**Jenkins Overview:**

Jenkins provides continuous integration services for software development. It is a server-based system running in a servlet container such as Apache Tomcat. Application code can be build and released using Jenkins.

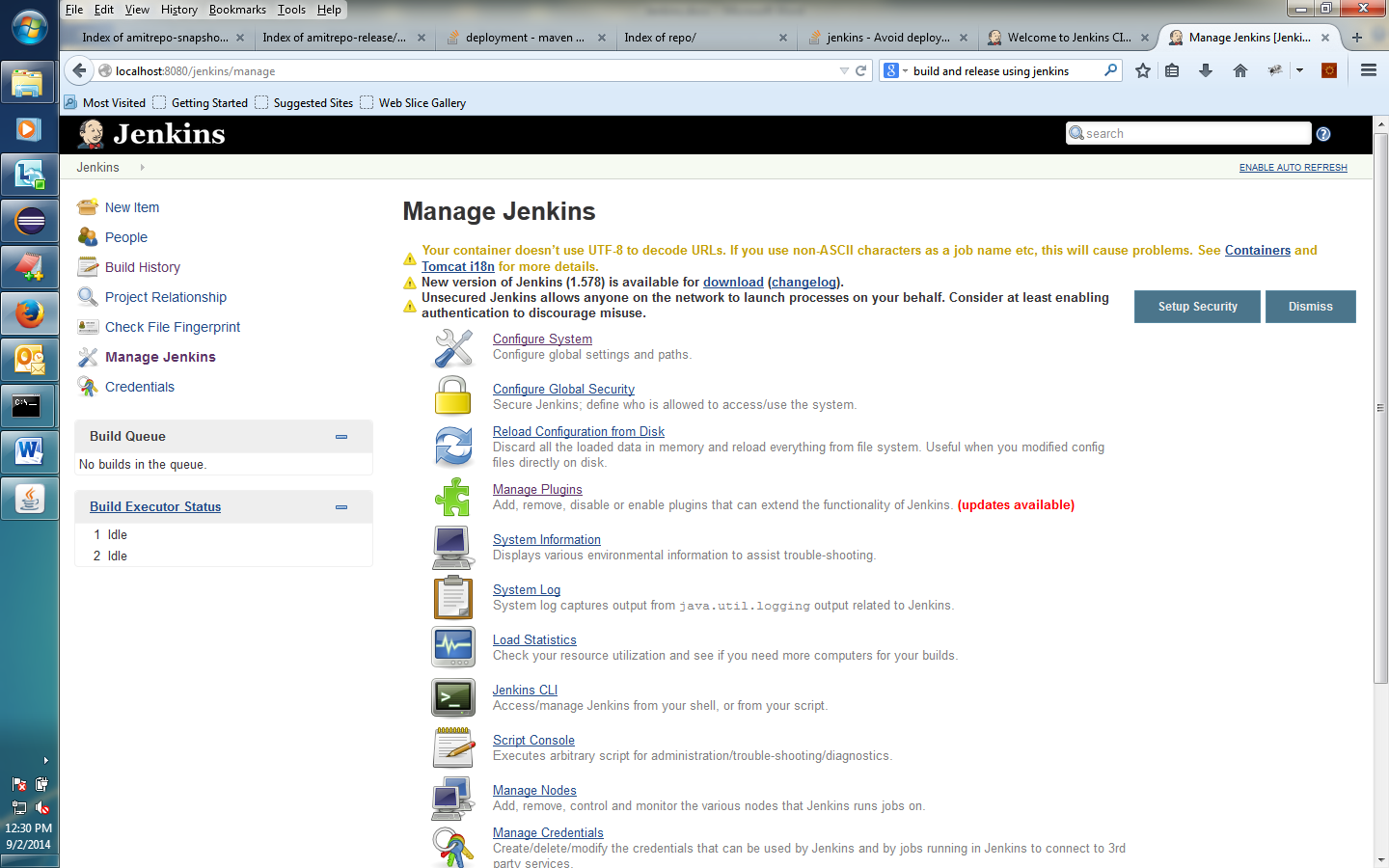
**How to install Jenkins:**

Download the Jenkins.war from [**http://jenkins-ci.org/**](http://jenkins-ci.org/)and deploy it on a tomcat web container. After deployment you can access Jenkins UI through browser using the URL [**http://localhost:8080/jenkins/**](http://localhost:8080/jenkins/).

**Plugins required for build and release:**

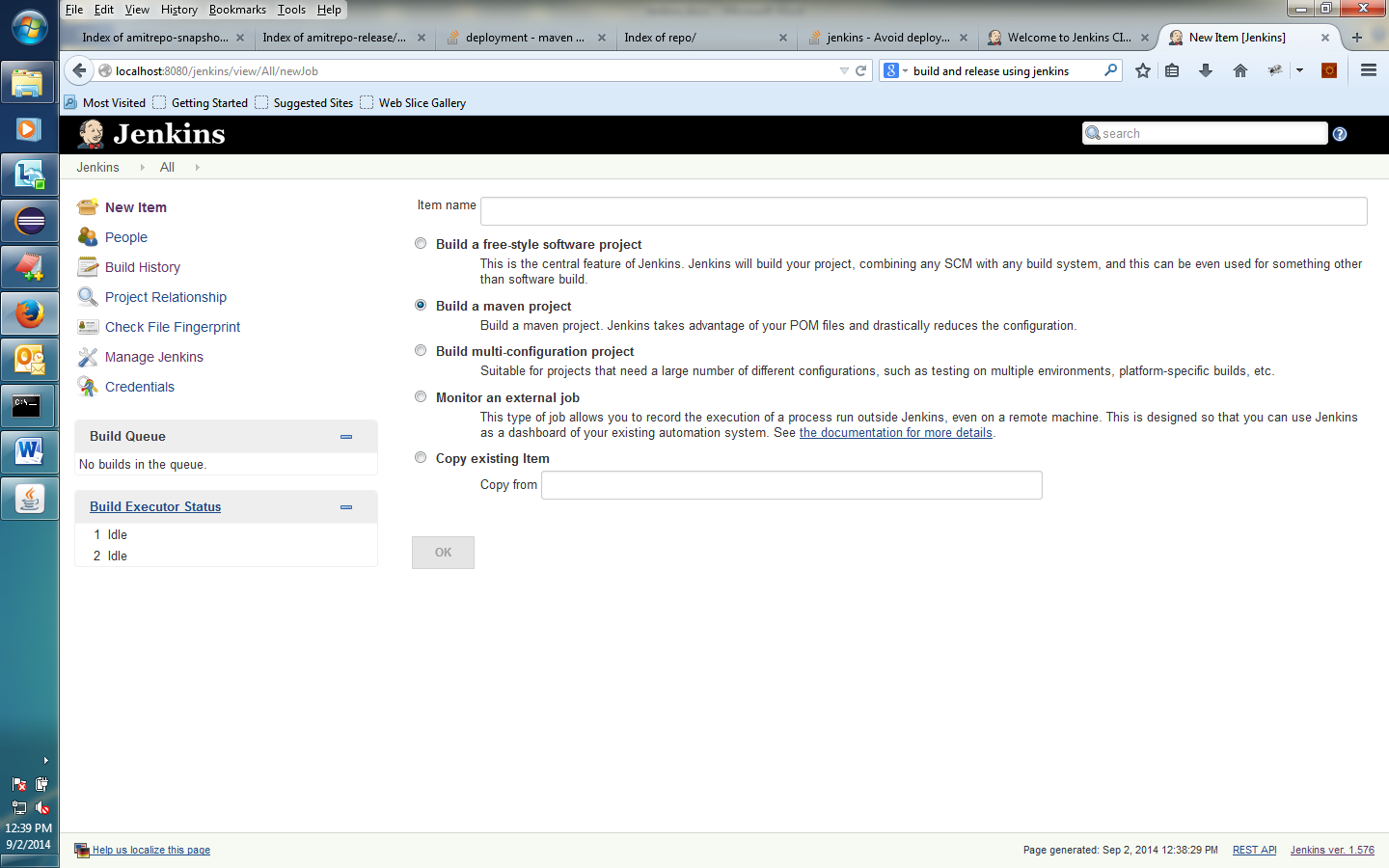
Plugins can be accessed through **[Manage Jenkins](http://localhost:8080/jenkins/manage)-->**[**Manage Plugins**](http://localhost:8080/jenkins/pluginManager) on the webui. The below plugins are required to build and release plugin and deploy the application war or ear to any servlet container

* [**Subversion Plug-in**](http://wiki.jenkins-ci.org/display/JENKINS/Subversion+Plugin) **:** This plugin is required to configure Jenkins to get the source code from Subversion.
* [**Maven Release Plug-in Plug-in**](http://wiki.jenkins-ci.org/display/JENKINS/M2+Release+Plugin) **:** This plugin is required to release the application code.
* [**Deploy to container Plugin**](http://wiki.jenkins-ci.org/display/JENKINS/Deploy+Plugin) **:** This plugin is required to deploy the build code to any tomcat/Glassfish servlet container.



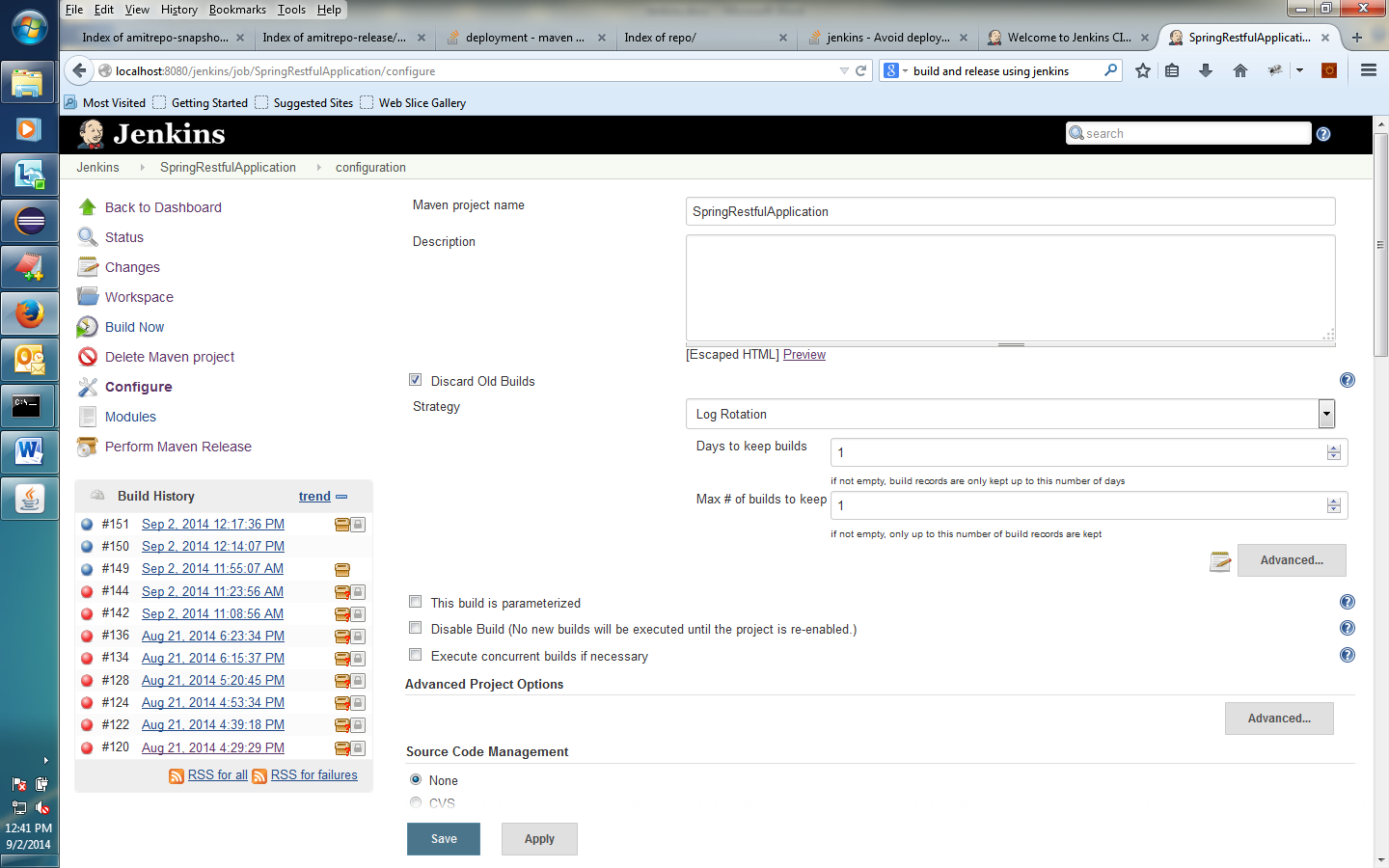
**Create Jobs for build and release:**

Add a Build maven project as shown below.

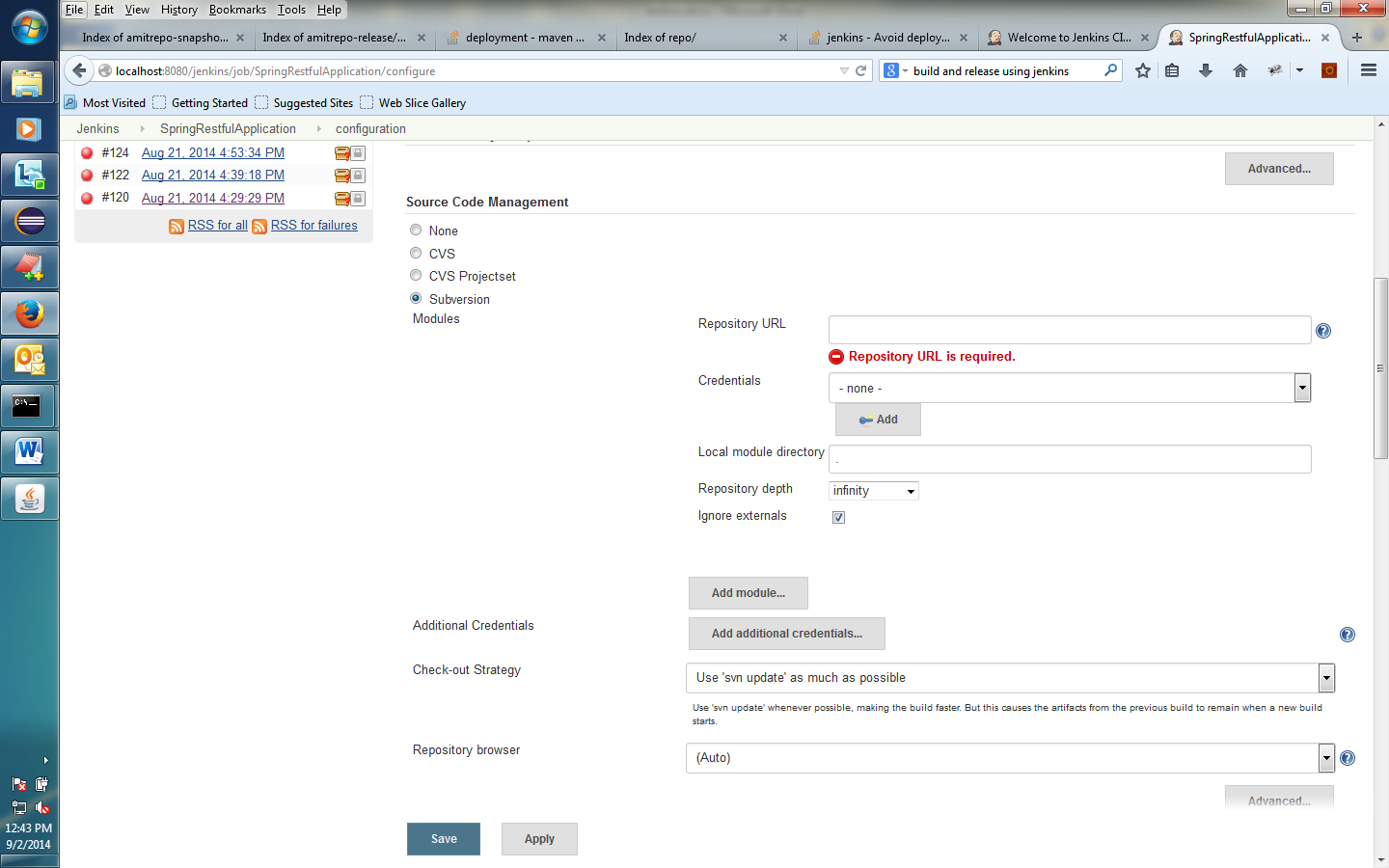


**Configure the job**

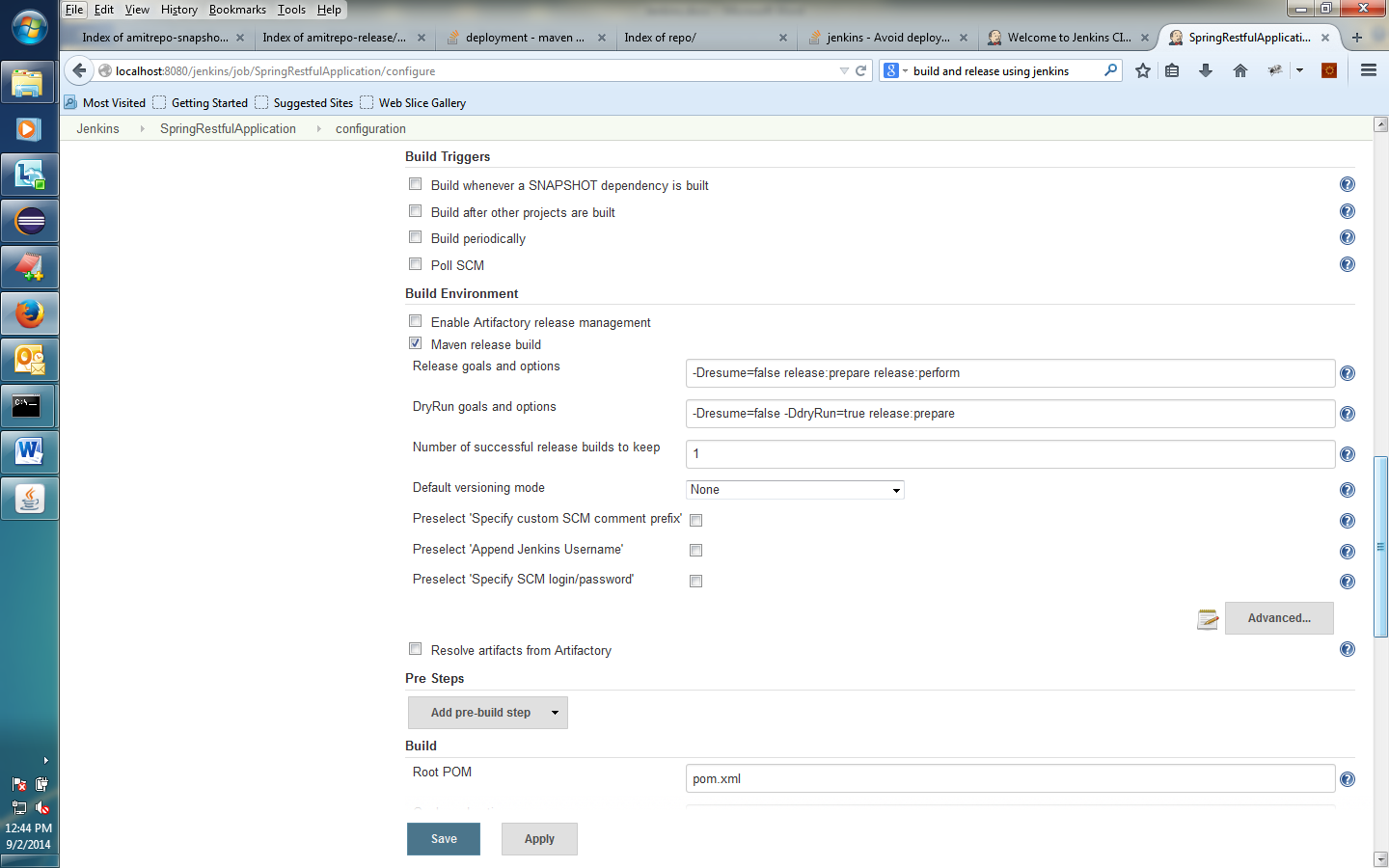
* Add the maven project name and configure the log rotation setting as shown in below example screenshot.



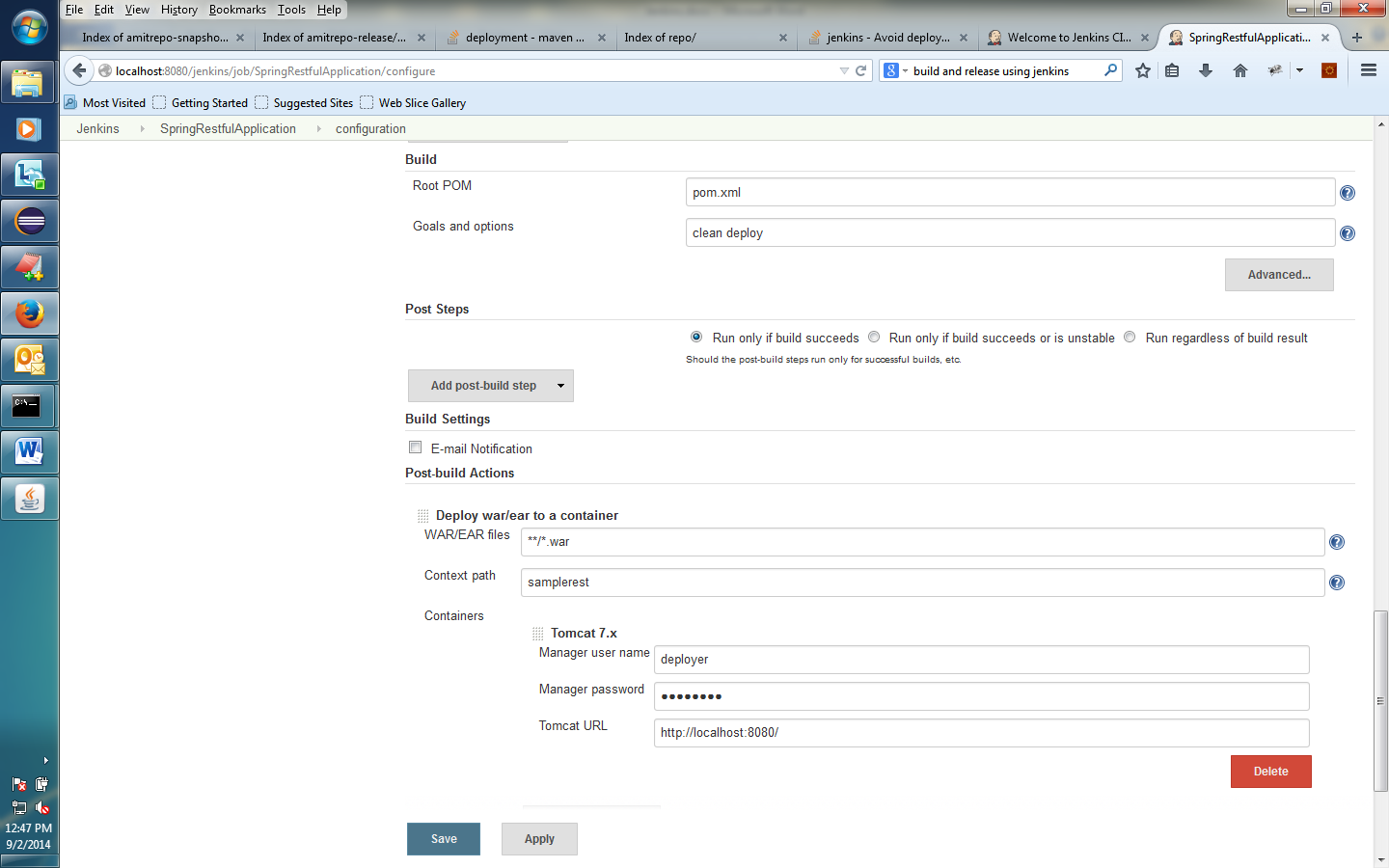
* Configure the subversion URL for the source code as shown below



* Configure Build Environment as shown below



* Configure Build and Post-Build Actions as shown below. This Post-Build actions settings will deploy the build war or ear to the servlet container.

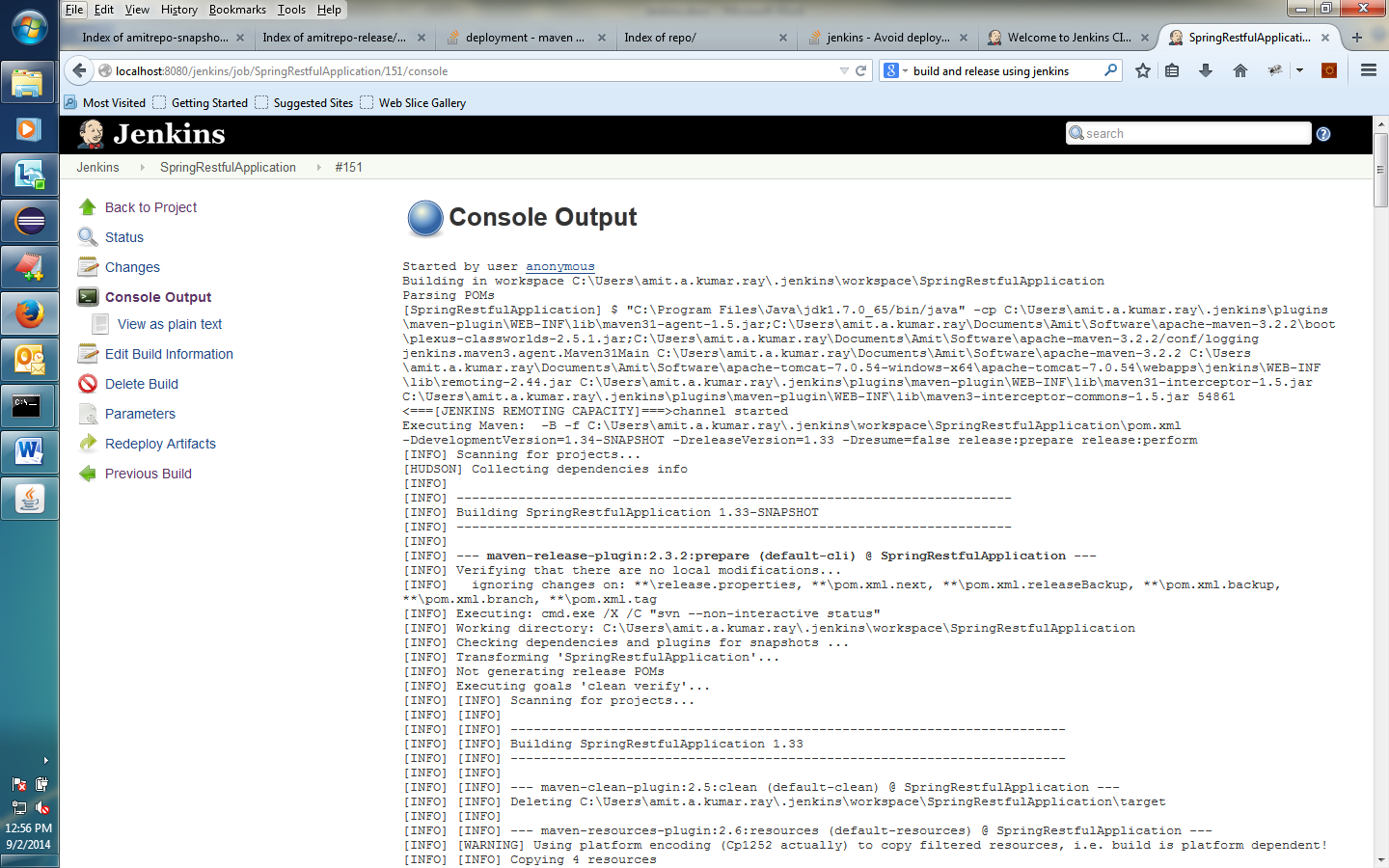


**Perform build and release:**

Click on the [**Build Now**](http://localhost:8080/jenkins/job/SpringRestfulApplication/build?delay=0sec) link from left to start the build process. You can see the build progress in the build History section. Also you can see the complete console output or logs by clicking on respective build in the build History section.

Once build is completed successfully click on the **Perform Maven Release** link from left to start the release process. You can see the release progress in the build History section. Also you can see the complete console output or logs by clicking on respective release in the build History section.

* Sample Console Output



**Deploy Artifacts to remote repository:**

After build and release the artifacts can be deployed to remote repository. The below repository details needs to mentioned on the pom.xml. Its always good practice to put these details to the parent pom.xml. Also if there is any uid/pwd required for the remote repository, then that can be put on the maven settings.xml.

* **POM.XML**

<distributionManagement>

<repository>

<id>releaserepo</id>

<name>repo for release</name>

<url>http://localhost:8081/artifactory/amitrepo-release/</url>

</repository>

<snapshotRepository>

<id>snapshotrepo</id>

<name>repo for snapshot</name>

<url>http://localhost:8081/artifactory/amitrepo-snapshot/</url>

</snapshotRepository>

</distributionManagement>

* **Maven settings.xml**

<server>

<id>releaserepo</id>

<username>admin</username>

<password>password</password>

</server>

<server>

<id>snapshotrepo</id>

<username>admin</username>

<password>password</password>

</server>

More Details about Jenkins are available in **https://wiki.jenkins-ci.org/display/JENKINS/Use+Jenkins**