

*DEPARTMENT OF INFORMATION TECHNOLOGY* Experiment No5

|  |  |  |  |
| --- | --- | --- | --- |
| **Semester** | Semester VIII | | |
| **Subject** | DevOps Lab | | |
| **Subject Professor In-charge** | Prof. Yash Shah | | |
| **Laboratory** | L11B | | |
| **Student Name** | | Ashwini Jadhav | |  |
| **Roll Number** | | 17101B0038 | |  |
| **Grade and Subject Teacher’s Signature** | |  |  |  |

|  |  |  |
| --- | --- | --- |
| **Experiment Number** | 5 | |
| **Experiment Title** | To install Jenkins and create different builds on a single job | |
| **Resources / Apparatus Required** | Hardware:  Compatible Computer System | Java Development Kit (JDK) v.11 |
| **Objectives** | Explore and implement builds on Jenkins | |

|  |  |
| --- | --- |
| **Theory** | **What is jenkins?**  Jenkins is an open source Continuous Integration server capable of orchestrating a chain of actions that help to achieve the Continuous Integration process (and not only) in an automated fashion.  Jenkins is a widely used application around the world that has around 300k installations and growing day by day.  **Need of Jenkins:**  The reason Jenkins became so popular is that of its monitoring of repeated tasks which arise during the development of a project. For example, if your team is developing a project, Jenkins will continuously test your project builds and show you the errors in early stages of your development.  By using Jenkins, software companies can accelerate their software development process, as Jenkins can automate build and test at a rapid rate. Jenkins supports the complete development lifecycle of software from building, testing, documenting the software, deploying and other stages of a software development lifecycle.  **Installation steps:**   1. **Double-click on unzipped jenkins.msi. You can also Jenkin using a WAR (Web application ARchive) but that is not recommended.**   IMG_256   1. In the Jenkin Setup screen, click Next.   IMG_256   1. Choose the location where you want to have the Jenkins instance installed (default location is C:\Program Files (x86)\Jenkins), then click on ****Next**** button.   IMG_256   1. Click on the Install button.   IMG_256   1. Once install is complete, click Finish. 2. After completing the Jenkins installation process, a browser tab will pop-up asking for the initial Administrator password. To access Jenkins, you need to go to browse the following path in your web browser.   http://localhost:8080  If you can access the above URL, then it confirms that Jenkins is successfully installed in your system.  IMG_256   1. The initial Administrator password should be found under the Jenkins installation path (set at Step 4 in Jenkins Installation).   For default installation location to C:\Program Files (x86)\Jenkins, a file called ****initialAdminPassword**** can be found under C:\Program Files (x86)\Jenkins\secrets.  However, If a custom path for Jenkins installation was selected, then you should check that location for ****initialAdminPassword**** file.  IMG_256   1. Open the highlighted file and copy the content of the ****initialAdminPassword****file.   IMG_256   1. Paste the password it into browser's pop-up tab ([http://localhost:8080/login?form=%2F](http://localhost:8080/login?form=/)) and click on Continue button.   IMG_256   1. Click on the "Install suggested plugins button" so Jenkins will retrieve and install the essential plugins   IMG_256   1. Jenkins will start to download and install all the necessary plugins needed to create new Jenkins Jobs. |
| **Output** | 1. Create new project by clicking on New Item on the Dashboard   Screenshot (327)   1. In the next screen, enter the Item name, in this case we have named it Devops Exp 5. Choose the ‘Freestyle project option’   Screenshot (328)   1. The following screen will come up in which you can specify the details of the job.   Screenshot (329)   1. We need to specify the location of files which need to be built. If you have a repository hosted on Github, you can also enter the url of that repository here. In addition to this, you would need to click on the Add button for the credentials to add a user name and password to the github repository so that the code can be picked up from the remote repository.   Screenshot (330)   1. Go to the Build section and click on Add build step → Execute Windows batch command. In the command window, enter the following commands and then click on the Save button.   Screenshot (331)   1. Once saved, you can click on the Build Now option to see if you have successfully defined the job.   Screenshot (332)   1. Once the build is scheduled, it will run. The following Build history section shows that a build is in progress.   Screenshot (333)   1. Once the build is completed, a status of the build will show if the build was successful or not. In our case, the following build has been executed successfully.   Screenshot (334)   1. Click on the Console Output link to see the details of the build   Screenshot (336)   1. To make changes to your build, click on configure.   Screenshot (340)   1. To carry out triggers, in this case ‘build periodically’, we go to build triggers.   Screenshot (341)   1. In ‘Poll SCM’,  it periodically polls the SCM to check whether changes were made (i.e. new commits) and builds the project if new commits where pushed since the last build   **Screenshot (346)** |
| **Conclusion** | Thus, we have implemented builds using jenkins and saved project to jenkins. |