

CIT 425: Software Development and Operations (DevOps)

Gandaki University

IV Year, I Sem

Program: Bachelor of Information Technology

Subject: Software Development and Operations (DevOps)

Course Code: CIT 425

Credit Hour: 2+1=3

Contact Hours: 45

Year: IV

Semester: I

Lab Type: Computer Lab

Lab Hours: 15

1 Course Objectives

The key objectives of learning Software Development and Operations (DevOps) are:

1. Learn and understand how various technologies work together in DevOps. Get a firm understanding in DevOps Processes, Tools and Technologies.
2. Learn the basics of working in DevOps environments like Linux, AWS, Bash & Python Scripting, Jenkins, Ansible, Docker, Kubernetes and more

Course Description

This course provides an introduction to the concepts, practices, and tools used in DevOps. Students will learn about the fundamental principles of DevOps and how they can be used to improve the software development and deployment process. The course will cover a range of topics including automation, continuous integration and delivery, testing, and monitoring.

This course starts from very basics of command line, hands on demonstrations of many tools & technologies. Also most importantly it will show the readers how various technologies in DevOps work together by setting up their own projects.

2 Course Outcomes

- This course will enable the students to understand the fundamental principles and benefits of DevOps and how it can improve software development and deployment processes.

- This course will enable students to apply continuous integration and delivery practices to automate the software build, test, and deployment process.

3 Course Content

3.1 Introduction (3 Hrs)

1. Introduction to DevOps: Definition, Scope, Application.
2. Introduction to Continuous Integration(CI) and Continuous Delivery(CD).
3. Prerequisites Info and Setup: Windows, MacOS, Cloud Setup, Linux Setup. Server Management in Linux.
4. Basics of Networking

3.2 DevOps Basics with Virtualization (5 Hrs)

1. Basics of Virtualization: Definition, Types and Heirarchy of Virtualizations.
2. Learn Virtualization: Manual and Automatic. OS Environments: Linux, MacOS, and Windows
3. Bash Scripting. Basics of Scripting.
4. Automating day to day Admin Tasks.

3.3 DevOps on Cloud (6 hrs)

1. Essentials of Cloud Computing
2. Cloud and Virutalization Architecture
3. Cloud Deployment Architecture
4. Cloud Providers - An Overview
5. Introduction to DevOps on Cloud Infrastructure like Amazon Web Service(AWS)

3.4 GIT- A Version Controlling Tool (6 hrs)

1. Introduction to Version Control
2. Git - A CLI, Essentials of GIT in industry.
3. Practice with GIT: Tracking, Monitoring, Ignoring, Staging, Merging etc.
4. Working with Remotes
5. GIT Workflows

3.5 Jenkins (5 Hrs.)

1. Essentials of Continuous Integration.
2. Architecture of Jenkins.
3. Jenkins Tool Management System.
4. User Management in Jenkins.
5. Jenkins Workshop: Authentication, Authorization.
6. Maven Overview, Lifecycle
7. Creating Jobs and Automatic Build Settings
8. Building Delivery Pipeline. Plugin Management and Notification System.

3.6 Docker (8 Hrs.)

1. Introduction to Real World Transportation Challenges. Docker Introduction.
2. Architecture of Docker
3. Docker Containers: Creation, Share and Publish.
4. Docker Networking: Network Types, Docker Compose.
5. Docker Swarm: Introduction, Creating and Scaling docker bundles.

3.7 Kubernetes (4 Hrs.)

1. Introduction to Kubernetes. Kubernetes Cluster Architecture - Brief Overview.
2. Concepts: Pods, Replica Sets, Deployments and Namespaces.
3. Kubernetes Services and Networking
4. Persistent volumes and persistent volume claims - An Overview.
5. Design of PODS.
6. Understanding labels, selectors, jobs and schedulers.

3.8 Ansible (8 Hrs.)

1. Introduction to Ansible: Basics and Architecture of Ansible. Control Machine and Managed Node Requirements
2. Inventory: Hosts, Groups.
3. Ansible Modules
4. Adhoc Commands

5. YAML script
6. Playbook: Handlers and Variables, Tasks, Privilege Escalation.
7. Roles: Directory Structure, Duplication and Execution, Default Variables, Dependencies, Search Path, Ansible Galaxy
8. Including and Importing: Playbooks, Task Files and Roles.

4 Lab Objectives and Case Study

4.1 Case Study

An app concept (existing or proposed) is prepared as a project at the end of the semester. The concept shall:

1. expose their data and functionality through service interfaces.
2. allow teams to communicate with each other through these interfaces.

4.2 Lab Objectives

1. To understand DevOps practices which aims to simplify Software Development Life Cycle
2. To be aware of different Version Control tools like GIT, CVS or MercurialStudy of different web development To understand DevOps practices which aims to simplify Software Development Life Cycleframeworks for SEO.
3. To Integrate and deploy tools like Jenkins and Maven, which is used to build, test and deploy applications in DevOps environment
4. To be familiarized with selenium tool, which is used for continuous testing of applications deployed.
5. To use Docker to Build, ship and manage applications using containerization
6. To understand the concept of Infrastructure as a code and install and configure Ansible tool.

5 Textbook(s):

1. Gene Kim, Patrick Debois, John Willis, Jez Humble and John Allspaw (2021). The DevOps Handbook: How to Create World-Class Agility, Reliability, and Security in Technology Organizations
2. Gene Kim, Kevin Behr, George Spafford(2018). The Phoenix Project: A Novel about IT, DevOps, and Helping Your Business Win.

6 Reference(s):

1. Jennifer Davis and Ryn Daniels(2016). Effective DevOps: Building a Culture of Collaboration, Affinity, and Tooling at Scale .
2. Jez Humble and David Farley(2010). Continuous Delivery: Reliable Software Releases through Build, Test, and Deployment Automation.