C3PO Build/Deploy Instructions for c3po webapp

Contents

[Software Installations (required) 1](#_Toc431479391)

[Maven 1](#_Toc431479392)

[Tomcat 1](#_Toc431479393)

[Database 2](#_Toc431479394)

[Web Application (clinical3PO) 2](#_Toc431479395)

[Data Services 2](#_Toc431479396)

**Before the actual steps are listed, it is assumed**

1. Hortonworks hadoop is installed (relevant version).
2. The login used to copy data to hadoop (hdfs) and the login used to start the apache tomcat is same.

Software Installations (required)**.**

1. Install java/jdk, set appropriate variables (JAVA\_HOME, PATH), 1.6 is the least version that is required.
2. Set PATH to have hadoop binaries (so that hadoop related commands can be executed).
3. Install apache maven, for installation instructions look at the MAVEN section.
4. Install apache tomcat, for installation instructions look at the Tomcat section.
5. Requires Firefox browser.
6. CLASSPATH variable should NOT be set (at least anything related to c3po project).

Maven

1. Download apache maven (http://maven.apache.org/download.cgi) , 3.0 is the least version that is required.
2. Extract the binary to a folder, update the PATH variable to include bin directory from the extracted folder. Create an environment variable M2\_HOME which points to the extracted folder.

Tomcat

1. Download the tomcat installer (<http://www.bizdirusa.com/mirrors/apache/tomcat/tomcat-7/v7.0.53/bin/apache-tomcat-7.0.53.tar.gz>)
2. Extract to a folder, update the PATH variable to include bin directory from the extracted folder.
3. Create an environment variable CATALINA\_HOME which points to the extracted folder.
4. catalina.sh start will start the server, catalina.sh stop will stop the server.

Database

1. Create the database.
2. Execute the scripts located in utilities\db-objects. Read Readme.txt to know the order of execution of the \*.sql scripts.
3. The same database entries (username,password…should be entered in settings.xml).

Web Application (clinical3PO)

1. Download the source from git hub
2. Copy the profile segment from README.txt which is under utilities/build/maven (project source) to settings.xml which is present under conf (apache maven)
3. Compile and Build the source using mvn clean install -DskipTests –Penv-properties (at the project root folder). The clinical3PO war will be created under app/target
4. Copy the clinical3PO.war (under target) to webapps (under apache tomcat).
5. Start the catalina, catalina.sh start (it is assumed that bin directory of apache tomcat is in PATH).
6. Check logs (under apache tomcat) for any issues.
7. Application can be accessed through http://<targetmachine>:8080/clinical3PO.
8. Now click existing searches and see if page is loading properly (without data it will load)
9. Please do additional steps if hadoop jobs have to be created, (below tasks).

Data Services

1. When any patient id is given for search, application in turn calls a shell script that executes a hadoop job, this project deals with the source code pertaining to this activity.
2. Change directory to data-services, execute mvn clean assembly:assembly -Dbinary=true -DskipTests -Penv-properties, a binary with a name clinical3PO-data-services-1.0.0-SNAPSHOT-bin.tar is created under target directory.
3. Extract this tar archive under a specified directory, whatever the directory that was given against < clinical3PO.hadoop.shellscripts.dir> in settings.xml, for example below we are saying the relevant scripts (hadoop related) and data services are stored under /home/c3po/c3po-hadoop-scripts.

<!-- Directory where shell scripts related to hadoop search are stored -->

< clinical3PO.hadoop.shellscripts.dir >/home/ c3po / clinical3PO -hadoop-scripts</ clinical3PO.hadoop.shellscripts.dir >