | Processes | Resources | Calendar | Work Items | Reports | Settings

Home > Resources > seema

Resource: seema

Agent: seema
Parent Resource: /new_vehicle_store

Main | Inventory | History | **Configuration** | Changes

Basic Settings | **Resource Properties**

Resource Properties

Version 2 of 2

[Add Property](#) [Batch Edit](#)

Name	Value	Description	Actions
<input type="text"/>	<input type="text"/>		
IIS-MSD/commandPath	C:\Program Files\IIS\Microsoft Web Deploy V3\		Edit Delete

1 record - [Refresh](#) [Print](#) Rows: 10

xii. Then go to the application → click on the component tab → Go to your component.

IBM UrbanCode Deploy dW admin

Dashboard | Components | Applications | Configuration | Processes | Resources | Calendar | Work Items | Reports | Settings

Home > Applications > new_vehicle_store

Application: new_vehicle_store

Created By: admin
Created On: 6/22/2016, 5:21 PM

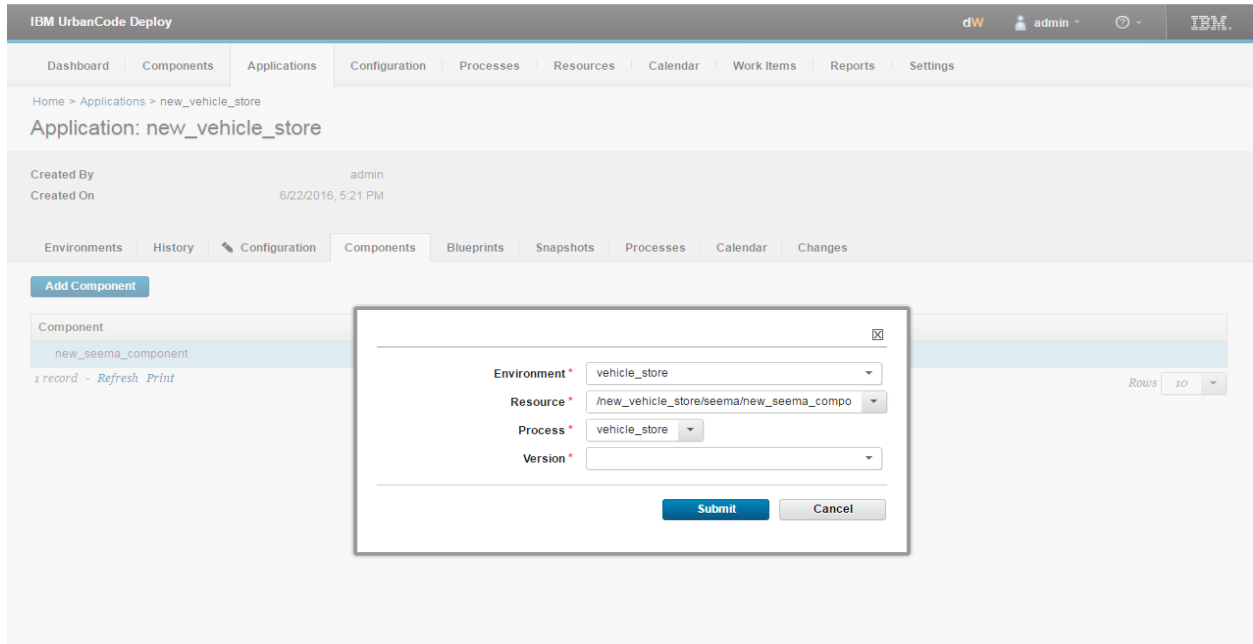
Environments | History | **Configuration** | **Components** | Blueprints | Snapshots | Processes | Calendar | Changes

[Add Component](#)

Component	Description	Actions
new_seema_component		Run Process Remove

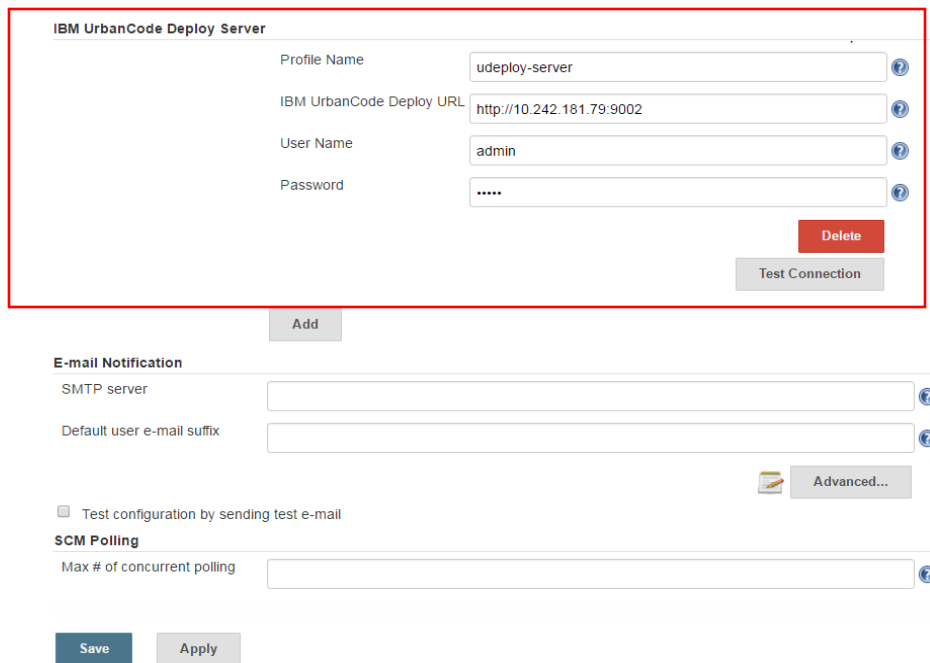
1 record - [Refresh](#) [Print](#) Rows: 10

xiii. Click on the **RUN PROCESS**, it will show this window. Here we need version this version we need to generate from the Jenkins.



xiv. Jenkins Configuration with the UDeploy server

Firstly, download the IBM UrbanCode Deploy plugin → go to Manage Jenkins → go to Configure System then configure it as:



Xv. Create a new job → add Publish artifacts to IBM UrbanCode Deploy in post build step with in Jenkins.

Build

Add build step ▾

Post-build Actions

Publish artifacts to IBM UrbanCode Deploy ⓘ

IBM UrbanCode Deploy Server ⓘ

Use Another User

User Name ⓘ

User Password ⓘ

Component ⓘ

Base Artifact Directory ⓘ

Directory Offset ⓘ

Version ⓘ

Version Description ⓘ

Version Properties

Include ⓘ

Exclude

Skip publishing (e.g. temporarily)

Deploy ⓘ

Deploy Application ⓘ

Deploy Environment ⓘ

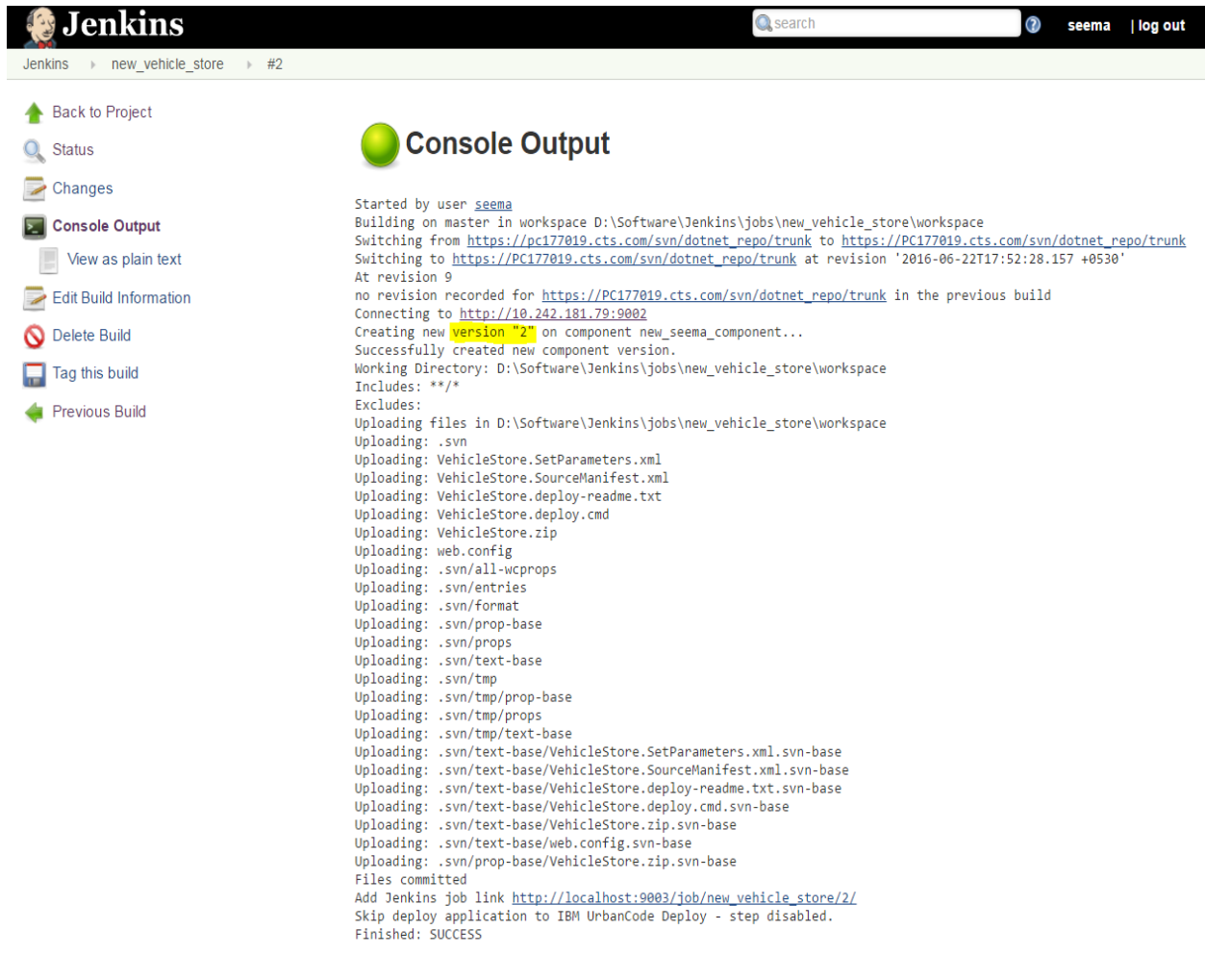
Deploy Process ⓘ

Delete

Add new build action ▾

Save **Apply**

Once it is done build the job it will generate the build number which will be the version required in the UDeploy.



The screenshot shows the Jenkins web interface. At the top, there's a navigation bar with the Jenkins logo, a search box, and the user name 'seema' with a 'log out' link. Below the navigation bar, the breadcrumb trail reads 'Jenkins > new_vehicle_store > #2'. On the left side, there's a sidebar with several links: 'Back to Project', 'Status', 'Changes', 'Console Output' (which is highlighted), 'View as plain text', 'Edit Build Information', 'Delete Build', 'Tag this build', and 'Previous Build'. The main content area is titled 'Console Output' and displays the following text:

```
Started by user seema
Building on master in workspace D:\Software\Jenkins\jobs\new_vehicle_store\workspace
Switching from https://pc177019.cts.com/svn/dotnet\_repo/trunk to https://PC177019.cts.com/svn/dotnet\_repo/trunk
Switching to https://PC177019.cts.com/svn/dotnet\_repo/trunk at revision '2016-06-22T17:52:28.157 +0530'
At revision 9
no revision recorded for https://PC177019.cts.com/svn/dotnet\_repo/trunk in the previous build
Connecting to http://10.242.181.79:9002
Creating new version "2" on component new_seema_component...
Successfully created new component version.
Working Directory: D:\Software\Jenkins\jobs\new_vehicle_store\workspace
Includes: **/*
Excludes:
Uploading files in D:\Software\Jenkins\jobs\new_vehicle_store\workspace
Uploading: .svn
Uploading: VehicleStore.SetParameters.xml
Uploading: VehicleStore.SourceManifest.xml
Uploading: VehicleStore.deploy-readme.txt
Uploading: VehicleStore.deploy.cmd
Uploading: VehicleStore.zip
Uploading: web.config
Uploading: .svn/all-wcprops
Uploading: .svn/entries
Uploading: .svn/format
Uploading: .svn/prop-base
Uploading: .svn/props
Uploading: .svn/text-base
Uploading: .svn/tmp
Uploading: .svn/tmp/prop-base
Uploading: .svn/tmp/props
Uploading: .svn/tmp/text-base
Uploading: .svn/text-base/VehicleStore.SetParameters.xml.svn-base
Uploading: .svn/text-base/VehicleStore.SourceManifest.xml.svn-base
Uploading: .svn/text-base/VehicleStore.deploy-readme.txt.svn-base
Uploading: .svn/text-base/VehicleStore.deploy.cmd.svn-base
Uploading: .svn/text-base/VehicleStore.zip.svn-base
Uploading: .svn/text-base/web.config.svn-base
Uploading: .svn/prop-base/VehicleStore.zip.svn-base
Files committed
Add Jenkins job link http://localhost:9003/job/new\_vehicle\_store/2/
Skip deploy application to IBM UrbanCode Deploy - step disabled.
Finished: SUCCESS
```

xvi. Then click on Run Process in UDeploy and select your environment and process and **select the version** from the dropdown → Click on Submit → it will deploy the application in the IIS Server.

IBM UrbanCode Deploy dW admin

Dashboard | Components | Applications | Configuration | Processes | Resources | Calendar | Work Items | Reports | Settings

Home > Applications > new_vehicle_store

Application: new_vehicle_store

Created By: admin
Created On: 6/22/2016, 5:21 PM

Environments | History | Configuration | Components | Blueprints | Snapshots | Processes | Calendar | Changes

[Add Component](#)

Component
new_seema_component

1 record - Refresh Print Rows: 10

Environment*

Resource*

Process*

Version*

[Submit](#) [Cancel](#)

Click on Submit and RUN the process again by clicking on the RUN PROCESS.

IBM UrbanCode Deploy dW admin

Dashboard | Components | Applications | Configuration | Processes | Resources | Calendar | Work Items | Reports | Settings

Home > Components > new_seema_component > Process Request on new_seema_component

Deployment of Component: new_seema_component

Process: vehicle_store (Version 3)
Version: 3
Resource: new_seema_component
Agent: seema
Date: 6/24/2016, 12:29 PM
Requested By: admin

[Log](#) [Properties](#)

Execution Dock timeline at top

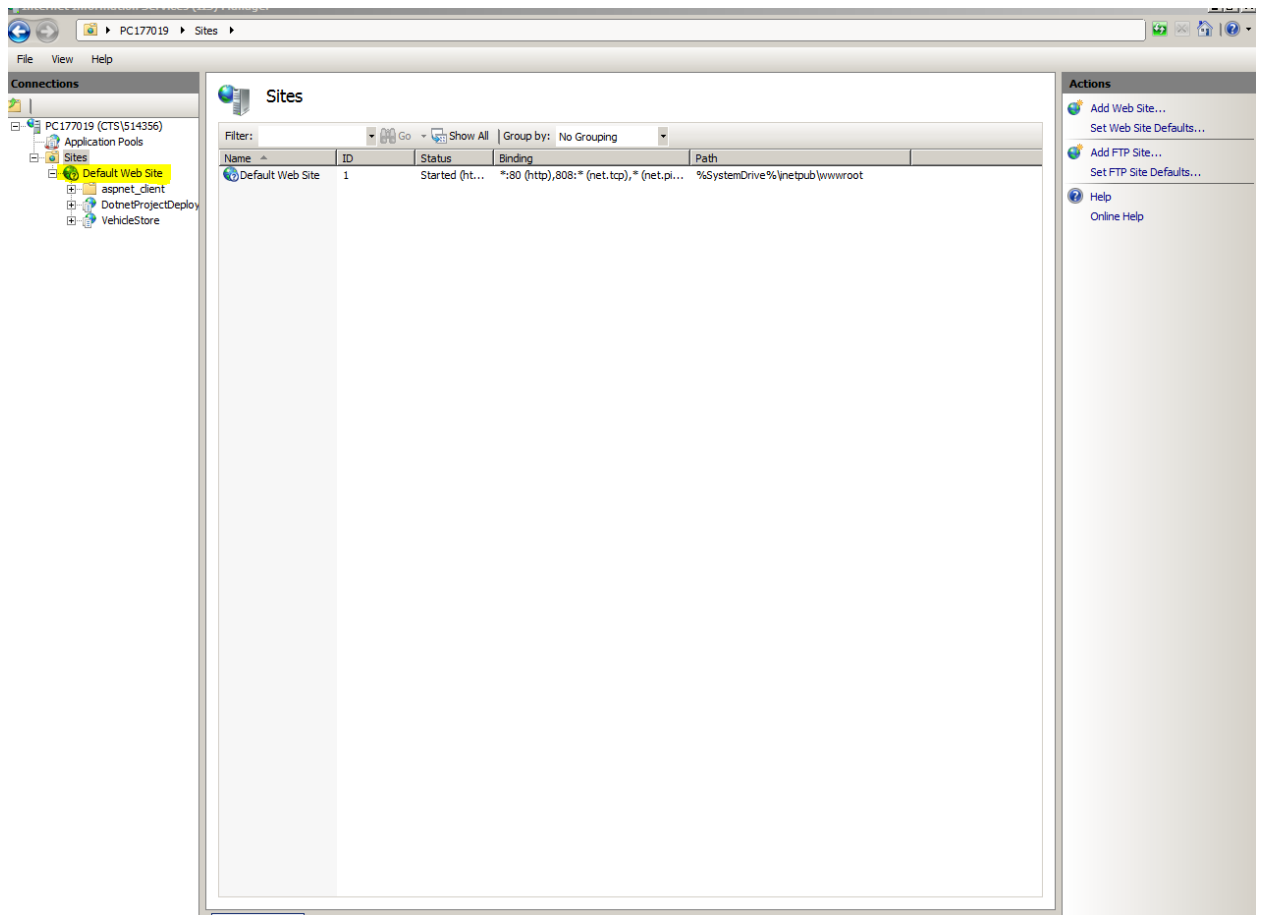
Start: 12:29:03 PM | Progress: 3/3 | Status: Success | Duration: 0:00:21 | End: 12:29:24 PM

[Download All Logs](#) Expand All Collapse All

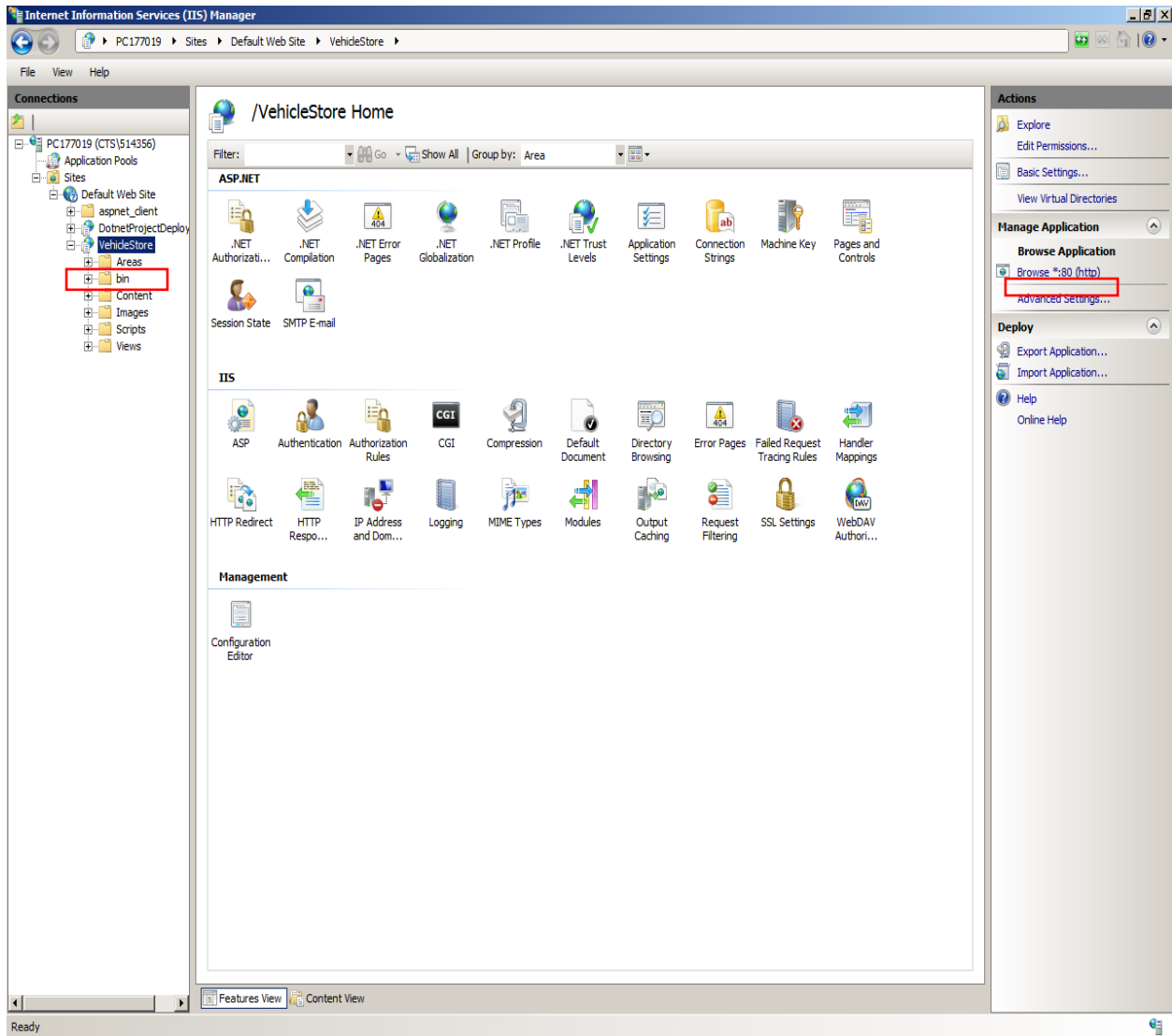
Step	Progress	Start Time	Duration	Status
1. Download Artifacts	✓	12:29:03 PM	0:00:08	Success
2. Update XML File with XPath	✓	12:29:12 PM	0:00:03	Success
3. msdeploy	✓	12:29:15 PM	0:00:08	Success
Total Execution	3/3	12:29:03 PM	0:00:21	Success

Deployment In IIS Server

1. Open Internet Information Service (IIS) Manager → click on the sites → click on the default web site.
2. The deployed website will be in the Default web Site.



- Expand default web site, the application will be visible over here. Now, to view the web page click on the deployed project name and then Browse *:80(http) on the right side of IIS server.



WEB DEPLOYED

The web page will be displayed like this(Your web app will be deployed over here):

ASP.NET Web API Home API

Welcome to ASP.NET Web API! Modify the code in this template to jump-start your ASP.NET Web API development.

ASP.NET Web API allows you to expose your applications, data and services to the web directly over HTTP.

To learn more about ASP.NET Web API visit the [ASP.NET Web API Website](#). The page features [videos](#), [tutorials](#), and [samples](#) to help you get the most from ASP.NET Web API. If you have any questions about ASP.NET Web API, visit [our forums](#).

We suggest the following steps:

- 1 Getting Started**
ASP.NET Web API is a framework that makes it easy to build HTTP services that reach a broad range of clients, including browsers and mobile devices. ASP.NET Web API is an ideal platform for building RESTful applications on the .NET Framework. [Learn more...](#)
- 2 Add NuGet packages and jump-start your coding**
NuGet makes it easy to install and update free libraries and tools. [Learn more...](#)
- 3 Find Web Hosting**
You can easily find a web hosting company that offers the right mix of features and price for your applications. [Learn more...](#)

Click on the API it will redirect you automatically to the ASP.NET Web API Page.

ASP.NET Web API Help Page

Introduction

Provide a general description of your APIs here.

Vehicles

API	Description
GET vehicles	No documentation available.
GET vehicles/{id}	No documentation available.
GET vehicles/{make}	No documentation available.
POST vehicles	No documentation available.
PUT vehicles?id={id}	No documentation available.
DELETE vehicles?id={id}	No documentation available.