

# DevSecOps Essentials

# COURSE OUTLINE





## DevSecOps Essentials – v1

---

### Course Outline

#### Module 01: Application Development Concepts

- **History of Application Development**
  - What is Application Development
  - Programming
  - Web and Mobile Development
- **Evolution of Application Development Methodologies**
  - Evolution of Application Development
  - Traditional Waterfall development model
  - Agile development methodology
  - Methodology Comparison
  - DevOps methodology
  - Choosing a Methodology
- **Introduction to Application Architectures**
  - Application Architectures
  - Types of Application Architectures
  - Monolithic Architecture
  - Microservices Architecture
  - Microservices Challenges
  - Serverless Architecture
  - Limitations to Serverless Architecture
  - Choosing an Application Architecture
  - Working with Applications in Production
  - Applications in Production
  - Application Production Environments
  - Designing the Production Environment

- Deployment Strategies
- Deployment Tools for Applications
- Monitoring and Troubleshooting
- Monitoring Tools in Production
- Continuous Monitoring and Management of Applications
- **Introduction to the Application Development Lifecycle**
  - Application Development Lifecycle
  - Steps 1 through 3 in the ADLC
  - Steps 4 through 6 in the ADLC
- **Application Testing and Quality Assurance**
  - Testing and Quality Assurance
  - Types of Application Tests
  - Best Practices for Application QA
  - Application Performance Management
  - Why is APM important?
  - Using Tools for APM
  - Popular APM Tools
- **Application Monitoring, Maintenance and Support**
  - Application Integration
  - What is Application Integration
  - Types of Application integration
  - Best Practices for Application Integration
  - Application Maintenance and Support
  - Best Practices for Maintenance and Support
  - Continuous Monitoring
  - Why is Continuous Monitoring Important?
  - What Tools assist with Monitoring
  - Configuration and Change Management
  - Role of Configuration and Change Management

## Module 02: Application Security Fundamentals

- **What is Secure Application Development**
  - Secure Application Development
  - Secure App Dev Principles
  - Secure App Dev Practices
- **Need for Application Security**
  - Application Security is a Need
  - Why is Application Security Important?

- Cloud Computing
- Artificial Intelligence and Machine Learning
- **Common Application Security Risks and Threats**
  - Consequences of Security Breaches
  - Common Attacks to Applications
- **OWASP Top 10**
  - What is the OWASP Top 10
  - List of OWASP Top 10 App Security Risks
  - Injection Attacks
  - Broken Authentication and Session Management
  - Cross-Site Scripting (XSS)
  - Insecure Direct Object References
  - Security Misconfiguration
  - Sensitive Data Exposure
  - Broken Access Control
  - Insufficient Logging and Monitoring
  - Insecure Cryptographic Storage
  - Insecure Communication
- **Application Security Techniques**
  - Security Techniques
  - Input Validation
  - Output Encoding
  - Encryption and Hashing
- **Secure Design Principles**
  - Security Requirements
  - Secure Design Principles
  - Least Privilege
  - Defense in Depth
  - Fail Securely
  - Secure by Default
  - Separation of Duties
  - Zero Trust
- **Threat Modeling**
  - Introduction to Threat Modeling
  - Benefits of Threat Modeling
  - Types of Threat Modeling
  - STRIDE Threat Modeling
  - Trike Threat Modeling

- PASTA Threat Modeling
- VAST Threat Modeling
- Threat Modeling Best Practices
- Evaluating Risk
- **Secure Coding**
  - Secure Coding Practices
  - Secure Coding in Action
- **Secure Code Review**
  - Secure Code Review
  - Secure Code Review in Action
- **SAST and DAST Testing**
  - Testing Methods in Action
  - Static Application Security Testing (SAST)
  - Dynamic Application Security Testing (DAST)
- **Secure Configurations**
  - Secure Configurations
  - Secure Configurations in Action
- **Educating Developers**
  - Educating Developers on Security
  - Ensuring Application Security
- **Role of Risk Management in Secure Development**
  - Security and Compliance Standards
  - Role of Risk Management in Developing Secure Applications
  - What is Risk Management
  - Four Steps of Risk Management
  - Risk Management in App Development
  - Best Practices for Mitigating Risk
- **Project Management Role in Secure Application Development**
  - Project Management for Protecting the Scope of Security in Development
  - What is Project Management?
  - PM use in App Development
  - Role of the Project Manager
  - PM Best Practices for Secure App Development

## Module 03: Introduction to DevOps

- **Introduction to DevOps**
  - Evolution of DevOps
  - Agile Development Methodology

- Benefits of DevOps
- Improved Quality
- Cost Savings
- **DevOps Principles**
  - DevOps Principles
  - Automation in DevOps
  - Infrastructure as Code (IaC)
- **DevOps Pipelines**
  - Principles of DevOps
  - Continuous Integration in DevOps
  - Continuous Delivery in DevOps
  - Continuous Deployment in DevOps
- **DevOps and Project Management**
  - Project Management and DevOps
  - Waterfall and DevOps
  - Agile and DevOps
  - Lean and DevOps

## Module 04: Introduction to DevSecOps

- **Understanding DevSecOps**
  - What is DevSecOps?
  - Goals of DevSecOps
- **DevOps vs. DevSecOps**
  - DevOps vs. DevSecOps
  - Emphasizing DevSecOps
- **DevSecOps Principles**
  - DevSecOps Principles
  - DevSecOps Collaboration
  - DevSecOps Automation
  - DevSecOps Security Testing
- **DevSecOps Culture**
  - Developing a DevSecOps Strategy
  - Challenges in Building a DevSecOps Culture
  - Best Practices for Building a DevSecOps Culture
- **Shift-Left Security**
  - What is Shift-Left Security?
  - Benefits of Shift-Left Security
  - Implementing Shift-Left Security

- Getting Started with DevSecOps
- **DevSecOps Pipelines**
  - DevSecOps Pipeline Overview
  - Secure Code Review
  - Container Security
  - DevSecOps Pipelines
  - DevSecOps Pipeline Steps
- **Pillars of DevSecOps**
  - Three Pillars of DevSecOps
  - The Importance of People in DevSecOps
  - The Importance of Process in DevSecOps
  - The Importance of Technology in DevSecOps
- **DevSecOps Benefits and Challenges**
  - Benefits of DevSecOps
  - Challenges of DevSecOps

## Module 05: Introduction to DevSecOps Management Tools

- **Project Management Tools**
  - Jira Project Management Software
  - Confluence Collaboration Software
  - Slack Team Communication Software
  - Microsoft Teams Collaboration Software
- **Integrated Development Environment (IDE) Tools**
  - Integrated Development Environments (IDEs)
  - Eclipse
  - Visual Studio
- **Source-code Management Tools**
  - Source-Code Management with GitHub
  - Source-Code Management with GitLab
  - Source-Code Management with Azure DevOps
- **Build Tools**
  - Introduction to Build Software
  - Types of Build Software
  - Maven
- **Continuous Testing Tools**
  - Introduction to Continuous Testing Software
  - Selenium
  - TestComplete



- Katalon Studio
- Gradle
- Conclusion

## Module 06: Introduction to DevSecOps Code and CI/CD Tools

- **Continuous Integration Tools**
  - Continuous Integration Overview
  - Jenkins
  - Bamboo
  - Other CI Tools
- **Infrastructure as Code Tools**
  - Introduction to Infrastructure as Code (IaC)
  - Terraform
  - Ansible
  - CloudFormation
  - Pulumi
- **Configuration Management Tools**
  - Configuration Management
  - Chef for Configuration Management
  - Puppet and Chef for Configuration Management
  - Containers Overview
  - Docker Overview
  - Kubernetes Overview
  - AWS Container Services
  - Container Management in Azure
  - Container Management in GCP
- **Continuous Monitoring Tools**
  - Why Continuous Monitoring is Critical in DevSecOps
  - Splunk for DevSecOps Monitoring
  - Nagios for DevSecOps Monitoring
  - ELK for DevSecOps Monitoring
  - AWS Config for DevSecOps Monitoring
  - Microsoft Defender for Cloud Developer Security
  - DevSecOps Management and Monitoring software tools – Conclusion



## Module 07: Introduction to DevSecOps Pipelines

- **Role of DevSecOps in the CI/CD Pipeline**
  - DevSecOps in CI/CD Pipeline
  - DevSecOps in Development Lifecycle
  - Ensuring Secure Deployments in DevSecOps
- **DevSecOps Tools**
  - DevSecOps Tools
  - Code Analysis Tools
  - Vulnerability Scanning Tools
  - Security Testing Tools
  - Continuous Monitoring Tools
- **Embracing the DevSecOps Lifecycle**
  - DevSecOps Lifecycle
- **DevSecOps Ecosystem**
  - Key Elements of DevSecOps Ecosystem
  - Key Elements of DevSecOps Pipeline and Ecosystem
- **Key Elements of the DevSecOps Pipeline**
  - Keys to a Successful DevSecOps Pipeline
- **Integrating Security into the DevOps Pipeline**
  - Integrating Security in DevOps Pipeline
  - Importance of Security in CI/CD Pipeline
  - Secure Coding Practices
  - Access Control
  - Continuous Monitoring and Incident Response

## Module 08: Introduction to DevSecOps CI/CD Testing and Assessments

- **Implementing Security into the CI/CD Pipeline and Security Controls**
  - Why We Need Continuous Security in DevOps
  - The Benefits of Continuous Security in DevOps
  - Implementing Continuous Security in DevOps
  - Security Controls to Protect the CI/CD Pipeline
- **Continuous Security in DevSecOps with Security as Code**
  - Why Continuous Application Security Testing is Important for Your Business
  - The Benefits of Continuous Application Security Testing
  - Implementing Continuous Security in DevOps
- **Continuous Application Testing for CI/CD Pipeline Security**
  - Continuous Testing for CI/CD Pipeline Security

- Types of Continuous Testing
- Different Types of Testing
- Continuous Testing Best Practices
- Best Practices for Implementing Security as Code
- Implementing Security as Code
- **Application Assessments and Penetration Testing**
  - Types of Application Assessments
  - Types of Assessments to integrate into CI/CD Pipeline
  - Features of different types of assessments in CI/CD Pipeline
  - Automated Vulnerability Scanning Tools
  - Vulnerability Scanning
  - Vulnerability Scanning in CI/CD Pipeline
  - Integrating Vulnerability Scanning into CI/CD Pipeline
  - Best Practices for Implementing Vulnerability Scanning in CI/CD Pipeline
  - Penetration Testing
  - Penetration Testing in the CI/CD Pipeline

## Module 09: Implementing DevSecOps Testing & Threat Modeling

- **Integrating Security Threat Modeling in Plan Stage**
  - Introduction to Security Threat Modeling
  - Integrating Security Threat Modeling in the Planning Stage of Application Development
  - Importance of logging and monitoring of applications
  - Importance of configuration management
- **Integrating Secure Coding in Code Stage**
  - Importance of code testing
  - Secure Application Development Lifecycle
  - Build Stage Security Tools and Techniques
  - Test Stage Security Tools and Techniques
  - Release Stage Security Tools and Techniques
  - Deploy Stage Security Tools and Techniques
  - Secure Coding Practices in the Application Coding Stage
  - Best Practices for Secure Coding
- **Integrating SAST, DAST and IAST in Build and Test Stage**
  - Integrating SAST, DAST, and IAST in the Build Stage
  - Benefits of Integrating SAST, DAST, and IAST in the DevSecOps Pipeline
- **Integrating RASP and VAPT in Release and Deploy Stage**
  - RASP and VAPT Integration in Release and Deploy Stage
  - Benefits of RASP and VAPT Integration in Release and Deploy Stage

- Conclusion

## Module 10: Implementing DevSecOps Monitoring and Feedback

- **Integrating Infrastructure as Code (IaC)**
  - What is Infrastructure as Code?
  - Why Integrate IaC into DevSecOps?
  - Tools for IaC Integration in DevSecOps
  - Challenges in IaC Integration into DevSecOps
  - Best Practices for IaC Integration into DevSecOps
- **Integrating Configuration Orchestration**
  - What is Configuration Orchestration?
  - How Does Configuration Orchestration Increase Security Posture?
  - Tools for Configuration Orchestration
- **Integrating Security in Operate and Monitor Stage**
  - Securing Operations and Monitoring
  - Importance of Security in Operate and Monitor Stage
  - Benefits of Automated Security Practices
- **Integrating Compliance as Code (CaC)**
  - What is Compliance as Code?
  - Benefits of Compliance as Code
- **Integrating Logging, Monitoring, and Alerting**
  - Integrated Logging, Monitoring, and Alerting During Application Development
  - Integrated Logging, Monitoring, and Alerting When an application is in Production
  - Tools for Securing Operations and Monitoring
- **Integrating Continuous Feedback Loop**
  - Continuous Feedback Loop
  - Creating a Continuous Feedback Loop
  - Integrating Continuous Feedback Loop into Application Development Lifecycle
  - Conclusion