

# Standard Tool Chain Manual Setup Steps

## 1. Steps for Integration of all the tools

Details:

### i. Jenkins-UCV Integration

- a) Login to Jenkins tool in browser URL: `http://IP-address:9292`
- b) Enter the username and password
- c) When first time user logins to Jenkins, he may get the "Install Suggested Plugin", when user click on this link, it will install all the necessary plugins.
- d) Jenkins Url will be auto populated, click on "Save and Finish".
- e) Now Restart the Jenkins tool.
- f) The above two steps does not appear all the times.If it doesn't appear, just go ahead with the next steps as follows.
- g) Click on Manage Jenkins.
- h) Click on Manage Plugins.
- i) Under "Available", search for "Urban Code Velocity" plugin.
- j) Click on "Install without restart".
- k) Now check the checkbox for "Restart Jenkins when no jobs are running".
- l) Click on "Manage Jenkins".
- m) Click on "Configure system".
- n) Under Urban Code Velocity section enter the following details:  
Integration Id  
Integration Token  
Velocity Base Urls  
Jenkins Credentials→this need to be added before, just select the credentials from the list.
- o) And click on Save and Restart the Jenkins after UCV plugin installation.

Steps to get Integration Id and Integration token

- a) Login to UCV tool.
- b) Click on Settings button (Gear symbol at RHS).
- c) Click on Integrations, add Integrations and then select Jenkins option.
- d) Give the name for the Integration and click on save button.
- e) Copy paste the data in Jenkins Config section for UCV.
- f) Will get the Integration Id and Integration token.

### ii. UCD-UCV Integration

- a) Login to UCD tool
- b) Click on "Settings".
- c) Under "Security" section click on "token".
- d) Click on Create Token.
- e) Select the user as admin
- f) Give the data like Expiration date and Expiration time (the time for which you want the pipeline to be active for use)
- g) And click on "save".
- h) You will get the Newly created token.
- i) Copy the token and save it notepad.

- j) Now Login to UCV tool.
- k) Click on Setting, then go to Integrations and then Add Integrations.
- l) Select Urban Code Deploy.
- m) Give the name, UCD Server Url and the UCD access token (which you saved in notepad).
- n) Click on save. Integration should be added, and the status should be successful.

## **2. Steps to Integrate Jenkins with UCD**

Details:-

- i. *Steps to Configure Urban Code Deploy with Jenkins*
  - a. Login to UCD tool with valid username and password.
  - b. Click on "Settings" tab, under that navigate to "System settings".
  - c. Under General Settings tab, under Agent for Version import, select "ucda-docker" and click on "save" button.
  - d. Now go to the Resources tab, and click on "Create Top Level Group",
  - e. In the "Create resource" pop-up, give the Resource name, for e.g. JpetStore-resource and click in save button.
  - f. Hover over the Resource Name, i.e. JpetStore, user can see the "Actions" link.
  - g. Under the Actions link, select the "Add Agent".
  - h. Under the "Agent" field, select "ucda-docker" and click on save button.
  - i. Click on "Applications" tab.
  - j. Click on "Create Application", give the Application name i.e. JpetStore-App and click on "Create" button. Now the application will be successfully created.
  - k. Click on the Application name, i.e. JpetStore-App.
  - l. Now click on "Create Environment", it will open the "Create Environment" pop-up.
  - m. Give the Environment name for e.g. "Test" and click on "save" button.
  - n. Click on the "Environment name" which you just now created, i.e. "Test".
  - o. Under Environment name, click on "Add Base Resources".
  - p. Select the Resource which you created earlier, i.e. "Jpetstore-resource".
  - q. Now login to Jenkins tool and click on "Build Now" for the Pipeline "jpetstore".
  - r. Jenkins tool will create the component "Jenkins-jpet-component" with latest version of the build.
- ii. *Steps for Creation of process in UCD: -*

Precondition: Download the "Tomcat Apache UCD Plugin" and install in UCD tool.

- (
- 1. Login to UCD tool.
- 2. Navigate to the "Settings" tab.
- 3. Under Automation section, link on "Automation Plugins".
- 4. Click on "Load plugin".
- 5. Upload Tomcat 7.868953.zip.
- Link: <https://developer.ibm.com/urbancode/plugin/tomcat-ibmucd/>
- )

- a) Login to the UCD tool with valid username and password.
- b) Click on "Components" tab.
- c) Click on the "Component name" for eg. Jenkin-jpet-component.
- d) Click on the "Processes" tab.

- e) Click on "Create Process".
- f) In "Create Process" pop-up, give the Process name and click on "Save" button.
- g) Click on the "Design" tab and design the process for the component.
- h) Click on "Download Artifacts" and drag and drop after the "Start" step. This can be found under Repositories→Download Artefacts
- i) Click on "Start Tomcat" and drag and drop after "Download Artifacts" step. This can be found under Application Server→Start Tomcat.
- j) Click on Edit symbol, for "Start Tomcat".
- k) Look for the "launcher" text area and give the path. (This is the path Apache Tomcat Server is mounted.) and click on "save" button.
- l) Click on "Copy Directory" and drag and drop after the "Start Tomcat" step. This can be found under Utilities→File Utils→Copy Directory.
- m) Click on Edit symbol for "Copy Directory" and feed the following data and then click on "OK" button.  
 Name: Copy Directory  
 Source Directory: "." (here. Specifies current working directory)  
 Destination Directories: Apache Server Static Path.

Make sure that every component of the design is connected with each other.

### iii. *Steps to Create Snapshot in Urban Code Deploy.*

- a. Click on "Applications" tab.
- b. Click on Application name, i.e JPetStore App.
- c. Click on "Snapshots" link.
- d. Click on "create snapshot" link.
- e. Give the Snapshot name for e.g Test-Pipeline and click on "Save" button.
- f. Click on "Add" link , next to the Component name, ie. jenkins-jpet-component.
- g. Click on "latest available".
- h. Click on "Application" tab.
- i. Click on Application name i.e "JPetStore-App".
- j. Click on "Run button" next to the environment name, i.e Test.
- k. In the "Run Process on Test" tab, select the SnapshotName (which you created just now ) from "Snapshot" dropdown and click on submit.
- l. Now go to UCV tool.
- m. Click on "Pipeline" tab.
- n. Click on the "Pipeline name", i.e JPetStore.

### **3. Steps to Configure the UCV tool to see the Reports**

Details:

#### i. *Steps to add the App in UCD tool from Jenkins source and UCD source.*

- a) Login to Urban Code Velocity tool with valid username and password.
- b) Click on Pipelines tab and click on "create pipeline" link.
- c) Give the Name, Description and select the Team and click on Save button.
- d) Click on "Add app" and select the option as "UrbanCodeDeployment" and give the "application Name" and click on Save button.
- e) Click on "Add app" and select the option as "Jenkins" and give the "application Name" and click on Save button.

- f) Click on plus sign under the Input section.
- g) Under the "Create version", select automatically and click on "Continue".
- h) Select the Job name from Jenkins and click on "save" button. (Initially the status of job will be "Not Yet Run".
- i) Login to Jenkin tool.
- j) Navigate to the Pipeline solution for which you need to see the UCV reports.
- k) Click on "Build Now" on the LHS.
- l) Navigate to the UCV tool and check for the App created for Jenkins. It should display the recent Build number with green symbol.
- m) Check for the App created for Urban Code Deployment, It should display Snapshot version from Urban Code Deployment tool.

ii. *Steps to create the Report in UCV tool.*

- a) Login to Urban Code Velocity tool with valid username and password.
- b) Click on "Reports" tab.
- c) Click on "Add report" link.
- d) Select the type of Report from the "Type of Report".e.g "Custom"
- e) Give the Report Name in "Report name" text area.
- f) Give the brief description about the report in "Description"
- g) Click on "Add" button. Report should be created successfully.
- h) Now click on the Report name which you created just now.
- i) Click on "Add Card"
- j) Under "Select a Card", select the type of options for e.g "Deployment Count".
- k) In "Time Period" field, select the time duration for e.g. "Last 30 days".
- l) In "Environment" field, select Environment Name, for e.g. "All Environments".
- m) Under the Application field, choose the application name, e.g. Jpetstore app.
- n) In "Team" field, select "all teams".
- o) In "Line of Business", select "Not Mapped" and click on "Add". So Now User could see the details like Application deployment, Successful Deployments and Failed Deployments etc.
- p) Click on "Add Chart" link.
- q) Give the Title name in the "Title".
- r) Select the type of chart for e.g. Total Application Deployment Counts by environment".
- s) Select the time from "Time period" selection box e.g. "last 30 days" and click on "Add button. User could see the Report which created in graphical format bottom of the page.

#### **4. Steps HCL Functional Tester Configuration**

**Precondition:HCL Functional tester tool should be installed in windows machine.**

i. *Steps to Create TestScript in HFT by recording.*

- a) Open HCL Functional Tester tool
- b) Select the TestWorkBench perspective.
- c) Click on File tab, select New, and then select TestWorkBench project.(File-->New-->TestWorkBench Project)
- d) Give the project name, click on "Finish",
- e) Select "Web UI Test" and then select "Next".
- f) Give a TestName and click on "Finish".

- g) Select the Browser e.g Google Chrome , click on "Next" and "Finish".
- h) Goto the Application Url and perform some tests and close the Browser.
- i) User can see "Test Generation completed" pop-up and click on "Open Test"
- j) Open the test script from Test Navigator and click on "Run Test".
- k) Now Run the test script from TestWorkBench.
- l) Once the test is complete , automatically the Test Results will be opened.

ii. *Steps to Integrate Jenkins with RFT tool.*

- a) Download the HCL Web UI Tester(HCL Functional Tester WEb UI 9.2.0.1 Jenkins Plugin) and save it.
- b) Login to Jenkins tool as admin.
- c) Click on "Manage Jenkins" on the LHS.
- d) Click on "Manage Plugins".
- e) Goto "Advanced" tab, click on "Choose File" browse the path for HCL Web UI Tester and then click on "Upload".

iii. *Steps to create Slave Node in jenkins*

*Precondition: Agent.jar and Slave-Agent.jnlp is required to make the Slave online.*

- a) Login to Jenkins tool as admin.
- b) Click on "Manage Jenkins".
- c) Click on "Manage Nodes".
- d) Click on "New Node".
- e) Give a Node Name, select "Permanent Agent" check box and click on "OK".
- f) Now open a browser and copy paste the below Url to get the "agent.jar" which will go to the "Downloads" folder.  
ipaddressofjenkins:portnumber/jnlpJars/agent.jar
- g) Now navigate to the Nodes name which you have created.
- h) Click on the "Launch" link, which will launch Agent slave.
- i) Now open the cmd , navigate to Downloads section and type the command which you see below the "Run from agent commandline"
- j) Click on "Configure link".
- k) Under Remote root directory, give the path for the Jenkins e.g  
C:\Program Files (x86)\Jenkins\
- l) Click on "Manage Jenkins" and click on "Configure Global Security".
- m) Under Agents choose port for TCP port for JNLP agents, choose "Random " if all ports are open otherwise choose "Fixed" and give the specific port number e.g "49187" for Jenkins tool.

iv. *Steps to create job in Jenkins to run HFT test*

- a. Login to Jenkins tool with valid username and password.
- b. Click on "New Item" and give any name in the "Enter an item name"
- c. Click on "Free Style project" and click on "OK".
- d. Under the "General" tab, select the checkbox "Restrict where this project can be run "
- e. Under the "Label Expression" give the Agent Name .

- f. In "Build Triggers" tab, select the checkbox "Build after other projects are built"
- g. In " Projects to watch", give the Project Name .
- h. Select the radio button "Trigger only if build is stable".
- i. Click on "Save" to save all the changes.
- j. Click on "Configure" link on the LHS.
- k. Under "Build" tab, click on "Add build step" then select "Run an HCL Web UI Tester test".
- l. Give the details like Name, Workspace , Project , Test Suite Name, IMShared Location.
- m. Here "Test" is the name of the testscript,"Workspace" is path where HFT testscript are stored," Project Name" is the TestWorkBench projectname," IM Shared Location" is the path where Installation manager is present.
- n. Under "Exported Statistical Report" in html , give the path for the Result folder  
These results will be saved in Jenkins workspace.

## **Known Limitations**

- HCL DevSecOps Solution version 1.0 supports only Linux environment.
- Below browsers are supported:  
Google Chrome version 69.0.(64 bit)  
Mozilla FireFox version 62.03(64 bit)

