Assignment - I

Name - Priyadarshani Anil Satpute
class - BEIT - 53
Sub - DevOps Lab
Quiz Score - 9/10

Q.1 What is Jenkins? How does the Jenkins helps for faster software development?

Ans- Jenkins is an open source automation server. With Jenkins, organizations can accelerate the software development process by automating it. Jenkins manages and controls Software delivery process throughout the entire lifecycle, including build, document, test, package, stage, deployment, static code analysis and much more. Jenkins is used to build and test your software projects continuously making it easier for developers to integrate changes to the project, and making it easier for users to obtain a fresh build.

Q.2 List the DevOps open Source tools along with their benifits.

Ans- ~~Docker~~ -

- Define and plan - which focuses on planning DevOps workflows for iterations, release management, and issue tracking. Notable tools vendors in this space include Atlassian, CA Technologies, IBM, iRise, and Jama software

- code, build and configure - which focuses on code development and review. Source code management and code merging. Notable tools / tool vendors include Bit Bucket, Electric cloud, GitLab, GitHub, and IBM.
- Test - which verifies that the quality of the software release and code are maintained throughout the development process and that the highest quality deploys to production.
- Packaging and preproduction - which refers to achivities involvs the relese is ready for development. Notable tool vendors include IBM, Inedo's ProGet, Jfrog's Artifactory.
- Release, deploy, and orchestration - which is the process of actually releasing software and usually involves change management. Tools / tool vendors in this space include Automatic, Clarive, BMC, IBM, Flexagon, VMware and XebiaLabs.

Q 3 What is the Docker technology and what are its benefits.

Ans Docker is an open source application that allows administrators to create, manage, deploy and replicate applications using containers. Containers allow a developer to package up an application with all of the parts it needs, such as libraries and other dependencies, and deploy it as one package. By doing so, thanks to the container, the developer can rest assured

that the application will run on any other Linux machine regardless of any customized settings that machine might have that could differ from the machine used for writing and testing the code.