Contents

CSV 101 - DevOps Overview 2

CSV 203 - Build & Release Management 3

CSV 204 - Build & Release Management Lab 5

CSV 205 – DevOps Automation (Automation & Continuous Monitoring) 6

CSV 206 – DevOps Automation Lab (Automation & Continuous Monitoring Lab) 7

CSV 401 - System Monitoring 8

CSV 402 - System Monitoring Lab 9

CSV 403 - System Provisioning & Code Quality Testing 10

CSV 404 - System Provisioning & Code Quality Testing Lab 11

CSV 101 - DevOps Overview

Unit 1: Traditional SDLC Models, their problems, & their solutions overview

Software, History of Software Engineering and Software, Development Methodologies, Traditional Software Development Models, Waterfall Model, Classical Waterfall Model, Traditional IT Organizations, Developers vs IT Operations Conflict, Birth of Agile, Four Values of the Agile Manifesto, Agile and Lean

Unit 2: Introduction to DevOps

Definition of DevOps: Challenges of traditional IT systems & processes, History and emergence of DevOps, DevOps definition and principles governing DevOps, DevOps and Agile, The need for building a business use case for DevOps

Unit 3: Purpose of DevOps

Purpose of DevOps, Minimum Viable Product (MVP), Benefits of MVP, Application Deployment, Automated Application Deployment, Application Release Automation (ARA), Components of Application Release Automation (ARA), Continuous Integration, Best Practices of CI, Benefits of CI, Continuous Delivery, Process

Unit 4: CAMS (Culture, Automation, Measurement and Sharing)

CAMS – Culture, Cultural aspects of DevOps, Continuous Improvement and Problem Solving, encourage, Experimentation and Learning, CAMS – Automation, Delivering high value - DevOps way, Continuous Delivery Automation: CAMS – Measurement, Metrics used for tracking, Performance Predictors, Continuous Monitoring, CAMS – Sharing, Test-driven development, Configuration Management, Infrastructure Automation, Root Cause Analysis, Blamelessness, Organizational Learning, Test Driven Development, TDD – Categories of Tests, Configuration Management, Source Code Management - Version Control, Infrastructure Automation Tools, Root Cause Analysis

Unit 5: Linux Basics and Admin

Introduction to Linux (Operating System), Importance of Linux in DevOps, Linux Basic Command Utilities, Linux Administration, Environment Variables, Networking, Linux Server Installation, RPM and YUM Installation

Unit 6: Introduction to DevOps Tools & Technologies

Git & GitHub (SCM), Docker (Containerization), Jenkins (CI/CD Pipelines), Terraform (Provisioning), Maven (Build & Release Management), Ansible (Configuration Management), Selenium (Test Automation), AWS (Cloud Computing), SonarQube (Code Quality Checking), Prometheus/Nagios (Monitoring)

CSV 203 - Build & Release Management

Unit 1: Introduction to Modern Build and Release Management Build concepts, Introduction to modern Build and Release Management, Cloud-native build management, Build Reporting – Sample GitHub Actions Report, Build Reporting – Build status and badges, Release Planning in Agile environments, Containerization and packaging, Authentication and Authorization in CI/CD, Zero-downtime deployments, Declarative Dependency Management in modern ecosystems

Unit 2: Dependency Management in Modern Ecosystems Build Tools – Webpack, NPM scripts, Poetry (Python), Using package registries, NPM Registry – Local, Public, Private, NPM package resolution algorithm, Dependency Management in JavaScript/TypeScript projects, Example of Dependency Management: package.json, Example of Dependency Management: Poetry pyproject.toml, Dependency Identification and versioning, Dependency Scopes in modern frameworks, Transitive Dependencies in npm and pip, Features and challenges of Transitive Dependencies

Unit 3: Documentation and Reporting in Modern Development Documentation Overview, Different types of documentation (API docs, user guides, architecture docs), Automated Reporting, Docusaurus for documentation sites, Documentation as Code, Swagger/OpenAPI for API documentation, Advanced Reporting tools (Datadog, New Relic), Unit Testing in JavaScript and Python, Unit Testing Techniques for frontend and backend, Testing Frameworks – Jest, PyTest, Code Coverage for modern languages, How Code Coverage is Calculated in dynamic languages, Modern Code Coverage Tools, Comparison of Code Coverage Tools for JS/TS and Python, Sample Code Coverage Report in modern CI tools, Code Coverage – Pros and Cons in Agile development

Unit 4: Understanding a Modern Release Cycle Project Release Lifecycle in cloud-native environments, Different Stages of a Cloud-Native Release Cycle, Distributed Version Control, Different Types of Distributed Version Control Systems, Setting up GitLab, First Time Git Config with 2FA, Creating an Organization in GitLab, Creating a Repository in GitLab, Working with Feature Branches in GitLab, Merging code using Merge Requests, Deploying Builds to Cloud Platforms, GitLab CI/CD pipeline stages, GitHub Actions workflow syntax, Automated Rollback Strategies

Unit 5: Advanced Build and Release Strategies Continuous Integration and Continuous Delivery (CI/CD) in cloud-native environments, GitLab CI/CD advanced features, GitHub Actions advanced workflows, Azure DevOps pipelines, Blue-Green Deployments in Kubernetes, Canary Releases with Istio, Feature Flags implementation, A/B Testing in cloud environments, Rolling Updates in container orchestration, Artifact Management with Artifactory, Trunk-Based Development, GitFlow and GitHub Flow strategies

Unit 6: DevOps Metrics and Observability Key Performance Indicators (KPIs) for Cloud-Native DevOps, Mean Time to Recovery (MTTR) in microservices, Change Failure Rate in event-driven architectures, Deployment Frequency in serverless environments, Lead Time for Changes in GitOps workflows, Time to Market for cloud-native apps, Customer Ticket Volume and automated issue tracking, Application Performance Monitoring (APM) with tools like Datadog, Error Budgets in SRE practices, Service Level Objectives (SLOs) for cloud services, Service Level Indicators (SLIs) in distributed systems, Service Level Agreements (SLAs) for cloud-native applications

CSV 204 - Build & Release Management Lab

Unit 1: Node.js and npm Fundamentals Setting up Node.js and npm, understanding package.json structure, creating a project using npm, understanding package-lock.json files, using private npm registries (e.g., Verdaccio) in npm configuration.

Unit 2: Package Management and Dependencies Artifactory requirements in package.json, working with environment variables and secrets management, deploying npm packages to Artifactory, using npm scopes for organization-level packages.

Unit 3: Testing and Code Quality Using Jest to run unit tests for JavaScript/TypeScript projects, implementing linting and code formatting with ESLint and Prettier.

Unit 4: Build Tools and Development Environment Using Webpack for bundling and optimization, using VS Code for running npm scripts and debugging.

Unit 5: Containerization Creating Docker images for Node.js applications, automatic building and publishing of container images with GitHub Actions.

Unit 6: CI/CD Implementation Implementing a CI/CD pipeline using GitLab CI for a Node.js project.

CSV 205 – DevOps Automation (Automation & Continuous Monitoring)

Unit 1: Introduction to Automation

The Software Delivery Pipeline, Overview of the Continuous Delivery Pipeline, Fully Automated Software Delivery Process, The Build Process, Automated build, Automated Test, Automated Deployment, Benefits of Automated Deployment, Automated Deployment and DevOps Adoption, Overview of Rapid Application Development (RAD), Phases in RAD, Essential Aspects of RAD, Code generation, Categories of Code Generators

Unit 2: Advantages of Automation

Advantages of Automation, Automation Scenarios, Archiving Logs, Auto-Discard Old Archives, MySQL (RDBMS) Backups, Email Web Server Summary, Ensure Web Server is Running, User Command Validation, Disk Usage Alarm, Sending Files to Recycle Bin, Restoring Files from Recycle Bin, Logging Delete Actions, File Formatter, Decrypting Files, Bulk File Downloader, System Information, Install LAMP Stack, Get NIC's IP, Scenarios Where Automation Prevents Errors

Unit 3: Automating the Linux Tasks

Introduction to cron/crontab, scope of crontab, use-case of crontab, understanding crontab syntax, writing your own cron tasks, editing/updating the crontab tasks, seeing the cron tasks

Unit 4: Introduction to Monitoring

What is Monitoring, need of monitoring, different types of monitoring, Linux Monitoring metrics, Linux In-Build Monitoring metrics, SNMP Introduction, Introduction to various other monitoring tools

Unit 5: Monitoring using Nagios

What is Nagios, Nagios Installation, Configuring Nagios, Infrastructure monitoring using Nagios, Managing the Nagios server, Nagios Notifications, Nagios Core Backups, Nagios Managing Time Options, Nagios Event Handlers

Unit 6: Advanced Automation and Monitoring Techniques

Infrastructure as Code (IaC), Ansible for configuration management, Terraform for infrastructure provisioning, Kubernetes for container orchestration, Prometheus for metrics collection and alerting, Grafana for visualization, ELK stack for log management, Continuous monitoring in microservices architectures, Chaos engineering principles and tools

CSV 206 – DevOps Automation Lab (Automation & Continuous Monitoring Lab)

Unit 1: Scripting Fundamentals Conditional statements and loops, working with automation scripts that save time and effort.

Unit 2: File and Database Management Automation Automatically delete archive files that are older than two days, take MySQL backups every 12 hours and move them to the backup directory.

Unit 3: Server Monitoring and Management Email the summary of web server requests every day, continuously monitor and restart the web server if it is not running.

Unit 4: System Administration and Security Block executing forbidden commands, monitor disk usage and alert if it is beyond the given threshold.

Unit 5: Scheduling and Time-based Tasks Working with Cron, managing time in Nagios.

Unit 6: Monitoring Tools and Techniques Nagios installation and configuration, working with SNMP, log monitoring using Nagios, implementing a basic Prometheus and Grafana monitoring setup.

CSV 401 - System Monitoring

Unit 1: DevOps and Monitoring

Introduction to monitoring, Goals of monitoring, DevOps approach to monitoring, Network operations center, Role of NOC in DevOps world, Telemetry and metrics, Types of monitoring: end user, infrastructure, application, log monitoring and analysis

Unit 2: End User Monitoring

Overview, Objectives of end user monitoring, Types of end user monitoring, Real user monitoring, Synthetic transaction monitoring, Server-side monitoring, Benefits of end user monitoring, Tools overview

Unit 3: Infrastructure Monitoring

Overview of infrastructure monitoring, Monitoring components and metrics, Agent vs agentless monitoring, Reactive monitoring vs proactive monitoring, Cloud vs on premise, Network and security monitoring, Infrastructure monitoring challenges, Tools overview

Unit 4: Application Monitoring

Overview, How to Measure Application Performance, Key Functionalities, Application vs infrastructure monitoring, Monitoring components and metrics, Dependency monitoring, Tool's overview

Unit 5: Log Monitoring and Analysis

Overview, Objectives of log monitoring, Metrics, Log monitoring vs analysis, Log analysis techniques, Purpose and benefits of log analysis, Log analysis best practices, Purpose and Benefits of Log Analysis, Best Practices, Tools overview

Unit 6: Monitoring Techniques and AI in Monitoring

Visualization – Dashboards, Alerts, Alert triage process, DevOps dashboard with Hygieia, AI in monitoring – AIOps, Machine learning for anomaly detection, Predictive analytics in monitoring, Automated root cause analysis, Intelligent alerting and noise reduction, Tools overview

CSV 402 - System Monitoring Lab

Unit 1: Basic Monitoring Stack Setup Setting up a basic monitoring stack (Prometheus, Grafana, Node Exporter), configuring and using Nagios for infrastructure monitoring.

Unit 2: Log Analysis and Custom Metrics Implementing log analysis using the ELK stack (Elasticsearch, Logstash, Kibana), creating custom Prometheus exporters for application-specific metrics.

Unit 3: Visualization and Synthetic Monitoring Designing and implementing Grafana dashboards for various metrics, setting up synthetic monitoring using tools like Selenium and Puppeteer.

Unit 4: Distributed Tracing and Application Performance Implementing distributed tracing using Jaeger or Zipkin, configuring and using New Relic for application performance monitoring.

Unit 5: Enterprise Monitoring and Log Management Setting up and configuring Zabbix for enterprise monitoring, implementing log rotation and archiving strategies.

Unit 6: Advanced Monitoring Techniques Creating custom alerts and notifications in Prometheus Alertmanager, implementing a basic AIOps solution using machine learning for anomaly detection, setting up and configuring Datadog for full-stack observability, implementing SLO-based alerting using Prometheus and Alertmanager.

CSV 403 - System Provisioning & Code Quality Testing

Unit 1: Introduction to Provisioning

What is Provisioning – Basic Definition, Software Definition, Concepts of Provisioning, Why Provisioning Should be Exclusive, Configuration Management, Configuration Management Tools, Why Provisioning is not Configuration Management, Provisioning Tools, Test Machines for Provisioning, Deployment, Relationship between Deployment and Provisioning

Unit 2: On Premise Provisioning

Understanding ’On Premise Provisioning, what is On Premise?, Provisioning Infrastructure, Server Templating, Server Templating, Connectivity with Servers, What is a Client?, What is Templating?, Server Side Templating, Challenges of Server Side Templating, Advantages of Server Side Templating, Server Side Templating Vs Client Side Templating

Unit 3: Provisioning on Cloud

Introduction, Cloud Providers, Benefits of Cloud Computing, Types of Cloud Computing, Types of Deployment Model, Types of Service Model, Life Cycle of Provisioning on Cloud, Automated Provisioning on Cloud, what is Cloud Automation? Benefits of Cloud Automation, what is SonarQube? Code Quality Checks Performed by SonarQube

Unit 4: Code Quality and Static Analysis

Features of SonarQube, Code Scanner, Application of Code Scanner, Organizational Improvement Using Code Scanner, On Premise to Cloud Migration Strategies, what is Cloud Migration? Types of Cloud Migration Strategies, Benefits of Cloud Migration, Network Security Enablement from On-Premises to Cloud, What are Microservices?, Azure Kubernetes Service (AKS), Benefits of AKS, Benefits of EKS

Unit 5: System Provisioning and Configuration Management

State of Various Tools in Provisioning and Configuration, Infrastructure as Code, Continuous Integration/Continuous Deployment, Configuration Management, Configuration Management in DevOps, Monitoring, Reasons for Using Provisioning and Configuration Tools, Automation, Preventing Errors and Tracking of Changes, Tools and their Capabilities

Unit 6: Advanced Provisioning and Testing Strategies

Immutable Infrastructure, Container Orchestration with Kubernetes, Serverless Computing, Infrastructure Testing (e.g., TestInfra, Goss), Chaos Engineering principles and tools, Performance Testing strategies, Security Testing in CI/CD pipelines, Contract Testing for Microservices, Property-Based Testing, Mutation Testing

CSV 404 - System Provisioning & Code Quality Testing Lab

Unit 1: Infrastructure as Code Basics Working with Infrastructure as Code: Automation of your Infrastructure, AWS Configuration for Terraform, opening an AWS Account, creating IAM Admin User for Terraform.

Unit 2: AWS Resource Management Working on creating security groups, spinning up EC2 instances using github.com, exploring variables and resources.

Unit 3: Terraform Advanced Concepts Working with modules, working with state management.

Unit 4: AWS Network and Security Creating an AWS VPC using Terraform, creating an AWS security group using Terraform.

Unit 5: AWS Identity and Storage Creating an AWS IAM policy using Terraform, creating an AWS S3 bucket instance using Terraform.

Unit 6: Code Quality and Containerization Implementing SonarQube for code quality analysis in a CI/CD pipeline, setting up and configuring Kubernetes cluster using Terraform.