



Cloud DevOps Engineer

Syllabus

**BEFORE YOU START****Overview:**

Companies are looking for talented DevOps engineers to remain competitive in this agile world. Enroll now to operationalize infrastructure at scale and deliver applications and services at high velocity, an essential skill for advancing your career. Learn to design and deploy infrastructure as code, build and monitor CI/CD pipelines for different deployment strategies, and deploy scalable microservices using Kubernetes.

Prerequisites

A WELL-PREPARED LEARNER HAS EXPERIENCE WITH:

- Application Architecture
- Basic programming in Python or Javascript
- Linux command line proficiency
- Networking basics
- Database basics

Educational Objectives

A graduate of this program will be able to design and deploy infrastructure as code on AWS, build and monitor CI/CD pipelines for different deployment strategies on the AWS platform using GitHub Actions, and deploy scalable microservices using Kubernetes on the AWS platform.

**LENGTH OF PROGRAM*:****4 months****SKILL LEVEL:****Intermediate****SCHOOL:****Cloud Computing****SOFTWARE/HARDWARE AND VERSION REQUIREMENTS:**

There are no software and version requirements to complete this Nanodegree program. All coursework and projects can be completed via Student Workspaces in the Udacity online classroom. Udacity's basic tech requirements can be found at <https://www.udacity.com/tech/requirements>.

*The length of this program is an estimation of total hours the average student may take to complete all required coursework, including lecture and project time. If you spend about 5-10 hours per week working through the program, you should finish within the time provided. Actual hours may vary.

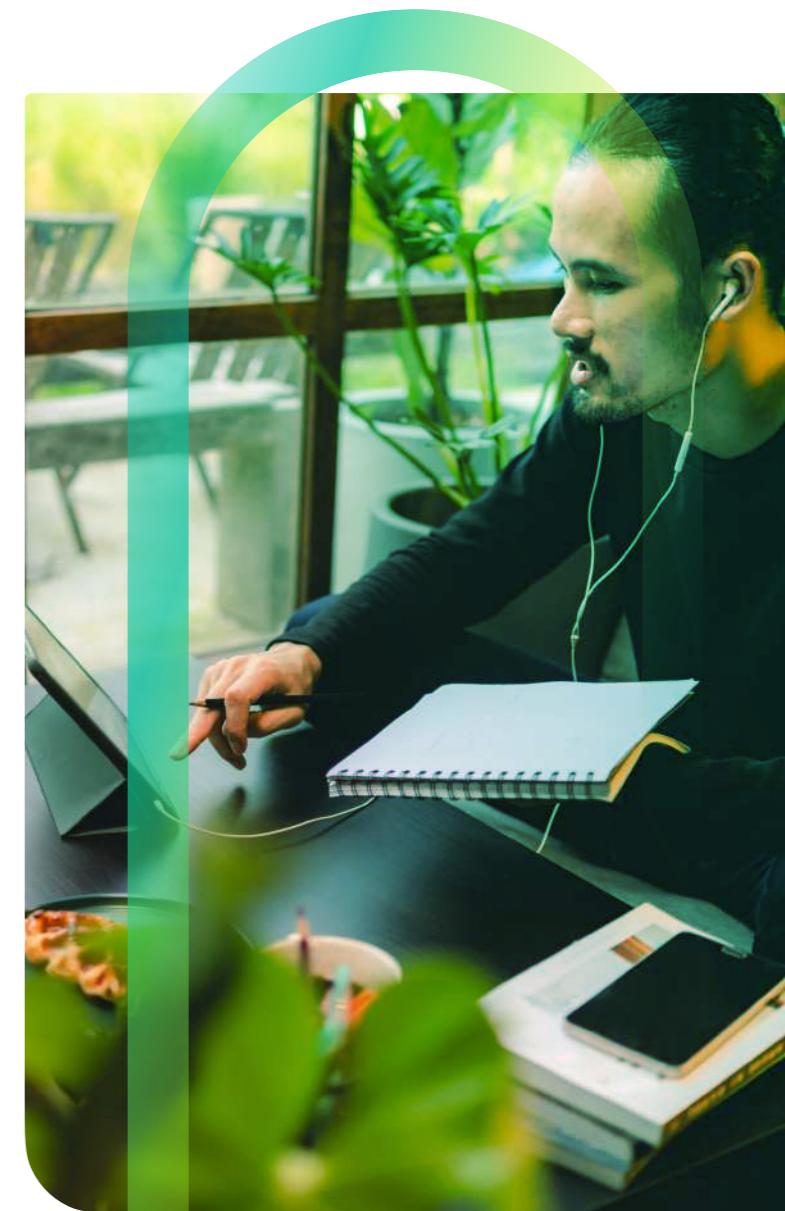
Course #1: Cloud Fundamentals

The cloud has become a key enabler for innovation with beneficial features like high availability, unlimited capacity, and on-demand scalability and elasticity. In this course, students will learn the fundamentals of cloud computing while being introduced to computing power, security, storage, networking, messaging, and management services in the cloud. While learning the fundamentals, students will explore tools and services offered by Amazon Web Services (AWS) through interactive hands-on exercises. By the end of the course, students will have deployed their first website to AWS.

PROJECT #1

Deploy Static Website on AWS

The cloud is perfect for hosting static websites that only include HTML, CSS, and JavaScript files that require no server-side processing. In this project, students will deploy a static website to AWS. First, they will create a S3 bucket, configure the bucket for website hosting, and secure it using IAM policies. Next, they will upload the website files to their bucket and speed up content delivery using AWS's content distribution network service, CloudFront. Lastly, they will access their website in a browser using the unique S3 endpoint.



Supporting Lesson Content

CLOUD COMPUTING

- Identify the types of Cloud computing
- Detail the benefits of using public, private, and hybrid cloud deployment models
- Describe the shared responsibility model and identify the responsibility of the user

SERVERS IN THE CLOUD AND COMPUTE SERVICES

- Launch and connect to an EC2 Instance via PuTTY or SSH
- Leverage the EBS dashboard to review a new EBS volume
- Launch a virtual server within a secure network
- Create and test a Lambda function
- Deploy a sample app to Elastic Beanstalk

STORAGE & CONTENT DELIVERY

- Upload files to a newly created S3 bucket
- Create a cloud-based NoSQL Database leveraging DynamoDB
- Create a MySQL database using RDS
- Access private S3 data via a CloudFront distribution

SECURITY

- Describe AWS security services and their uses
- Create an IAM policy and review the generated JSON
- Attach a custom policy to a new user
- Detail the benefits of using sign-on services for enterprise and application environments

NETWORKING & ELASTICITY

- Identify the benefits of networking in the cloud
- Create an Auto Scaling Group
- Configure and enable a Network Load Balancer
- Automatically launch Amazon EC2 instances in response to specified conditions

MESSAGING & CONTAINERS

- Identify the purpose and uses of SNS messaging
- Send alerts via SNS by creating, subscribing, and publishing an alert message to a topic

AWS MANAGEMENT

- Create a Cloud Watch event to notify via an SNS topic when an EC2 instance is created
- Create an S3 bucket with AWS CloudFormation
- Create an S3 bucket via the CLI

Course #2: Deploy Infrastructure as Code (IAC)

Learn how to provision cloud infrastructure resources using code and industry standard practices. Learners will gain a thorough understanding of CloudFormation essentials, including templates, stacks, and intrinsic functions. They will also acquire expertise in creating networking and web application architectures that conform to real-world best practices, and represent them visually using infrastructure diagrams. After that, they will use CloudFormation and bash scripting to implement all the required networking, compute, load balancing, storage and database resources for their real world design.

PROJECT #2

Deploy a high-availability web app using CloudFormation

In this project, learners will need to apply all the skills they learned throughout the course, demonstrating their competence in creating infrastructure diagrams, implementing infrastructure with CloudFormation, and using scripts to facilitate automation. The project will require the learners to design an infrastructure diagram for the solution, and then utilize CloudFormation to deploy a dummy server on an EC2 instance, using networking, auto scaling, load balancer and storage resources. They will also need to build their own bash scripts to manage their CloudFormation stacks.



Supporting Lesson Content

GETTING STARTED WITH CLOUDFORMATION

- Describe Infrastructure as a Code (IAC) as one of the best practices used in the DevOps model.
- Configure the AWS CLI using the AWS Identity and Access Management (IAM) service.
- Explain the fundamentals of CloudFormation.
- Contrast the manual vs. automated (script-based) provisioning of cloud resources.

SERVERS AND SECURITY GROUPS

- Create security groups with CloudFormation using the least privilege principle.
- Centralize server configuration using Launch Templates in CloudFormation.
- Deploy highly available servers with AutoScaling Groups in CloudFormation.
- Apply load balancing principles to servers using CloudFormation.

INFRASTRUCTURE DIAGRAMS

- Assess foundational resources for your infrastructure deployments in AWS.
- Design cloud architecture diagrams to implement business requirements following real-world best practices.

NETWORKING INFRASTRUCTURE

- Use the AWS CloudFormation CLI to trigger infrastructure deployments.
- Deploy basic networking AWS resources using CloudFormation (VPC, subnets, gateways, and routing).
- Parametrize your CloudFormation stacks using a JSON file.

Communicate CloudFormation stacks using template outputs.

STORAGE AND DATABASES

- Deploy highly available RDS databases using CloudFormation.
- Configure secure S3 buckets using CloudFormation.

Preview changes to CloudFormation stacks using change sets.

- Automate infrastructure workflows using parameterized scripts.

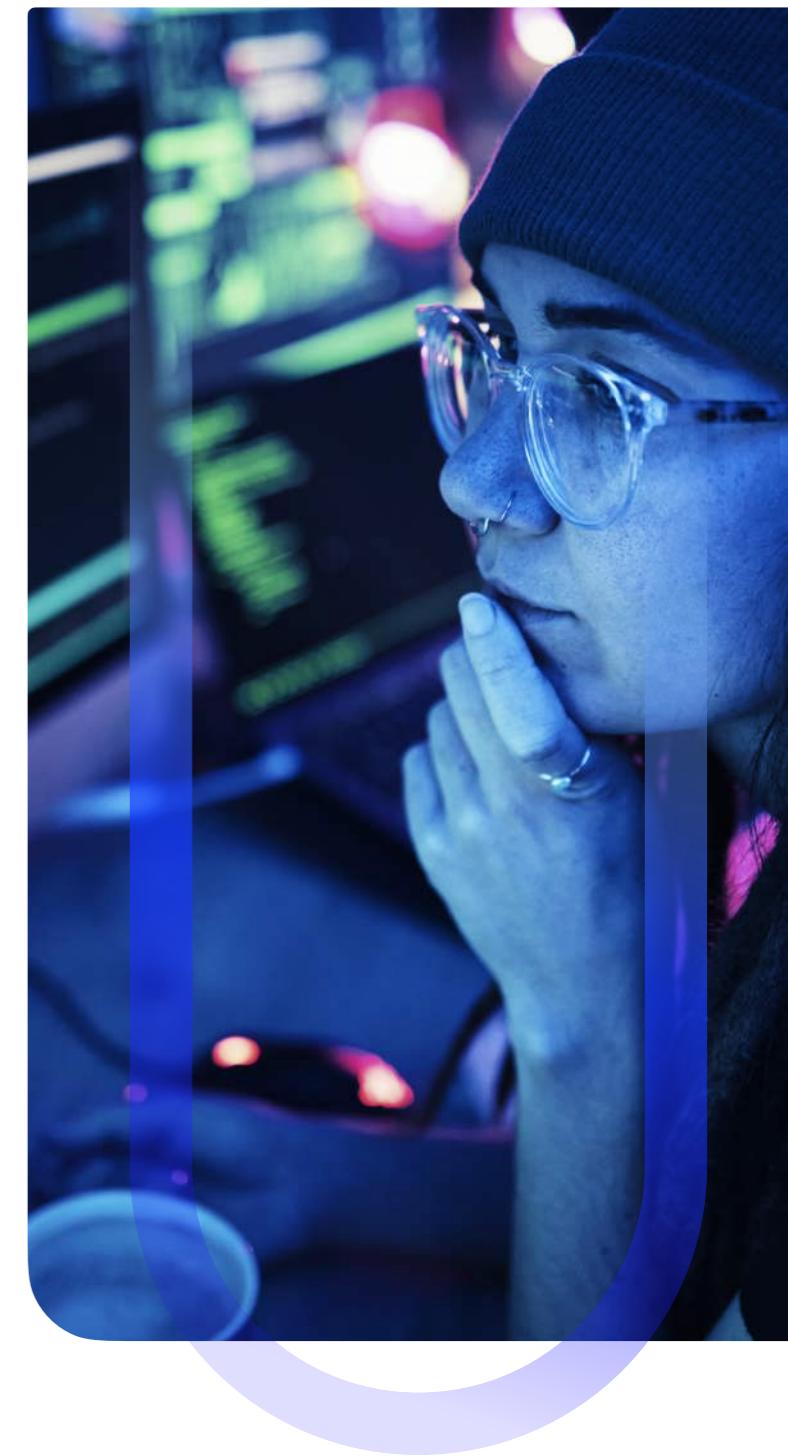
Course #3: Microservices at Scale using AWS & Kubernetes

In this course you will learn how to set up infrastructure to deploy and manage microservice applications with Kubernetes. You will first focus on how containers and microservices have changed how applications are deployed. Students then learn how Kubernetes is used to tie everything together. The course will have hands-on exercises to learn how to build applications with Docker, automate the Docker build process, and use Kubernetes to manage deployments.

PROJECT #3

Operationalizing a Coworking Space Microservice

In this project, students will take an existing application and deploy it with Kubernetes. The existing application is a set of APIs that enables users to request one-time tokens and administrators to authorize access to a coworking space. This service follows a microservice pattern and the APIs are split into distinct services that can be deployed and managed independently of one another. Students will demonstrate course skills by going through the end to end process to deploy this application to Kubernetes. This will be accomplished by building Docker images of the application with AWS CodeBuild, setting up a Kubernetes cluster, and deploying to the cluster with Docker images and Helm Charts.



Supporting Lesson Content

CONTAINERS FOR DEVOPS ON AWS

- Identify the need for and use of Docker
- Use Docker to create images
- Push images to a container repository
- Use Docker to create containers

MICROSERVICES FOR DEVOPS ON AWS

- Understand why microservices are popular with developers
- How to use build tools to efficiently build multiple Docker images

KUBERNETES ON AWS FUNDAMENTALS

- How to operationalize Docker containers with container orchestration
- Creating and managing a Kubernetes cluster with EKS
- Deploying Docker images with Kubernetes

OPERATIONALIZING KUBERNETES

- Understand how to maintain Kubernetes applications in production
- Use Helm charts to simplify deployments

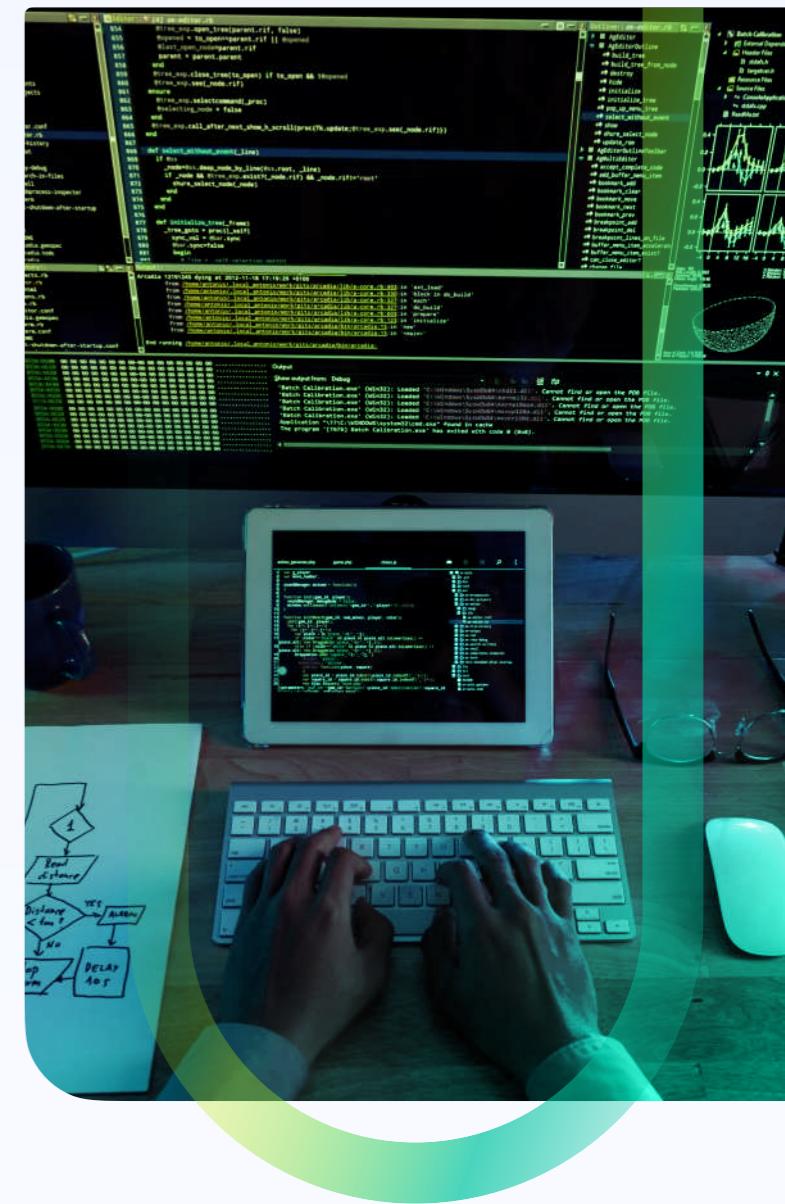
Course #4: Building a Continuous Integration Pipeline

In this course, you will learn how to stretch for Continuous Delivery as your “true north” while you build CI/CD systems that automate the “stuff” between developers and the cloud. You’ll gain essential knowledge and put it into practice as you learn about Continuous Integration and Continuous Deployment. You’ll combine your existing knowledge with new tools like GitHub Actions to implement and deploy to variety of environments. You’ll also learn how to keep an eye on your deployments with proactive monitoring and alerting.

PROJECT #4

Movie Picture Pipeline

In this project, students will design and implement a CI/CD pipeline using GitHub Actions. Students will demonstrate the skills they learned in this course to execute the deployment and verification of frontend and backend applications to an existing Kubernetes cluster.



Supporting Lesson Content

INTRODUCTION TO CI/CD

- Identify use cases for CI/CD
- Describe the principles of Continuous Delivery

CONTINUOUS INTEGRATION AND CONTINUOUS DEPLOYMENT

- Design a CI/CD pipeline using a visual mockup
- Articulate the benefits of CI/CD to stakeholders

ENABLING CONTINUOUS DELIVERY WITH DEPLOYMENT PIPELINES

- Create/deploy the necessary infrastructure/resources for building CI/CD pipelines
- Secure a CI/CD pipeline by leveraging secret storage
- Utilize caching strategies to speed-up application build times
- Create alerts to monitor your pipelines
- Automate deploying an application to multiple AWS accounts

BUILDING A CONTINUOUS INTEGRATION PIPELINE

- Create a variety of YAML Workflows
- Design a git branching strategy to support a development team's workflow
- Create a custom GitHub action and deploy it to GitHub Marketplace
- Design and execute a CI pipeline based on a given set of requirements

Course #1 Instructor

**Kesha Williams.** INSTRUCTOR

Kesha has over 20 years experience in software development and is a software engineering manager at Chick-fil-A, routinely leading innovation teams in proving out the use of cloud services to solve complex business problems. She was recently named an Alexa Champion by Amazon.

Course #2 Instructor

**Carlos Rivas.** INSTRUCTOR

Carlos is a Senior Solutions Architect at Infiniti Consulting where he helps institutions move traditional data centers to the cloud. He has worked for several large telecommunication providers managing and configuring network infrastructure, using Java, Groovy, Python, Perl, and PHP.

Course #3 Instructor

**Justin Lee.** DATA PLATFORM ENGINEER

Justin Lee designs and builds modern scalable systems and consults for Fortune 500 companies. He currently works in the Silicon Valley as a platform engineer to enable large volumes of users' data workflows. He has a BS in Computer Science from UCLA and is often mentoring and teaching developers through Codementor.

Course #4 Instructor

**Byron Sommardahl.** CHIEF TECHNOLOGY OFFICER

Byron is co-founder and Chief Technology Officer of Acklen Avenue, an agile software development company. Byron has been developing software since he was 9 years old, and is a true believer in anything that improves software maintainability, usability, and delivery.



Learn More at

WWW.UDACITY.COM